The Internet Protocol Journal

December 2003 Volume 6, Number 4

A Quarterly Technical Publication for Internet and Intranet Professionals

In This Issue

You can download IPJ back issues and find subscription information at: **www.cisco.com/ipj**

From The Editor

I will remember 2003 as the year when high-speed Internet access became widely available in public locations such as airports, hotels, and coffee shops. As a frequent traveler, I really appreciate not having to find a suitable telephone jack and corresponding country-specific telephone adapter plug in order to get my e-mail. The IEEE 802.11 "WiFi" standard has truly arrived. I even stayed in a new hotel in Norway that provided WiFi access in every room by placing base stations in the hallways. When I first stepped into my hotel room and noticed that it had only a *digital* telephone and no sign of any Ethernet jacks I worried, but a quick check revealed that I could purchase a scratch-off card at reception that provided me with a username and password valid for 24 hours. A clear example of a "technology generation leap."

The year 2003 was also the year in which unsolicited e-mail, or "spam," became a major problem for all Internet users. Various filtering systems have thankfully been devised and deployed, but this problem has no easy solution. It will be interesting to see what impact new antispam legislation will have over the coming months and years.

The first article presents an in-depth look at the IP Version 4 address space and its measured and projected consumption rate. When work first started on the design of IP Version 6, projections indicated that we'd run out of IPv4 addresses within a few years. Geoff Huston takes a fresh look at this in an article entitled "IPv4—How long do we have?"

The job of System Administrator, or "sysadmin," is a challenging one, and if your job includes keeping the network running 24 hours a day, you will probably appreciate some of the tips in our second article, entitled "Low-Tech Network Maintenance."

For the second time recently, Queen Elizabeth II has honored an Internet pioneer. Tim Berners-Lee, the inventor of the World Wide Web and director of the *World Wide Web Consortium* (W3C), was made a *Knight Commander, Order of the British Empire* in the 2004 New Years Honours list. (See "Fragments," page 28).

Which brings us to the IPJ publication schedule. If you are a regular subscriber to the IPJ, you probably have noticed a somewhat irregular publishing schedule in 2003. This December 2003 issue is indeed being published in January 2004. This results from our effort to produce timely quality articles in a world where the experts are not staff writers. Of course, you should still expect to receive four issues per year, and your feedback to **ipj@cisco.com** will help make IPJ even better.

—Ole J. Jacobsen, Editor and Publisher
ole@cisco.com

IPv4—How long do we have?

by Geoff Huston, Telstra

One of those stories that keeps on appearing from time to time is the claim that somewhere in the world, or even all over the world, we are "running out of IP addresses," referring to the consumption of unallocated IPv4 addresses[1]. In one sense this is a pretty safe claim, in that the IPv4 address pool is indeed finite, and, as the IPv4 Internet grows it makes continual demands on previously unallocated address space. So the claim that the space will be exhausted at some time in the future is a relatively safe prediction. But the critical question is not "if" but "when," because this is a question upon which many of our current technology choices are based.

Given this revived interest in the anticipated longevity of the IPv4 address space, it is timely to revisit a particular piece of analysis that has been a topic of some interest at various times over the past decade or more. The basic question is: "How long can the IPv4 address pool last in the face of a continually growing network?" This article looks at one approach to attempt to provide some indication of "when." Like all predictive exercises, many assumptions have to be made, and the approach described here uses just one of numerous possible predictive models—and, of course, the future is always uncertain.

The IPv4 Address Space

The initial design of IPv4 was extremely radical for its time in the late 1970s. Other contemporary vendor-based computer networking protocols were designed within the constraints of minimizing the packet header overhead in order to improve the data payload efficiency of each packet. At the time address spans were defined within the overall assumption that the networks were deployed as a means of clustering equipment around a central mainframe. In many protocol designs 16 bits of address space in the packet headers was considered to be extravagant. To use a globally unique address framework of 32 bits to address network hosts was, at the time, a major shift in thinking about computer networks from a collection of disparate private facilities into a truly public utility.

To further add to the radical nature of the exercise, the Internet Network Information Center was prepared to hand out unique blocks of this address space to anyone who submitted an application. Address deployment architectures in other contemporary protocols did not have the address space to support such address distribution functions, nor did they even see a need for global uniqueness of computer network addresses. Network administrators numbered their isolated corporate or campus networks starting at the equivalent of "1," and progressed onward from there. Obviously network splits and mergers caused considerable realignment of these private addressing schemes, with consequent disruption to the network service.

By comparison, it seemed, the address architecture of the Internet was explicitly designed for interconnection. But even with 32 bits to use in an address field, getting the right internal structure for addresses is not as straightforward as it may initially seem.

The Evolution of the IPv4 Address Architecture

IP uses the address to express two aspects of a connected device: the identity of this device (endpoint identity) and the location within the network where this device can be reached (location or forwarding identity). The original IP address architecture used the endpoint identity to allow devices to refer to each other in end-to-end application transactions, whereas within the network the address is used to direct packet-forwarding decisions. The address was further structured into two fields: a *network* identifier and a *host* identifier within that network. The first incarnation of this address architecture used a division at the first octet: the first 8 bits were the network number and the following 24 bits were the host identifier. The underlying assumption was one of deployment across a small number of very large local networks. This view was subsequently refined, and the concept of a class-based address architecture was devised for the Internet. Half of the address space was left as a 8/24-bit structure, called the *Class A* space (allowing for up to 127 networks each with 16,777,216 host identities). A quarter of the remaining space used a 16/16-bit split (allowing for up to 16,128 networks, each with up to 65,536 hosts), defining the *Class B* space. A further eighth of the remaining space was divided using a 24/8-bit structure (allowing for 2,031,616 networks, each with up to 256 hosts), termed the *Class C* space. The remaining eighth of the space was held in reserve.

This address scheme was devised in the early 1980s, and within a decade it was pretty clear that there was a problem with impending exhaustion. The reason was an evident run on Class B addresses. Although very few entities could see their IP network spanning millions of computers, the personal desktop computer was now a well-established part of the landscape, and networks of just 256 hosts were just too small. So if the Class A space was too big, and the Class C too small, then Class B was the only remaining option. In fact, the Class B blocks were also too large, and most networks that used a Class B address consumed only a few hundred of the 65,535 possible host identities within each network. The addressing efficiency of this arrangement was very low, and a large amount of address space was being consumed in order to number a small set of devices. Achieving even a 1 percent host density (expressed as a ratio of number of addressed hosts to the total number of host addresses available) was better than normal at the time, and 10 percent was considered pretty exceptional.

Consequently, Class B networks were being assigned to networks at an exponentially increasing rate. Projections from the early 1990s forecast exhaustion of the Class B space by the mid-1990s. Obviously there was a problem, and the *Internet Engineering Task Force* (IETF) took on the task of finding some solutions. Numerous responses were devised by the IETF.

As a means of mitigation of the immediate problem, the IETF altered the structure of an IP address. Rather than having a fixed-length network identifier of 8, 16, or 24 bits, the network part of the address could be any length at all, and a network identifier was now the couplet of an IP address field containing a network part and the bit length of the network part. The boundary between the network and host part could change across the network, so rather than having "networks" and "subnetworks" as in the class-based address architecture, there was the concept of a variable length network mask. This was termed the "classless" address architecture (or "CIDR"), and the step was considered to be a short-term expediency to buy some additional time before address exhaustion. The longer-term plan was to develop a new IP architecture that could encompass a much larger connectivity domain than was possible with IPv4.

We now have IPv6 as the longer-term outcome. But what has happened to the short-term expediency of the classless address architecture in IPv4? It appears to have worked very well indeed so far, and now the question is: how long can this supposedly short-term solution last?

Predictions of Address Consumption

Predicting the point of IPv4 address exhaustion has happened from time to time since the early 1990s within the IETF[2]. The initial outcomes of these predictive exercises were clearly visible by the mid-1990s: the classless address architecture was very effective in improving the address utilization efficiency, and the pressures of ever-increasing consumption of a visibly finite address resource were alleviated. But a decade after the introduction of CIDR addressing, it is time to understand where we are heading with the consumption of the underlying network address pool.

Dividing up the Address Space

There are three stages in address allocation. The pool of IP addresses is managed by the *Internet Assigned Numbers Authority* (IANA). Blocks of addresses are allocated to *Regional Internet Registries* (RIRs), who in turn allocate smaller blocks to *Local Internet Registries* (LIRs) or *Internet Service Providers* (ISPs).

Currently 3,707,764,736 addresses are managed in this way. It is probably easier to look at this in terms of the number of "/8 blocks," where each block is the same size as the old Class A network, namely 16,777,216 addresses. The total address pool is 221 /8s, with a further 16 /8s reserved for multicast use, 16 /8s held in reserve, and 3 /8s designated as not for use in the public Internet.

