The ISP Column
A monthly column on things Internet

September 2010

Geoff Huston

A Rough Guide to Address Exhaustion
(in 12 easy questions)

It seems that there is an increasing level of interest in the topic of IPv4 address exhaustion, so I thought this month I'll share a set of answers to the most common questions I've been asked on this topic in recent times.

What's the most significant challenge to the Internet today?

What a wonderfully open-ended question! There are so many challenges that one could identify: improving the level of security on the network, eradicating spam and viruses, improving capacity of the network's infrastructure, improving the efficiency of high speed data transfer, improving the accuracy of search engines, building more efficient and high capacity data centres, reducing the unit cost of Internet services, to name but a few.

If there is a common factor in many of these challenges, its scaling the network to meet an ever expanding agenda of more users, more devices, more traffic, more services and more policies. And with more users and more forms of use comes higher levels of diversity of use and greater need to replace implicit mechanisms of trust with explicit forms of trust negotiation and greater levels of demonstrable integrity of operation.

But these issues are all tactical in nature. They reflect the "how" of making the network work tomorrow by studying how to undertake marginal improvements on the network of today. However, its not clear that the networks not just of tomorrow or next year, but a decade or more hence should reflect the usage patterns and user population of today. Perhaps a more fundamental challenge is to understand what's missing in today's network that we will need in the future.

This leads to a pretty obvious challenge, at least for me. The basic currency of any network is identifiers. Identifiers allow the network to distinguish between clients and ensure that conversations occur between those parties how intended to communicate. In the world of packet switched networking, such as IP, these endpoint identifiers are synonymous with the concept of an "address". What's missing in today's network is an abundant supply of new addresses that will allow the network to scale up in size by a further factor of at least 1 million, and hopefully more than a billion-fold.

In fact the supply of addresses is not just inadequate for future needs for a decade hence. The stock of addresses is facing imminent depletion, and the question of availability of addresses is best phrased in terms of months rather than years.

> Perhaps the term "address" is somewhat of a misnomer in this context, but it may well be too late to change that now. The primary role of an "address" is not to uniquely identify the location of an end point of a network in relation to some positional or topographical

coordinate set, but to simply uniquely identify an end point to distinguish it from all other end points. Its location is not an intrinsic property of this so-called "address". But common convention is to call these end point identifiers "addresses", so I'll stick with the same convention here.

So my candidate "most significant challenge for the Internet today" is that we are running out of further supply of IP addresses.

What is an "IP Address" and why is it so important?

One of the revolutionary changes introduced by the so called "packet switched" network architecture of the Internet, as compared to its telephone predecessor that used "circuit switching" was that a massive amount of "intelligence" was ripped out of the network and placed into the devices that connect at the edge.

IP networks are incredibly simple, and at their most basic level they do very little. They are built of routers and interconnecting conduits. The function of a router is quite simple. As packets arrive at the router from the connected circuitry (or from a wireless interface) the packet is divided into a common IP header and a payload. The packet's IP header contains, among other components, two fixed length fields: the "address" of the packet's intended destination, and like a postal envelope, it also contains the "address" of the packet's creator, or the source. The router uses the packet's destination address to make a routing decision as to how to dispose of the packet. Most routers have a small amount of local knowledge - in the general case the router knows about a small set of addresses of directly connected computers, and it also knows the location of a connected router that lies on the "default" path to the rest of the network. For each incoming packet, the router inspects the destination address in the packet and either passes it to a connected computer if there is an address match, or otherwise passes it down the default path to the next router. And that's a working description of the entirety of today's Internet. The key aspect here is that every connected device must have a unique address. As long as this condition is satisfied, everything else can be made to work.

In the current version of the Internet protocol an "address" is a 32 bit field. This field can encompass some 4.4 billion unique values.

Why are we running out of addresses?

Blame silicon. Over the past 50 years silicon chip industry has graduated from the humble transistor of the 1950's to an astonishing industry in its own right, and the key to this silicon industry is volume.