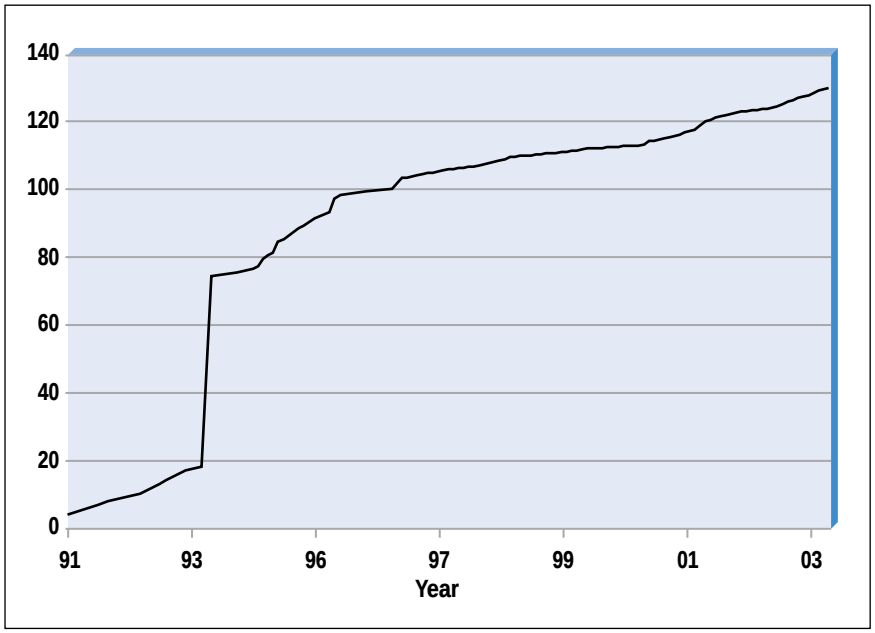
In looking at futures, there are three sources of data concerning address consumption:

- How quickly is the IANA passing address blocks to the RIRs, and when will IANA run out?
- How quickly are the RIRs passing address blocks to LIRs, and when will this run out?
- How much address space is actually used in the global Internet, and how quickly is this growing? When will this run out?

The IANA Registry

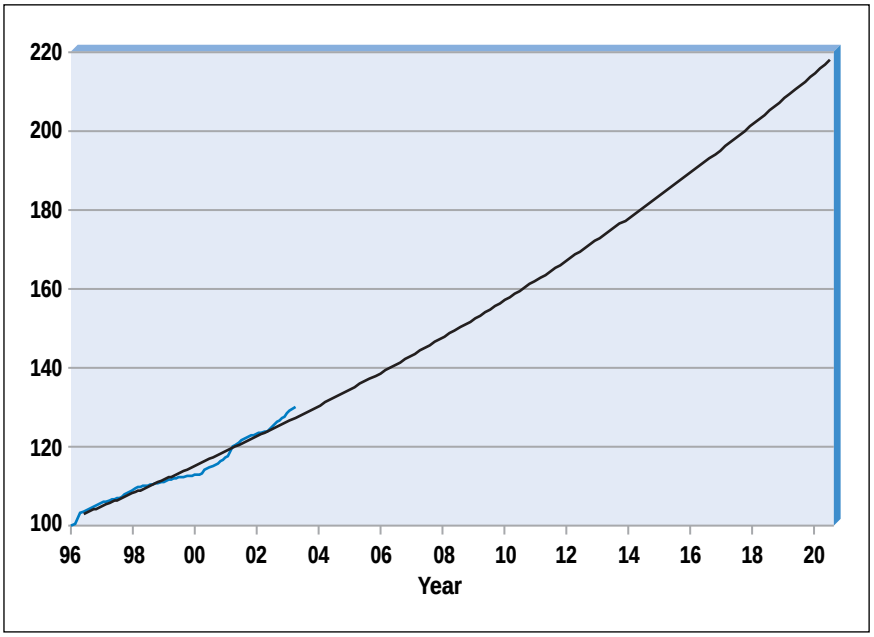
So the first place to look is the IANA registry file[3]. This registry reveals that of these 221 /8 blocks, 89 /8 blocks are still held as unallocated by the IANA, 129.9 /8 blocks have been allocated, and the remaining 2.1 / 8 blocks are reserved for other uses. The IANA registry also includes the date of allocation of the address block, so it is possible to construct a time series of IANA allocations, as shown in Figure 1.

Figure 1: IANA Allocated IPv4 /8 Address Blocks



Interestingly, there is nothing older than 1991 in this registry. This exposes one of the problems with analyzing registry data, in that there is a difference between the current status of a registry and a time-stamped log of the transactions that were made to the registry over time. The data published by the IANA is somewhere between the two, and the log data is incomplete; in addition, the current status of some address blocks is unclear. It appears that the usable allocation data starts in 1995. So if we take the data starting from 1995 and perform a linear regression to find a best fit of an exponential projection, it is possible to make some predictions as to the time it will take to exhaust the remaining unallocated 89 /8s. (Figure 2).

Figure 2: IANA Allocated IPv4 /8 Address Blocks



It is worth a slight digression into the method of projection being used here. The technique is one of using a best fit of an exponential growth curve to the data. The underlying assumption behind such a projection is that the growth rate of the data is proportional to the size of the data, rather than being a constant rate. In network terms, this assumes that the rate of consumption of unallocated addresses is a fixed proportion of the number of allocated addresses, or, in other words, the expansion rate of the network is a proportion of its size, rather than being a constant value. Such exponential growth models may not necessarily be the best fit to a network growth model, although the data since 1995 does indicate an underlying exponential growth pattern. Whether this growth model will continue into the future is an open issue.

The projection of 2019 as the date for consumption of the unallocated address space using this technique is perhaps surprising, because it seems that the network is bigger now than ever, yet the amount of additional address space required to fuel further accelerating growth for a further decade is comparatively small. This is true for many reasons, and the turning point when these aspects gained traction in the Internet appeared to be about 1995. They include:

- The first 1.6 billion addresses (equivalent to some 100 /8 blocks) were allocated using the class-based address architecture. Since this date address allocation has used a classless architecture, and this has enabled achievement of significantly improved efficiencies in using the address space.
- The RIRs came into the picture, and started using conservation-based policies in address allocations. The RIR process requires all address applicants to demonstrate that they can make efficient and effective use of the address space, and this has dampened some of the wilder sets of expectations about the address requirements of an enterprise.

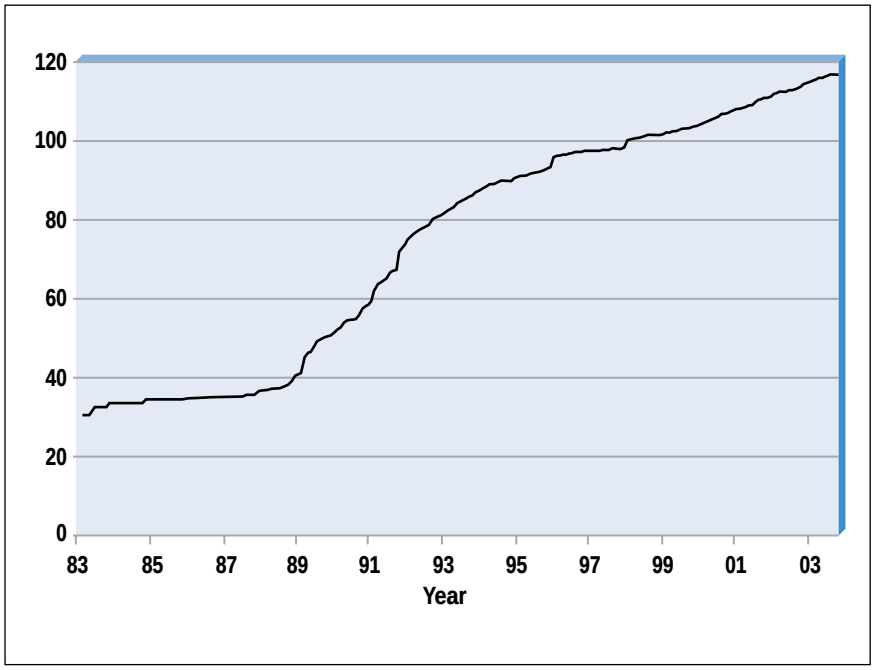
- Address compression technologies became widely deployed. Dynamic *Network Address Translation* (NAT) devices have, for better or worse, become a common part of the network landscape. NAT devices allow large "semi-private" networks to use a very small poll of public addresses as the external view of the network, while using private address space within the network. *Dynamic Host Configuration Protocol* (DHCP) has allowed networks to recycle a smaller pool of addresses across a larger set of intermittently connected devices.

Whether these factors will continue to operate in the same fashion in the future is an open question. Whether future growth in the use of public address space operates from a basis of a steadily accelerated growth is also an open question. The assumption made in this exercise is that the projections depend on continuity of effectiveness of the RIR policies and their application, continuity of technology approaches, and absence of disruptive triggers. Although the RIRs have a very well-regarded track record and there are strong grounds for confidence that this will continue, obviously the latter two assumptions about technology and disruptive events are not all that comfortable. With that in mind, the next step is to look at the RIR assignment data.