Individual processor chips may take hundreds of millions of dollars to design but if fabricated in sufficient volume each processor chip may take as little as a few dollars to manufacture and distribute. The larger the production run of the silicon die, the lower the unit price of the resultant chip. These days we produce a huge volume of computers every year. In 2008 alone around 10 billion computer processors were manufactured. While most of these microprocessors are simple 8 bit processors that are used to open doors or run lifts, a sizeable proportion are used in devices that support communications, whether its in laptop computers, iPhones, or even more basic communication applications. Typically we don't invent a new communications protocol for each new application. We recycle. And these days if we want a communications protocol for a particular application its easiest to simply embed the IP protocol engine onto the chip. The protocol is cheap, well tested, and it works across almost any scale we can care to imagine from a couple of bits per second a couple of billion bits per second.

So its not just the entire human population of the planet who may well have a desire to access the Internet in the future, but equally importantly its the emerging world of "things" that

communicate. Whether its the latest fashion in mobile phones, or more mundane consumer electronics devices such as televisions or games consoles, all these devices want to communicate, and to communicate they need to have a unique identification code to present to the network, or, an "address".

These days we are turning on in excess of 200 million new internet services every year, and today we've used up most of the 4.4 billion addresses that are encompassed by the IP protocol

When will we run out?

As of September 2010 there are some 151 million addresses left in the general use pool of unallocated addresses that are managed by the central pool administration, the Internet Assigned Numbers Authority (IANA). The world's IP address consumption rate peaked earlier this year at a new all time high of an equivalent rate of 243 million addresses per year.

A detailed model of address consumption based on this recent address consumption data indicates that the IANA will exhaust its address pool in the first half of 2012, probably in May.

The five Regional Internet Registries (RIRs) will still have pools of addresses available for general use at that time, but from that point, as they further deplete their local pools, IANA will be unable to provide any further addresses to replenish them. At this stage the Asia Pacific Regional Registry, APNIC, is experiencing the highest level of demand in the world, and currently account for some two thirds of all address consumption. At the current level of demand APNIC would completely exhaust its address pools by the end of 2011, or at best in early 2012. While the current models show that the other regions would be able to manage available address pools for slightly longer, this does not take into account the multinational nature of many of the largest of the service providers, and it is at this stage not known how much address consumption pressure will shift outward from APNIC to the other RIRs once APNIC's available address pool is fully depleted. So it may well be that 2012 will see IPv4 addresses cease to be generally available in most parts of the world.

What's the plan?

This news of imminent exhaustion of the supply of addresses is not a surprise. While the exact date of predicted address exhaustion has varied over time, the prospect of address exhaustion was first raised in technical circles back in August 1990, and work has been undertaken since that time to understand what might be possible and how that could be achieved.

There was an intense burst of engineering activity in the 1990's that was intended to provide a solution for this forthcoming address. The major outcome of this effort was the specification of a successor IP protocol to that of IPv4, called IPv6.

Why IPv6 and not IPv5?

It would be reasonable to expect the successor protocol of IP version 4 to be called IP version 5, but as it turned out version 5 if the Internet Protocol family was already taken. In the late 1980's the Internet Protocol itself was the topic of a considerable level of research, as researchers experimented with different forms of network behaviour. Version 5 of the Internet Protocol was reserved for use with an experimental IP protocol, the Internet Stream Protocol, Version 2 (ST-II), written up as RFC 1190 in 1990. When it came time to assign a protocol number of the "next generation" of IPv4, the next available version number was 6, hence IPv6.

The outcome of this was a relatively conservative change to the IP protocol. The major shift was to enlarge the address fields from 32 bits to 128 bits in length. Other changes were made, that were though to be minor improvements at the time, although hindsight has managed to raise some doubts about that!

The design intent of IPv6 is a useable lifetime of more than 50 years, as compared with a "mainstream" deployment lifetime of IPv4 of 15 year, assuming that you are prepared to draw a line at around 1995 and claim that at that time the protocol moved from an interesting academic and research project to a mainstream pillar of the global communications industry.

That 50 years of useable life for IPv6 is admittedly very ambitious, as it is intended to encompass a growth of the ubiquity of silicon from the current industry volumes of hundreds of millions of new connected devices every year to a future level of activity that may encompass in the order of hundreds of billions to possibly some trillions of new connected devices every year.

So the technical plan to address the address exhaustion problem was to perform an upgrade of the Internet and convert the internet from IP version 4 to IP version 6.

Nothing else needs to be changed. This is not intended to be a radical or revolutionary change. The change from circuit switching to packet switching was a revolutionary change for both the communications industry itself, and for you and I as enthusiastic communicators. The change from IPv4 to IPv6 is intended to be a polar opposite, and at best it is intended to be a seamless and largely invisible transition. Email will still be email. The web should still look just as it always did, and anything that works on IPv4 is expected to work on IPv6. IPv6 is not in and of itself any faster, not any cheaper, nor even is it all that much better. The major change in IPv6 is that it supports a larger address field.