The RIR Registries

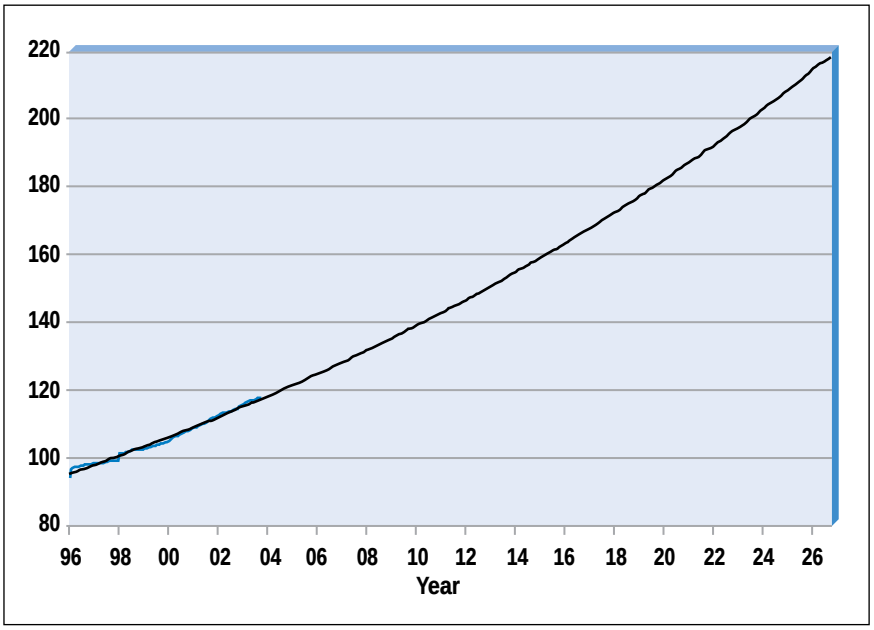
The RIRs also publish a registry of their transactions in "stats" files. For each currently allocated or assigned address block the RIRs have recorded, among other items, the date of the RIR assignment transaction that assigned an address block to a LIR or ISP. Using this data we can break up the 129.9 /8 blocks further, and it is evident that the equivalent of 116.7 /8 blocks have been allocated or assigned by the RIRs, and the remaining space, where there is no RIR allocation or assignment record, is the equivalent of 13.2 /8 blocks. These transactions can again be placed in a time series, as shown in Figure 3.

Figure 3: RIR Assigned IPv4 /8 Address Blocks



The post-1995 data used to extrapolate forward using the same linear regression technique described previously to find a curve of best bit using the same underlying growth model assumptions yields:

Figure 4: RIR Assigned IPv4 /8 Address Blocks—Projection



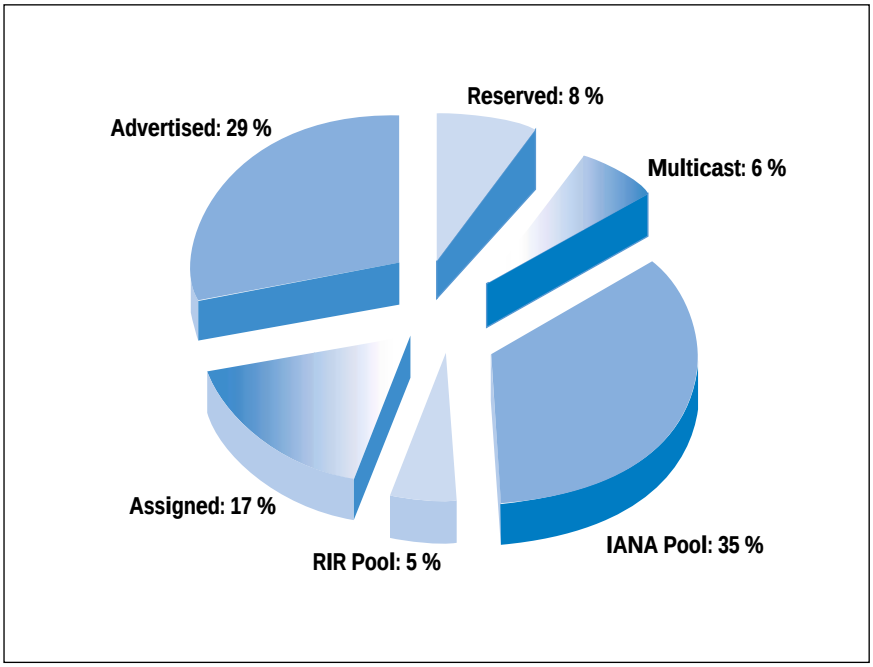
This form of extrapolation gives a date of 2026 for the time at which the RIRs will exhaust the number pool. Again the same caveats about the use of this approach as a reliable predictor apply here, and the view forward is based on the absence of large-scale disruptions, or some externally induced change in the underlying growth models for address demand.

The BGP Routing Table

When addresses are assigned to end networks, the expectation is that these addresses will be announced to the network in the form of routing advertisements. So some proportion of these addresses is announced in the Internet routing table. The next task is to establish the trends of the amount of address space covered by the routing table. The approach used has been to take a single view of the address span of the Internet. This is the view from one point, inside the AS1221 network operated by Telstra.

The data as of October 2003 shows that some 29 percent of the total IPv4 address space is announced in the *Border Gateway Protocol* (BGP) routing table, whereas 17 percent has been allocated to an end user or LIR but is not announced on the public Internet as being connected and reachable. A total of 5 percent of the address space is held by the RIR's pending assignment or allocation (or at least there is no RIR recorded assignment of the space), while 35 percent of the total space remains in the IANA unallocated pool. A further 8 percent of the space is held in reserve (Figure 5).

Figure 5: IPv4 /8 Address Space



This BGP data is based on an hourly inspection of the amount of address space advertised within the Internet routing table. The data collection commenced in late 1999, and the data gathered so far is shown in Figure 6. The problem with this data is that there is some considerable amount of fluctuation in the amount of address space advertised over time. The major step changes are due to a small number of /8 advertisements that periodically are announced and withdrawn in BGP. In order to obtain reasonable data for generating projections, some noise reduction on this data needs to be undertaken. The approach used has been to first filter the data using a constant value of 18 /8 prefix announcements, and then use a sliding average function to create a smoothed time series. This is indicated in Figure 7.

The critical issue when using this data for projection is to determine what form of function can provide a best fit to the data. A good indication of the underlying trends in the data can be found by analyzing the first-order differential of the data. An underlying increasing growth model would have an increasing first-order differential, whereas a decreasing growth model would have a negatively inclined differential. A least-squares best-fit analysis of the data shows that the growth rates have not been consistent over the past three years. A reasonable fit for this data appears to be a constant growth model, or a linear growth projection, with a consumption rate of some 3 /8 blocks per year.

Figure 6: Advertised IPv4 /8 Address Space (/8 Blocks)

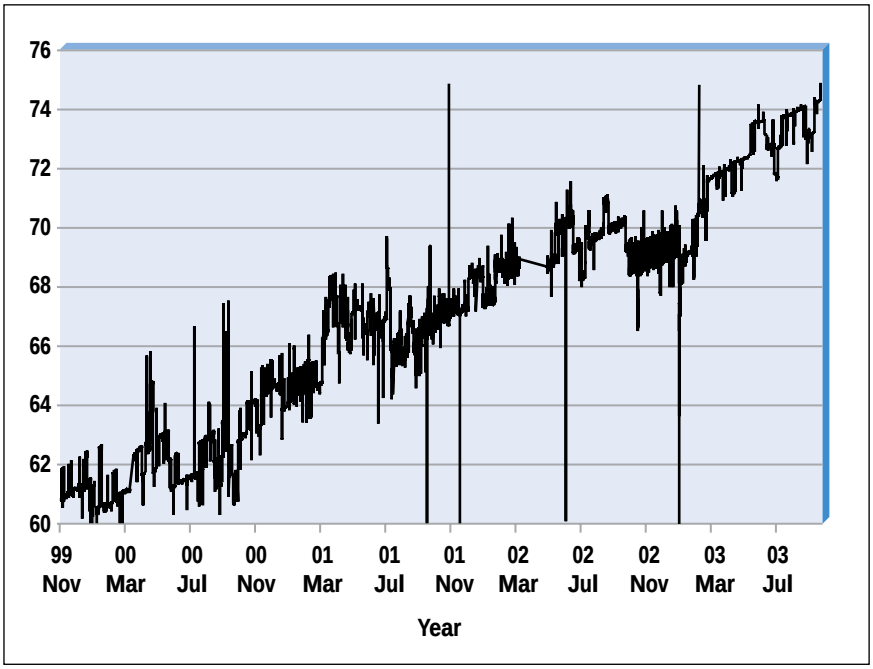
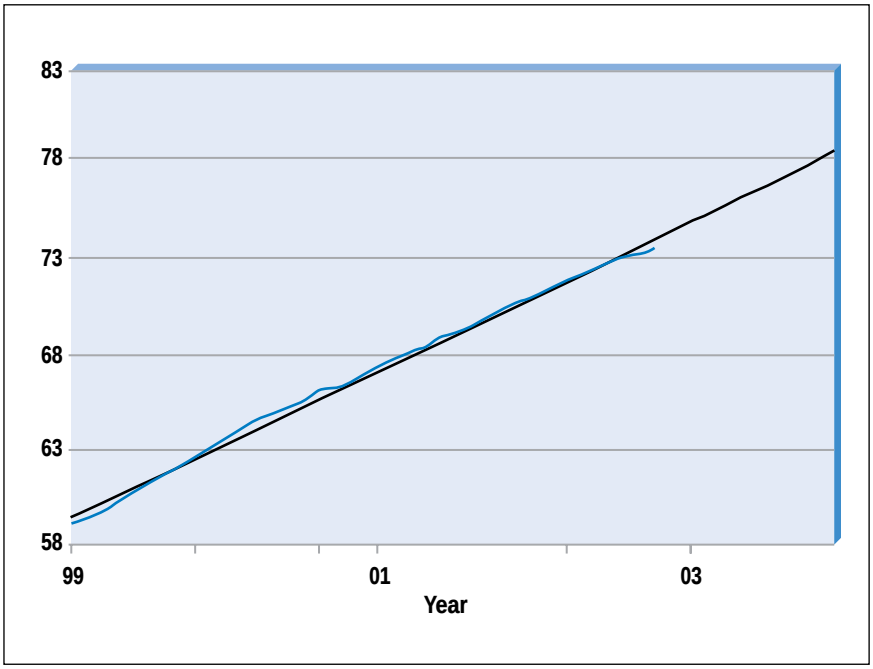


Figure 7: Smoothed IPv4 /8 Advertised Address Blocks



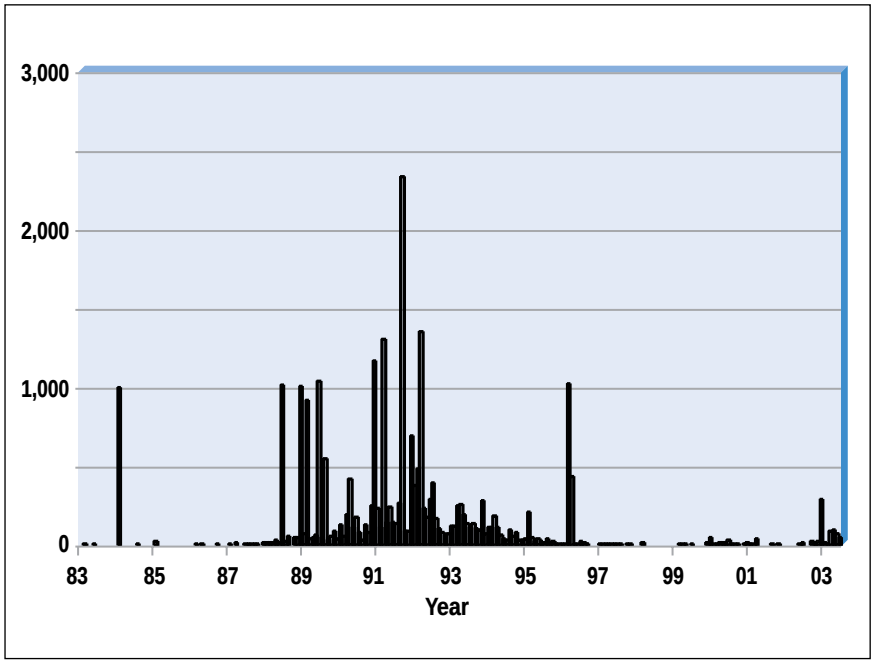
Combining the Three Views

One question remains before we complete the projections for IPv4 address space. There are 43.3 /8 blocks, or some 17 percent of the total IPv4 address space that has been allocated for use, but is not visible in the Internet routing table. This is a very significant amount of address space, and if it is growing at the same rate as the advertised space, then this will have a significant impact on any overall model of consumption of the use of address space.

The question here is whether this "invisible" address pool is a legacy of the address allocations policies in place before the RIR system came into operation in the mid 1990s, or some intrinsic inefficiency in the current system. If it is the latter, then it is likely that this pool of unannounced addresses will grow in direct proportion to the growth in the announced address space, whereas if it is the former, then the size of the pool will remain relatively constant in the future.

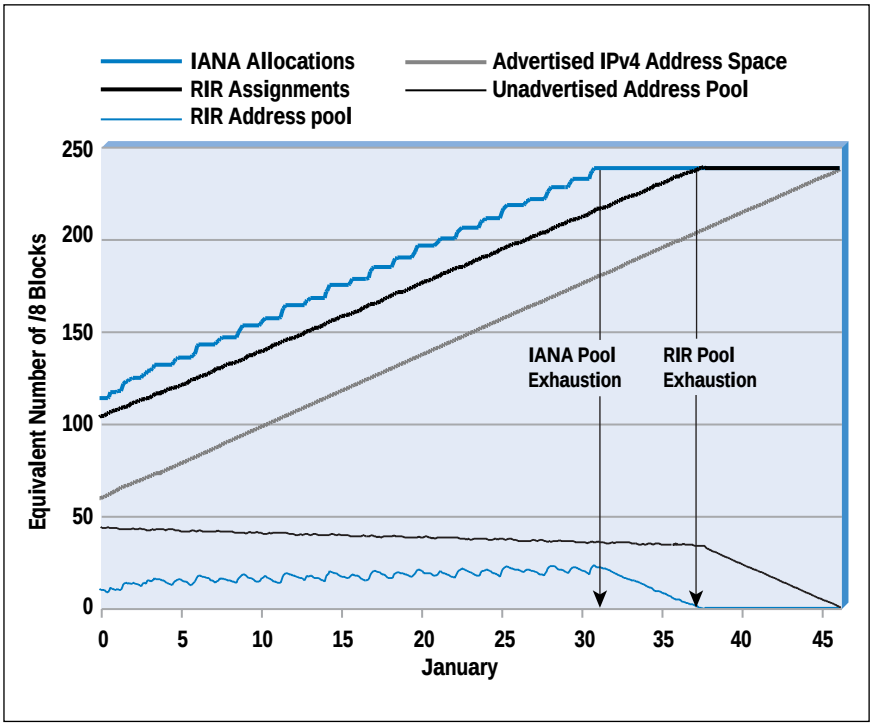
We can look back through the RIR allocation data and look at the allocation dates of unannounced address space (Figure 8). This view indicates that the bulk of the space is a legacy of earlier address allocation practices, and that since 1997, when the RIR operation was fully established, there is an almost complete mapping of RIR allocated address space to BGP routing announcements. The recent 2003 data indicates that there is some lag between recent allocations and BGP announcements, most probably due to the time lag between an LIR receiving an allocation and subsequent assignments to end users and advertisement in the routing table.

Figure 8: Age Distribution of Unadvertised Address Blocks (/8 Address Blocks)



This confirms that in recent years all the address space that has been assigned by the RIRs appears in the Internet routing table, implying that projections of the amount of address space advertised in the routing table is a good correlation to projections of address space consumption. With this in mind it is now possible to construct a model of the address distribution process, working backward from the BGP routing table address size. From the sum of the BGP table size and the LIR holding pool, we can derive the total RIR-managed address pool. To this number is added the RIR holding pool low size and its low threshold where a further IANA allocation is required. This allows a view of the entire system, projected forward over time, where the central driver for the projection is the growth in the network itself, as described by the size of the announced IPv4 address space. This is shown in Figure 9.

Figure 9: IPv4 Projections of Address Consumption



It would appear that the point of effective exhaustion is the point where the RIRs exhaust available address space to assign. In this model, RIR exhaustion of the unallocated address pool would occur in 2037.

Uncertainties

Of course such projections are based on the underlying assumption that tomorrow will be much like today, and the visible changes that have occurred in the past will smoothly translate to continued change the future. This assumption obviously has some weaknesses, and many events could disrupt this prediction.

Some disruptions could be found in technology evolution. An upward shift in address take-up rates could occur because of an inability of NAT devices to support emerging popular applications. Widespread deployment of peer-to-peer applications implies the need for persistent address presentation, which may imply greater levels of requirement for public address space. The use of personal mobile IP devices (such as PDAs in their various formats) using public IPv4 addresses would place a massive load on the address space, simply because of the very large volumes associated with deployment of this technology[4].

Other disruptions have a social origin, such as the boom and bust cycle of Internet expansion in recent years. Another form of disruption in this category could be the adoption of a change in the distribution function. The current RIR and LIR distribution model has been very effective in limiting the amount of accumulation of address space in holding pools, and allocating addresses based on efficiency of utilization and conformance to the routing topology of the network.

Many other forms of global resource distribution use a geopolitical framework, where number blocks are passed to national entities, and further distribution is a matter of local policy[5]. The disruptive nature of such a change would be to immediately increase the number of "holding" points in the distribution system, locking away larger pools of address space from being deployed and advertised and generating a significant upward change in the overall address consumption rates due to an increase in the inefficiency of the altered distribution function.

The other factor to be aware of is the steadily decreasing "buffer" of unallocated addresses that can be used to absorb the impacts of a disruptive change in address consumption rates. Although at present some 60 percent of the address space—or some 2.6 billion addresses—are available in the unallocated address pools or held in reserve, this pool will reduce over time. If a disruptive event is, for example, a requirement to directly address some 500 million devices, then such an event would reduce the expectancy of address space availability by some years, assuming it occurred within the period when sufficient address space remains to meet such a surge of demand.

The other source of uncertainty is that this form of predictive modeling assumes that the ratios of actual connected devices and the amount of address space deployed to service this device pool remain relatively constant.

This model also assumes some form of continuity of current address allocation polices. This is not a likely scenario, because it is likely that address policies will reflect some notion of balance between the level of current demand against future demands. As the unallocated address pool shrinks it is possible that policies will alter to express the increased level of competitive demand for the remaining resource. Consumption rates would be moderated by such a change in allocation policy. The commonly cited intended evolutionary path for the Internet is to a transition to ubiquitous use of IPv6, and at some point in that transition process it is reasonable to assume that further demands for IPv4 space will dwindle. It may be that at such a "crossover" time allocation policies may then be altered to reflect a drop in both current and future demands for IPv4 address space.

In attempting to assess the possible future path of address allocation policies, it is also evident that, from a market rationalist perspective, there is a certain contrivedness about the current address allocation process. The current address management system assumes a steady influx of new addresses to meet emerging demands, and the overall address utilization efficiency is not set by any form of market force, but by the outcomes of the application of RIR address allocation policies to new requests for address space. A market rationalist could well point to the use of market price as a means of determining the most economically efficient form of utilization of a commodity product. Such a position is based on the observation that the way that the consumer chooses between alternative substitutable services is by a market choice that is generally price sensitive.