How many addresses are in IPv6?

In theory, there are 2 to the power 128 unique addresses in IPv6. That's a very large number. If each IPv6 address were a single gain of sand then the entire IPv6 address space would construct 300 million planets each the size of the earth!

But theory and practice align only in theory. In practice the IPv6 address plan creates a useable span of addresses that encompasses between 2 to the power 50 to 2 to the power 60 devices. While this is nowhere near 2 to the power 128, this is still a range of numbers that are between 1 million to 1 billion times the size of the IPv4 address space.

How do we transition to IPv6?

Unfortunately IPv6 is not "backward compatible" with IPv4. Backward compatibility would allow for a piecemeal transition, where IPv6 could be regarded as a fully functional substitute for IPv4, so that the existing network base would remain using IPv4 forever, while the most recent devices would could use IPv6 and all devices could communicate with each other. The lack of such backward compatibility implies that this is simply not possible. IPv4 and IPv6 are distinct and different communications protocols, in the same way that English and, say, German are distinct and different languages.

There have been attempts to design various forms of automated protocol translator units that can take an incoming IPv4 packet and emit a corresponding IPv6 packet in the same manner as a language interpreter. However, this approach also has some major limitations so its only useable in very limited contexts.

The implication of this lack of backward compatibility and inability to perform automated translation within the network is that if we want to preserve comprehensive any-to-any connectivity during the transition we have to equip each device that is performing a transition

with both protocol stacks, or, in effect, allow the device to become "bi-lingual," and conduct a conversation in either IPv4 or IPv6, as required. This has been termed a "dual stack" transition.

When my computer supports IPv6 can I return my IPv4 address?

Each device needs to maintain its capability to converse using IPv4 while there are still other devices out there that remain IPv4-only. So a device that becomes IPv6-capable cannot immediately free off its IPv4 address. It will need to keep this Ipv4 capability and operate in dual stack mode for as long as there are other devices and services out there that are only reachable using IPv4.

The implication of this constraint is that during this transition we will need to add dual-stack devices to the Internet and consume both IPv4 and IPv6 addresses during this transition.

So, no, you'll need to keep your IPv4 address for as long as there are folk out there wqith whom you want to communicate who have not also migrated to be a dual stack IPv4 and IPv6 capable entity.

What needs to be done to transition the network to IPv6?

What is encompassed in "transition"? Do all Internet Service Providers have to decide when and how to re-program their systems and re-configure their routers, switches and middleware? Will they need to replace all the customers' modems with ones that support IPv6? What's the agenda?

This level of uncertainty about the transition to IPv6 is evidently widespread in today's Internet. Most of the actors in the Internet are unsure about what needs to be done, from the largest of the service providers down to individual end users. Yes, on the face of it its a simple matter of reprogramming devices from being just IPv4-capable to being capable of supporting both Ipv4 and Ipv6, but its not quite so simple. Dual Stack operation is not easy nor will it just happen without any form of applied impetus. Imagine that this is a transition from everyone on the planet speaking Latin to each other to everyone speaking Esperanto. If this was a simple matter of everyone stopping using one language and being rebooted to use the other language, then imagine the plight of the first person to undertake this transition - from being connected and being able to communicate with everyone else using Latin, this first users would find themselves speaking exclusively Esperanto to ... nobody! They would in effect have been disconnected from the network.

So the transition is a little trickier than just turning a big switch from V4 to V6. Because this is a piecemeal and fragmentary transition, each device, each router, each firewall, each load server, and all those other components of the network service platform need to be programmed with an additional protocol, and become, in effect bilingual. And in this case there are no magic interpreters than can "translate" between V4 and V6. So it is only when then entire network is bilingual in a dual stack mode can we then turn off IPv4 and consider the transition to be complete.

For an extended period of time the Internet is going to have to operate as two Internets. We've never tried that before, at least not of a grand scale as this - its often been likened to replacing the jet engines of an airplane while the plane is in flight. Somehow we now not only have to sustain a growth rate of at least some 250 million new connections per year, but at the same time retrofit V6 to the existing installed base, in addition to continuing to support V4. The complexity of this operation is significant, and there is considerable confusion about what to do, when to do it, how much all this will cost and who will pay. So yes, we are all unsure about what needs to be done.