If price is removed from an IPv4 address market, the choices made by market players are not necessarily the most efficient choices, and some would argue that the current situation underprices IPv4 at the expense of IPv6.

However, in venturing into these areas we are perhaps straying a little too far from exploring the degree of uncertainty in these predictive exercises. A discussion of the interaction between various forms of distribution frameworks and likely technology outcomes is perhaps a topic for another time.

So just how long does IPv4 have?

The assumptions used here include assuming that the trends in the growth in the advertised space are directly proportional to the future consumption rates for IP addresses, and that the constant growth model remains a best fit for this time series of data. It also assumes a continuation of the current utilization efficiency levels in the Internet, a continuing balance between public address utilization and the use of various forms of address compression, and continuity of current address allocation policies, as well as the absence of highly disruptive events. With all this in mind, then it would appear that the IPv4 world, in terms of address availability, could continue for up to another three decades or so without reaching any fixed boundary of exhaustion.

But it must be remembered that each of these assumptions is relatively sweeping, and to combine them as we have done here is pushing the predictive exercise to its limits, or possibly beyond them. Three decades out is way over the event horizon for any form of useful prediction for the Internet, so if we restrict the question to at most the next five to eight years, then we can answer with some level of confidence that, in the absence of any significant disruptions to the current deployment model of the Internet, there is really no visible evidence that IPv4 will exhaust its address pool by 2010, based on the available address consumption data.

Data Sources

IANA IPv4 Address Registry:
`http://www.iana.org/assignments/ipv4-address-space`

Registry "stats" report files:
APNIC: **`ftp://ftp.apnic.net/pub/apnic/stats`**
ARIN: **`ftp://ftp.arin.net/pub/stats`**
LACNIC: **`ftp://ftp.lacnic.net/pub/stats`**
RIPE NCC: **`ftp://ftp.ripe.net/ripe/stats`**
BGP Address Data: **`http://bgp.potaroo.net`**

Notes

[1] "Tackling the net's number shortage." BBC News, World Edition, 26 October 2003. The item starts with the claim: "BBC ClickOnline's Ian Hardy investigates what is going to happen when the number of net addresses—Internet Protocol numbers—runs out sometime in 2005."
http://news.bbc.co.uk/2/hi/technology/3211035.stm

[2] The work was undertaken in the *Address Lifetime Expectations* (ALE) Working Group of the IETF in 1993–1994. The final outcome from this effort was reported from the December 1994 meeting of this group: "Both models currently suggest that IPv4 addresses would be depleted around 2008, give or take three years."

[3] This registry is online at:
http://www.iana.org/assignments/ipv4-address-space

[4] On the other hand, it is evident that the growth of the Internet in recent years has been fueled by the increasing prevalence of NAT devices. In order for applications to be accepted into common use in today's Internet, they need to be able to function through various NAT-based constraints, and increasing sophistication of applications in operating across NAT devices is certainly evident today.

[5] Such a geopolitical distribution system is used in the E.164 number space for telephony ("ENUM").

GEOFF HUSTON holds a B.Sc. and a M.Sc. from the Australian National University. He has been closely involved with the development of the Internet for the past decade, particularly within Australia, where he was responsible for the initial build of the Internet within the Australian academic and research sector. Huston is currently the Chief Scientist in the Internet area for Telstra. He is also the Executive Director of the Internet Architecture Board, and is a member of the APNIC Executive Committee. He is author of *The ISP Survival Guide*, ISBN 0-471-31499-4, *Internet Performance Survival Guide: QoS Strategies for Multiservice Networks*, ISBN 0471-378089, and coauthor of *Quality of Service: Delivering QoS on the Internet and in Corporate Networks*, ISBN 0-471-24358-2, a collaboration with Paul Ferguson. All three books are published by John Wiley & Sons. E-mail: **gih@telstra.net**

Low-Tech Network Maintenance

by Locum sysadmin

In an ideal world, we all maintain networks composed of shiny, high-end equipment. Server rooms are stacked to the brim with racks of blinking lights. Neat bundles of cable wend their way through cable loops to orderly, labeled patch bays. When the occasional piece of equipment fails, a hot replacement is slotted in by trained technicians, often before users even notice the outage. Sleek, modern servers hum contentedly, offering their services all day, every day. All is well.

And then there are the other environments ...

Imagine, if you will, that you are a programmer, working for a small company. You are perhaps vaguely aware that all is not well with the small network that you use each day, but the system administrator (*sysadmin*, if there is one) is so busy with other duties that addressing your concerns seems to be last on the list. The occasional delay in CVS checkouts or e-mail that just never quite makes it seem like minor issues compared to... well, whatever it is that so occupies the sysadmin.

Or perhaps there is no sysadmin ... the network topology is neither ring, nor star, but more "accreted." It is possible that the nephew of one of the managers was responsible for its setup. Like coral, successive waves of employees have washed over the network, leaving their small additions—a cheap 8-port hub here, some gaffer-taped wiring there.

You become aware that your LAN/WAN environment is a real-world test of how deeply Ethernet hubs may be cascaded. A trip to the server room (or server closet) reveals a mess of cabling that closely resembles blue spaghetti. Access to the outside world can take several forms, but it is not uncommon to find a couple of dialup modems lurking quietly in the mess, unnoticed until a failure in the regular link means a failover to the pleasures of 30 employees sharing a 33.6k modem. The concept of labeling cables never made it to this paleolithic theme park, so if you ever trip on one of the floor-dwelling blue vines, locating its original socket can be a challenging occupation.

The servers themselves seem to be an interactive museum display charting the history of computing up until the late 1990s. Old UNIX boxes spill a mess of cables and hard drives over the bench, generic white-box servers of unknown vintage litter the room, "Powered by Linux" or FreeBSD stickers adorning them. Discolored 15-inch monitors sometimes display a blue screen of death, letting you know that some people still love NT4. Assorted tape drives blink quietly away, backing up regularly, though no one seems quite sure what they are backing up, or how to recover them. An elderly Sun box whiles away its retirement transferring mail and playing host to the occasional crackers who exploit security holes in its ancient *sendmail*, then give up in disgust.

The spare parts for the network might occupy a shelf in the server room, or perhaps they nestle on top of a rack unit. A motley assortment of chewed-looking Category 5 cables, network cards so ancient that their manufacture date is in Roman numerals, and a sculpture of BNC connectors—the thought of turning here for help fills you with dread. A dead network adapter usually means a surreptitious raid of the petty cash and a trip to the local computer-parts store for a no-name Ethernet card.

Then—as it always does—disaster strikes. Somewhere, something goes wrong. One thing that you can be sure of is that it will happen at the worst possible time. It is likely that a crucial presentation will be under way, or perhaps a software release is due by close of business. Maybe you are hosting a server for a client, and the client has noticed its absence, and is on the phone, using words like "unscheduled outage" and "penalty clause." If your clients are so inclined, words like "kneecap" and "sledgehammer" might also be heard. Another fact you can be reasonably sure of is that the sysadmin will not be present, and the next-most technical person will be called upon to work up a minor miracle to fix the ailing network.

Sound far-fetched? Believe it or not, I have been in this situation more than once. What follows are some hints that may help in fixing networks in suboptimal conditions, and as always, with the understanding that it must be done as cheaply as possible.

Many of the hints use features found on Linux boxes, beloved for its technical excellence (and its low cost). Most of the tips here can be adapted for whatever type of operating system you have.

Audible Ping

Ping is the venerable tool that we all know and love, and is the reigning king of the low-tech diagnostic tools. Linux (and other operating systems that use GNU tools) features an extension to *ping* that produces a beep on receipt of a response. The *audible ping* is designated by the **`-a`** command-line option.

Something as simple as **`ping -a missinghost.your.net`**, left running from a console in the server room, can alert you when you have finally reestablished network connectivity. It is like having a cable tester that can traverse routers.

Where Are You?

In a server room full of unlabeled generic boxes, it can sometimes be tricky to know which box is which. The following conversation is typical:

Hapless1: "Okay, I've logged into **`srv7`** by SSH [*Secure Shell Protocol*], and I think its second hard drive has died. Can you turn off its power switch when I shut it down?"

Hapless2: "Sure, which box is it?

Hapless1: "Ummm... its hostname is **`srv7`**..."

Hapless2: "None of them are labeled!"

Hapless1: "Okay... [`cat /proc/cpuinfo`] it's a Pentium 2."

Hapless2: "That narrows it down to five boxes..."

This kind of guessing game can continue for quite some time. Following the ground-breaking research of Murphy, if you guess wrong, it is reasonably certain that you will pick a critical server to drop. My least-favourite twist on this is when the boxes have been labeled—but labeled wrong—or labeled with yellow post-it notes (which fall off as the temperature in the server room increases).

If you are using a Linux box, and it has a CD-ROM drive, why not try ejecting it? Using the `eject /dev/cdrom` (or other device name as appropriate) command will make the box spit out its CD tray. It is like telling the real `srv7` to put its hand up.

[Cautionary note: Be careful of doing this to machines where the CD-ROM tray is behind a closed door, such as with the Digital Prioris or the IBM NetVista. Like a tractor-pull for plastic components, you *will* find out whether the server door is stronger than the internal tray mechanism of the CD-ROM drive.]

[Disappointing note: Calling eject on a nonremovable drive does not cause the hard drive to eject its platters. Bummer! A hard drive that could unleash a couple of platters at 10,000 revolutions per minute would be an interesting sight.]

Change Default Passwords (and record them for your successor)

Sometimes in one of these computer ghettos, you will stumble across an unexpectedly nice piece of equipment, such as a managed switch or a decent router. The chances are strong that it will have been left in its default configuration, so that any devious member of staff can *telnet* to it, change its configuration, leaving the network even more fouled up.

Your natural inclination should be to change these passwords—even if people do not act maliciously, they can sometimes foul up equipment accidentally. However, because you have been pressed into service as the network admin, remember that the same fate will likely befall another hapless victim one day. As a mark of consideration, record the equipment description, location, serial number, and new password, on paper. If the company has a safe, store it there. If the company has a safety deposit box, store it there. Make sure someone (a manager or director) knows about it. The time you save may be your own.

Do-It-Yourself Router

Perhaps you have identified that the network really ought to be split up—maybe moving testing to its own segment so that the incessant load-testing does not choke the network for everyone. However, requests for budget allocation to buy a router might not actually be fulfilled. It is at times like this that an old Pentium, two network cards, and a copy of the *Linux Router Project* (LRP) can be pressed into service as a cheap router.

The throughput of such a lo-fi router may not match that of a dedicated unit, but it may suffice for a small organization.

For bonus points, you might also consider setting up some firewall rules on the router, so that the next virus-ridden e-mail opened by someone in marketing does not flood the entire network with excess traffic.

Nagios

Network monitoring tools can make a world of difference to your quality of life as a temporary network administrator. Rather than waiting for users to alert you to a downed Internet connection, you can detect and repair problems as they occur. The ability to maintain logs of link downtime can also help support arguments to replace unreliable links.

Nagios[1] is a free network monitoring tool. It provides services such as:

- Monitor if a host is up
- Monitor if key services on a host are up
- Monitor if a host is running services it should not

A Web interface allows easy access to status reports. It can be configured to notify you when problems occur, for example, with an e-mail message. Of course, if the mail server is down, this notification method might not be so useful. Such a situation might be better handled by using the Nagios *Short Message Service* (SMS) messaging component.

Given that you might not have a dedicated *Global System for Mobile Communications* (GSM) modem available for sending these SMS notifications, you might like to investigate the Gnokii project[2]. Ostensibly a project to assist the user in communicating with a mobile phone handset (over data-link cable or infrared), with a capable handset users can initiate sending SMS messages from their handset with Gnokii.

Snort

Intrusion detection might seem a luxury on a network that is struggling to stay operational, but when the price is right (free) and you can spare time to set it up, *Snort* offers a range of features that is surprisingly good. *Snort* can even run without an IP address, making its host computer a fairly difficult target for intruders. The documentation at the *Snort* Website[3] is quite comprehensive, and I recommend it.

Squid

Squid[4] is a popular, free HTTP and FTP proxy server. The simple act of caching banner and button graphics for frequently accessed sites can give an apparent increase in Internet bandwidth. The impression for the end user is that things just get faster, because all those pretty graphics load immediately. You may know it is just a nifty trick, but why let on?

Nmap

One characteristic of chaotic networks is that, like weeds after heavy rain, network services spring up everywhere. Programmers are prime offenders in this respect. But be wary—a service with a security flaw, running on an exposed server, can provide an easy beachhead for crackers (a lesson I learned the hard way).

Nmap[6] is a free network scanner that can assist in finding servers that seem to be running more services than they ought to. It operates in several modes, and offers a range of switches to control its operation.

One of the features that seems more oriented toward people who are scanning networks they are not supposed to is the "Timing policy," specified with the **-T** command-line switch. The options offered here are *Paranoid, Sneaky, Polite, Normal, Aggressive, and Insane.* This feature actually comes in handy if the target of your attentions is heavily laden, or lives at the end of a slow link. If you are in the process of tuning a firewall to detect port scans, *Nmap* offers an excellent test facility too.

Another feature that will likely be helpful is the *Nmap* OS fingerprinting facility. Using a combination of techniques[5], it produces remarkably accurate results for most scans. Combine this result with a port scan and you can build a great picture of which machine has grabbed the wrong IP address (a favorite trick of laptop users: "I didn't know what my IP address was supposed to be, so I picked one.") You also can form a rough network map by OS-fingerprinting every active host on your network.

Immunization

It is a good idea to stay up-to-date on your tetanus shots because occasionally you will nick your hands on the sharp bits of metal found in computer equipment.

Traceroute

When licenses for your VisualRouteAnalyser2000 and TrafficGraphic tools have expired, remember that *traceroute* can be one of the most valuable tools to ascertain exactly where things are going wrong. The only (obvious) word of caution is to be aware that overzealous firewall rules can produce spurious results from *traceroute*.

Tag Cables

The desirability of labeling cables is so obvious that it seems silly to even mention it, but it might not have been standard practice for the sysadmin before you. All the more reason you should do the right thing. Sure, *you* know that the purple cable is the link from **gw-eng** to **gw-test**, but will the next person who has to diagnose network issues?

The other impediment to labeling cables is that the sheer volume of unmarked cables makes the task seem futile. Why bother labeling the new one you have just put in, when there are another 40 unknowns? Take heart—by gradually labeling a few here and there, the cables will gradually get less scary each time. Sometimes it can seem like the labor of Sisyphus, but every little bit helps.

Label Equipment

Post-it notes do not constitute an adequate label for network equipment or servers. You are strongly urged to preserve the sanity of other sysadmins by clearly labeling all equipment, using adhesive labels (in a pinch, the labels for a floppy disk will do).

At a minimum I would suggest that host name and operating system (where appropriate), IP address, and a dire warning against tampering with the unit be included. Bonus points are awarded to people who also maintain an equipment audit and record the details of the unit, plus a list of known services that it is running. Of course these will quickly become outdated, but with a known starting point confusion may be reduced.

Destroy Faulty Cables

After several hours of cable tracing, network-card replacement, checking switch link lights, and so on, it may be that you identify a network problem as being caused by a faulty network cable. It can happen anywhere, and is not necessarily a reflection on the skills of the [acting] sysadmin. (Although if the network cable has clearly been mangled and you should have spotted it with a quick visual inspection, you will probably feel a little silly if the time to locate the fault exceeded two hours).

So you whip a replacement cable out of your secret stash (you should have a secret stash of known-good cables) and voila! Network outage fixed. Now comes the most important duty of all—do not discard the damaged cable anywhere that subsequent admins might find it. On several occasions, damaged cables have been put back in operation, only to cause a repeat of the problem that caused them to be removed from service in the first place. It is not uncommon in server rooms to have an empty box that serves as a rubbish bin, but those unfortunates who come after you may not recognize its role as a waste repository in a time of crisis.

If waste is so abhorred that discarding cables is frowned upon, perhaps you can redo the ends of the cable and vigourously retest. Some even maintain that a long cable run can be split into several shorter runs and reused, because the cable fault is likely to be caused by a single break. I disagree—any cable that has broken in one place is likely to suffer further breaks. Demonstrating this principle to overly frugal managers is sometimes best achieved by ensuring the outcome of the demonstration. I suggest laying the cable through a close-fitting door frame and slamming the door on it a few times prior to testing.

Help Dying Equipment on Its Way

Sometimes it can be difficult to discard equipment. Combine this with the almost pathological frugality common in the small business owner, and you find the most decrepit network gear being nursed along. “I just know this old hub has another few years in it. Sure, a few of the Ethernet ports are stuffed, it overheats on warm days, and looks like it might have a mouse nest in the power supply, but that is no reason to discard it.” Nothing is going to convince the owner of this piece of gear that it is time to “redeploy” it in the rubbish bin.

Sometimes you have to be cruel to be kind. Without wanting to seem too much like the *Bastard Operator from Hell* (BOFH)[7], you may have to help some of this equipment meet its end. It is difficult to identify any one method that fulfills this requirement. My best suggestion is to avoid solutions that leave any externally visible marks (unless they are carbonization marks caused by electrical fault).

You may find that some equipment shows a perverse ability to survive conditions well outside their "recommended operating environment," and nothing short of a sledgehammer will cause those last two operational ports to die. My recommendation here is to do some network reorganization so that the people responsible for the retention of the equipment are directly affected by it. Nothing says "replace me" quite like frequent trips to the server room to toggle the power switch on an ailing hub. It is surprising how fast requisition orders get signed when managers can no longer browse their favorite Websites.

Conclusion

The crisis has passed. Your time as a sysadmin has passed, and you are free to return to your real job. You have acquitted yourself admirably as sysadmin, and you have learned something in the process.

Like the end of a horror movie, you know that it does not really end here. Somewhere, something is waiting to go wrong. Will you be ready the next time?

References

[1] Nagios: **http://www.nagios.org/**

[2] Gnokii project: **http://gnokii.org/**

[3] Snort: **http://www.snort.org**/

[4] Squid: **http://www.squid-cache.org/**

[5] **http://www.insecure.org/nmap/nmap-fingerprinting-article.html**

[6] Nmap: **http://www.insecure.org/nmap/**

[7] BOFH: **http://bofh.ntk.net/Bastard.html**

LOCUM SYADMIN is the nom de guerre of a roving programmer who often seems to find himself in sysdamin roles. Operating in deep secrecy, this elusive creature may sometimes be seen tracing cables and cursing. E-mail: **locum_sysad@yahoo.com**

Letters to the Editor

Ole,

I just finished reading the article about Secure BGP [*Border Gateway Protocol*] by Stephen T Kent. It was very informative and educational with regard to the application and overhead of using the additional BGP attributes and IPSec [*IP Security*]. However, it should be noted that the reliance of a PKI [*Public Key Infrastructure*]-based system, although strong, may also present another possible exploit. If the PKI KDS (*Key Distribution System*) is attacked and subsequently knocked out, including redundant *Key Distribution Engine* (KDE) servers, this may cause serious ramifications to the operation of *Secure BGP* [S-BGP].

Here is a very informative link regarding S-BGP resources for your readers: **http://www.ir.bbn.com/projects/s-bgp**

Also, did you know that the *North American Operators' Group* (NANOG) in conjunction with Cisco engineers recently conducted a BGP vulnerability test? This test confirms that BGP implemented properly is pretty secure in and of itself, without the need for something like S-BGP. The article, titled "BGP Vulnerability Testing: Separating Fact from FUD," was written by Sean Convery and Matthew Franz, Cisco Systems. The article can provide a contrast to the one submitted by Kent and give the technical community both sides of the BGP security issues. Following is the link:
http://www.nanog.org/mtg-0306/pdf/franz.pdf

I thoroughly enjoy IPJ and look forward to each issue. Keep up the great work.

—*Jeffrey J. Sicuranza, Applied Methodologies Inc.*
jsicuran@optonline.net

The author responds:

Ole,

Jeffrey makes a few observations about S-BGP in his letter, and they merit responses.

First, I would hope that the discussion of the security features of S-BGP and their direct derivation from the semantics of BGP was as informative as the discussion of performance aspects of the system. After all, a system with good performance but questionable security is probably a poor candidate to S-BGP routing.

Jeffrey raises the question of whether the reliance of S-BGP on certificates, CRLs [*Certificate Revocation Lists*], and address attestations creates significant vulnerabilities that need to be addressed. This is a fair question, but one which I think we have addressed.

The data that S-BGP stores in repositories is data that changes slowly, and thus the system tolerates unavailability of these repositories fairly well. Note that no router ever accesses these repositories in order to verify a route attestation received in an UPDATE. Instead, each ISP [*Internet Service Provider*] or multihomed subscriber NOC [*Network Operations Center*] accesses the repositories to retrieve this data, process it, and distribute the extracted public keys and authorization data to the routers in its network. We anticipate that this process might occur roughly every 24 hours. Because the information represented by the signed objects in the repositories changes very slowly, this retrieval rate seems appropriate. One would expect that these repositories can be engineered to meet these availability requirements. In the worst case, network operators can choose to keep working with the last set of data that they have successfully retrieved. This works because operators process the data before distributing it to their network, and thus can override expired CRLs, etc. So, I think the answer to Jeffrey's cited concern is that S-BGP is not very vulnerable to attacks against these repositories.

I strongly disagree with the conclusions Jeffrey draws from the BGP vulnerability tests he cites. Numerous incidents of BGP security breaches have been reported over the last few years, so there is no question that BGP, as implemented, deployed, and operated, is insecure. Correct implementation of BGP and improved network operator management practices certainly can reduce BGP vulnerabilities. However, the article in question is hardly a refutation of the wide range of vulnerabilities that exist both in practice and in principle. Much of it focuses on a narrow range of attacks, not broader security concerns.

In addressing broader security concerns, for example, the article argues that proper filtering of routes will mitigate the impact of a compromised router. But we know that such filtering is not feasible for many transit network connections, and route filterers are prone to configuration errors. Reliance on transitive trust (for example, assuming that peers filter routes appropriately) makes BGP intrinsically insecure. Relying on *all* ISP operators to *never* make exploitable errors in configuring their route filters, where such filters can be used, is a fundamentally flawed security approach. S-BGP accounts for the reality that not every ISP will operate its network perfectly, and employs mechanisms to allow other ISPs to detect and reject a wide class of errors (or attacks) that may result from such imperfect operation. Thus I reassert that the security vulnerability characterizations that appear in the S-BGP publications are accurate, not overblown.

As a side note, I find it odd that some critics of S-BGP argue that it fails to account for operational reality, yet they offer alternatives that are based on unrealistic assumptions about network operators acting perfectly!

—Steve Kent, BBN Technologies
kent@bbn.com

Book Review

IP for 3G

IP for 3G, Networking Technologies for Mobile Communications, by David Wisely, Philip Eardley, and Louise Burness, ISBN 0-471-48697-3, John Wiley & Sons, 2002.

I was looking for a book covering mobile communication issues from an IP perspective and IP issues from a mobile communications perspective in order to better clarify details of IP and *third-generation* (3G) convergence. The issue is becoming more and more concrete with the early implementations of 3G networks, so this is a timely book for networking professionals.

Organization

This well-organized textbook helps readers easily understand the "IP-for-3G" issues. It gives a clear vision of that convergence as well as the current snapshot of the recent developments about the subject within the research community. The book is more than an introductory textbook; but readers interested in more technical elaboration can refer to a detailed list of references and further readings given at the end of each chapter.

The book begins with a short chapter that explains the case for IP for 3G. The authors discuss in detail what the term means. They give possible interpretations of IP (Internet, IP Protocol, applications) and their consequent implications on the meaning of IP for 3G. Then they elaborate the IP case within first the "Engineering Reasons for IP for 3G" and then "Economic reasons for IP for 3G" sections.

The second chapter is an introduction to 3G networks. The chapter mostly concerns the core and the access part of 3G networks, skipping the air interface part, because core and access are where IP would make a real difference to the performance and architecture of a 3G network. The chapter reviews briefly the history of 3G developments, from conception to implementation. Then the architecture of *Universal Mobile Telecommunications Service* (UMTS) is introduced, followed by the section where elements of the core network and the architecture of the radio access part are examined. For each part, main functional components such as *Quality of Service* (QoS), mobility management, security, transport, and network management are discussed in detail.

The third chapter discusses the basics of IP and IP networks. Authors give excellent remarks about IP design principles, which are then compared to those of classical telecommunications. Subsequent short sections inform readers about IP addressing schemes, routing, layer behavior, etc. The final section covers the issue of application layer security, which is irrelevant to me for the content of this book. A note: Some of the following chapters require better IP know-how, especially about domain segmentation and intra- and interdomain routing issues. Readers with no prior information are encouraged to refer to other materials before examining the details of, for example, mobility management and QoS.

The fourth chapter is about the multimedia support and session management. First, the concept of session management is introduced. The chapter focuses mainly on the control plane functions of the session management, and the data plane functions are covered in detail in the sixth chapter. The concept of the *Virtual Home Environment* (VHE) is introduced, which forms one of the major requirements of the next-generation mobile system. The authors then review control plane session management protocols, namely H.323 and the *Session Initiation Protocol* (SIP). More discussion is given to SIP, because it is included in the next generation of UMTS standards as the major session management protocol.

The fifth chapter reviews a major problem of the IP-for-3G concept: mobility management. Other key issues of IP such as QoS, IPv6, and session management have always been subject to preceding studies, because those protocols have already been proposed for use in stationary networks. However, the issue of mobility management is a major subject to be investigated for any proper convergence scenario. Personally, I find that this is the biggest challenge of the "long-time-discussed" convergence of IP and mobile communications, and hard work is still ongoing in order to properly resolve the mobility problem. The chapter reviews the basics of mobility such as personal or terminal mobility. From there, macromobility (interdomain or global mobility) and micromobility (intradomain or local mobility) concepts are discussed, followed by proposed protocols for each type of mobility. Mobile IP is examined as the (unique) macromobility protocol. More attention is given to micromobility because it is the most sensitive part of the mobility, under the assumption that 3G BTSs (B nodes) will be simple routers with some extra capabilities. Two variants are discussed, mobile IP schemes, which are based on dynamic tunneling mechanisms, and "per-host forwarding" schemes based on dynamic routing functions. A comparison of major proposals for micromobility management protocols follows.

The sixth chapter considers current IP QoS mechanisms, their operation and capabilities. Those mechanisms created mostly for stationary IP networks may provide a bounded QoS for some "non-real-time" applications, but they are not enough to support any QoS request within the wireless or mobile environment. After giving details of current QoS mechanisms and discussing wireless implications for TCP QoS as well as mobility and wireless issues for *Real Time Protocol* (RTP) QoS, the chapter examines the key elements of QoS and generic features that any prospective QoS mechanism must have. Finally, the authors analyze recent Internet QoS mechanisms such as *Integrated Services* (IntServ), *Differentiated Services* (DiffServ), *Multiprotocol Label Switching* (MPLS), and *Resource Reservation Protocol* (RSVP). The closing section proposes a possible outline solution for how to provide IP QoS for 3G, based on previous work done during the EU BRAIN project.

In the final chapter, the authors summarize all previously given subjects to sketch out the vision of an "All-IP" mobile network. Principles, architecture, routing and mobility issues, QoS, security issues, and interfaces are all discussed to elaborate the generic vision of All-IP networks. Finally, 3G network evolution covering UMTS R4 and R5, and what is beyond 3G, are all discussed.

The book is perfect in the sense that it touches a very hot topic, most of the technical details of which are still in the process of evolving. The authors manage very well the level of details about each subject; they first discuss the overall material before examining details, so readers can obtain a generic but complete view before studying technical details. Each chapter is followed by a comprehensive list of references and further readings, each of them classified by topic. The only fault I find in the book is that SIP should be discussed in more detail.

Recommended

Overall, I would highly recommend this book to any network professional, especially one who is part of any IP-3G convergence process for mobile operators. Still, data network professionals can glean much from the book, because the aim is to carry—a little differently—the same old data, whether or not it contains multimedia, voice, or standard data information.

—Dr. K. Murat Eksioglu, RT.NET, Turkey
murat.eksioglu@o2.net.tr

[Ed.: A version of this review was previously published in the October 2003 issue of *IEEE Communications Magazine* (Vol. 41, No. 10). Used with permission.]

Fragments

Tim Berners-Lee Knighted by Queen Elizabeth

31 December 2003 — Tim Berners-Lee, the inventor of the World Wide Web and director of the *World Wide Web Consortium* (W3C), will be made a *Knight Commander, Order of the British Empire* (KBE) by Queen Elizabeth. This was announced earlier today by Buckingham Palace as part of the 2004 New Year's Honours list.

The rank of Knight Commander is the second most senior rank of the Order of the British Empire, one of the Orders of Chivalry awarded. Berners-Lee, 48, a British citizen who lives in the United States, is being knighted in recognition of his "services to the global development of the Internet" through the invention of the World Wide Web.

"This is an honor which applies to the whole Web development community, and to the inventors and developers of the Internet, whose work made the Web possible," stated Berners-Lee. "I accept this as an endorsement of the spirit of the Web; of building it in a decentralized way; of making best efforts to keep it open and fair; and of ensuring its fundamental technologies are available to all for broad use and innovation, and without having to pay licensing fees."

"By recognizing the Web in such a significant way, it also makes clear the responsibility its creators and users share," he continued. "Information technology changes the world, and as a result, its practitioners cannot be disconnected from its technical and societal impacts. Rather, we share a responsibility to make this work for the common good, and to take into account the diverse populations it serves." For more information see:

http://www.w3c.org/2003/12/timbl_knighted

SECSAC Publishes DNS Report

The *Security and Stability Advisory Committee* (SECSAC) has published a report entitled "DNS Infrastructure Recommendation." For details see:

http://www.icann.org/committees/security/dns-recommendation-01nov03.htm

Coordination, not Governance says ISOC re WSIS

The *Internet Society* (ISOC) published the following text at the *World Summit on the Information Society* (WSIS 2003) which was held in Geneva in early December, 2003:

ISOC is a global not-for-profit membership organisation founded in 1991 to provide leadership in Internet-related standards, education, and policy issues. We are dedicated to ensuring the open development, evolution and use of the Internet for the benefit of people throughout the world. Our education initiatives, for example, have helped bring Internet connectivity to virtually all developing countries over the last 12 years.

ISOC is the organisational home of the *Internet Engineering Task Force* (IETF)—an open consensus-based group responsible for defining Internet protocols and standards. Through our participation in WSIS 2003 we aim to increase understanding and awareness of what is important in order to develop and maintain the Internet's stability, open nature and global reach.

The Internet has come of Age

In many countries, the Internet has become a mass medium. This has brought with it reflexive pressure on policy makers to regulate it as if it were radio, television, or other mass media. While Governments naturally seek to address their citizens' interests regarding online privacy, spam, Internet security, intellectual property protection, the price of Internet access, and the digital divide, our position is that better use of technology, and broad participation in today's Internet coordination processes, not Government regulation, are the most effective and appropriate ways to satisfy these concerns.

The biggest barrier to the Internet fulfilling its immense potential could turn out to be misinformed and inappropriate intervention in the way in which the Internet's technologies, resources and policies are developed, deployed and coordinated. The Internet Society can help provide guidance here.

What is the nature of the Internet?

The Internet is a modern distributed communications medium. No one is in charge of the Internet and yet everyone is in charge. Unlike the antiquated system of national telephone network monopolies, the global Internet consists of tens of thousands of interconnected networks run by Internet Service Providers, individual companies, universities, Governments, and other institutions. Some of these are global in scope, others regional or local. Hundreds of different organisations and thousands of different companies make decisions every year that contribute to how the Internet develops.

These varied entities, together with the users of the Internet and the developers of Internet technologies and applications, have specific needs for coordination. Collaborative processes that are critical for the future stability and evolution of the Internet, and which should not be modified arbitrarily or abruptly, satisfy these needs.

Coordination, not Governance

It is misleading to use the term "Internet Governance" when the Internet is clearly not a single entity to govern. It is more useful to refer to "Internet Coordination." The multiple facets of the Internet require different types of coordination, each calling for specific competencies and sensitivities to balance the needs of the Internet user community globally and locally. Specific Internet Coordination activities are taking place globally at three levels:

- Coordination of the definition of Internet standards
- Coordination of the availability and assignment of Internet resources
- Coordination of the policies preventing misuse of the Internet

This coordination is best performed by the existing set of organisations using proven processes. Because of the diverse nature of these activities, it is unrealistic to expect a single body— Government or otherwise—to take on all these roles effectively.

Coordinating Internet standards

The IETF under the umbrella of the Internet Society, is one of the oldest and most successful Internet coordination processes. Other organisations are also involved in Internet-related standards, including the IEEE, the W3C and the ITU.

Many of the protocols at the heart of today's Internet (for example, TCP, IP, HTTP, FTP, SMTP, Telnet, PPP, POP3, the DNS protocol etc.) were developed through IETF standards activities. The results of the IETF are well engineered and practical open protocol standards that are trusted and open to global implementation with little or no licensing restrictions—they are freely available on the Internet, without cost, to everyone.

The strength of the IETF process lies in its unique culture and talented global community of network designers, network operators, service providers, equipment vendors, and researchers. They all openly contribute their individual technical experience and engineering wisdom in an environment that fosters innovation and the open exchange of ideas. This process, which is open to anyone, helps quickly identify and articulate problems of common interest. It also helps build the trust required to make the further investments necessary for a protocol to be usefully implemented and deployed. Ultimately, however, it is the Internet users themselves that determine whether or not a protocol is valuable and useful enough for widespread use. Here the IETF track record of producing useful, widely deployed protocols is unrivaled.

Coordinating Internet resources: The Internet Registry System

There has always been a need to manage the allocation of Internet resources such as the unique addresses that identify devices connected to the Internet (IP addresses), generic top-level domain names (for example, **`.org`**), country code top-level domain names (for example, **`.ch`**), domain names (such as **`www.isoc.org`**), and the systems that translate domain names into IP addresses (for example, the *Domain Name System* or DNS).

This coordination activity has been handled by long-standing, not-for-profit membership organisations such as the *Regional Internet Registries* (RIRs) and *top-level domain* (TLD) registries.

More recently, coordination at a global level has been supported by the *Internet Corporation for Assigned Names and Numbers* (ICANN). Established in 1998, ICANN is also a not-for-profit organisation. Business, technical, non-commercial, academic, governmental and end-user communities participate in ICANN.

These organisations are a meeting point for bottom-up, consensual, industrial self-regulation by the groups and individuals that use their services and resources.

Coordinating policies preventing misuse of the Internet

As we have seen, organisations such as the RIRs, TLD registries, ICANN and the IETF all have very specific roles. It is neither within their charters, nor within their capabilities, to take on responsibility for all areas of Internet Coordination—particularly that of preventing inappropriate use of the Internet. For example, areas such as "cyber crime" (for example, fraud and child pornography) require coordinated global attention by lawmakers—and not by those responsible for the equitable coordination of the underlying Internet infrastructure. Security matters also need to be addressed by organisations providing Internet access (not only by standards developers), and intellectual property iss ues may best be handled by organisations such as the *World Intellectual Property Organization* (WIPO).

In discussions about these broader Internet policy issues there is cooperation between all the organisations mentioned above. ICANN for example works with WIPO to implement its *Uniform Domain Name Dispute Resolution Policy* (UDRP). And the Internet Society, with technical advice from the IETF, works with Governments and policy makers to explain the effects and possibilities of new Internet technologies.

The way forward: Make your voice heard

Existing consensus-based processes have given us the Internet and have successfully coordinated its phenomenal growth: thousands of new networks, new policy procedures, new top-level domain names, new protocols etc. All of them constantly balance the needs and stability of today's Internet with future demands.

An open debate is now needed to move towards common, globally acceptable policies, processes and technologies to prevent misuse of the Internet. Governments have a vital role to play here as a concerted effort on the part of the Internet community, non-governmental organisations and Governments can help strengthen and extend today's successful coordination processes.

The successful continued development of the Internet for the benefit of everyone can be ensured by participation in these proven processes rather than by attempting to create new untested mechanisms that are inappropriate to the unique characteristics of the Internet.

The Internet Society remains dedicated to providing information and orientation about Internet structures and processes. We encourage broad participation in the activities of each of the organisations involved in Internet coordination. For more information on ISOC, visit: **www.isoc.org**

The Internet Protocol Journal is published quarterly by the Chief Technology Office, Cisco Systems, Inc.
www.cisco.com
Tel: +1 408 526-4000
E-mail: ipj@cisco.com

The Internet Protocol Journal, Cisco Systems
170 West Tasman Drive, M/S SJ-7/3
San Jose, CA 95134-1706
USA

ADDRESS SERVICE REQUESTED

PRSRT STD
U.S. Postage
PAID
Cisco Systems, Inc.