How long do we expect this dual stack transition to take?

If only we knew! The Internet today encompasses some 1.7 billion users, and there are hundreds of millions of devices out there that are configured to only talk IPv4. Some of these devices will surely die in the coming years, and others may be upgraded or re-programmed, but others will persist in operation for many years to come while continuing to speak only IPv4. Even looking at what is being sold today, while many general purpose computers (or at least their operating systems) are now configured to operate in dual stack mode, when you look at embedded devices such as DSL or cable modems, or firewalls, or a myriad of other devices that are integral to the operation of today's internet, many of these devices are still configured in firmware to operate exclusively using IPv4.

Some modelling of the transition process has projected an eighty year transition process. That is heading into the realms of the absurd given that our expectations for the operational lifespan of IPv6 has a lower bound of just 50 years. However, given the sheer scope of the conversion task and the current level of penetration of IPv6 to levels of between 2% to 5% of today's Internet, then an expectation that this transition could be substantially completed in as little as two years from today also strikes an unrealistic note, given that such a deadline implies a conversion rate of in excess of 1 million devices each and every day in that two year span.

So a more realistic assumption is that we will probably take around five years to complete this transition, and we will need to operate the Internet in dual stack mode with both IPv4 and IPv6 across this entire period.

But at the current level of growth of the Internet the IPv4 address pool cannot sustain a further five years of internet growth, at least not with the current amount of unallocated addresses remaining in the allocation pools. There are some 150 million addresses left in the common free pool operated by the IANA, and even if you add the address pools held by each of the Regional Internet Registries to the mix, the total is some 430 million addresses. The current address consumption rate is some 240 million addresses per year, so it looks like we have between 14 and 22 months before the IPv4 address pools are fully depleted. So the IPv4 address pool cannot withstand the pressures of a five year transition.

"Fully depleted" or even "run out" is perhaps not the most appropriate way to describe what will happen to IPv4 addresses in the coming months. Its probably more accurate to say "unobtainable at the current prices". When the current orderly process of allocation of IPv4 addresses comes to an end, that does not mean that IPv4 addresses will be completely unobtainable. In this world many things that are scarce are still obtainable - for a price. It's quite reasonable to anticipate that for as long as there is still a demand for IPv4 addresses there will be some form of "aftermarket" where addresses are traded for money. However, as with many markets, what is not possible to predict is the price for addresses that will be established by such a market-based address trading regime.

What about "address sharing" in IPv4? Why do we need IPv6 anyway, given that we could simple share addresses in IPv4?

Yes, of course address sharing is an option, and we have been doing it for many years already in IPv4. But is it a viable substitute for IPv6?

As part of the engineering effort to develop a successor protocol to IPv4 in the mid 1990's, the IETF published a novel approach of "address sharing" we call today "network address translation" or NAT. These days almost every DSL modem comes equipped with a NAT. Today a DSL provider will provide to their subscribers a single IPv4 address. At home I have a single IPv4 address to use, and probably so do you. But in my home I have about 20 connected devices of various sorts (I'm counting Tivos, game consoles, televisions, printers and such, as they are all in essence Internet connected devices, and I believe that I'm not unusual these days. All these devices "share" the single external DSL connection, so all of them "share" this single IPv4 address.

But address sharing has its limitations. When a single address is shared by a single household nothing unusual happens. But If I were to try to do the same address sharing trick of using a

single IP address to share across, say 2,000 customers, I'd cross over into a world of pain. Many applications today gain speed through parallelism, and they support parallelism through consuming port addresses. Each IP address can support the parallel operation of 65,535 sessions, using a 16 bit port identifier as the distinguishing identifier. But when address sharing comes into play these ports are shared across the number of devices that are all sharing this common address. When 2,000 customers are sharing a single address, and each customer has some 20 or so devices, then that's an average of 1.5 port addresses per device. Common applications that exploit parallels include such favorites as Gmail, Google Maps and iTunes. With a sufficiently constrained number of available ports to use these applications would cease to work. Indeed, many network applications would fail, and at a level of a single address shared across 2,000 households, I'd guess that up to half of these 2,000 customers would not have a working Internet at any single point in time.

Our experience suggests that address sharing only works up to a point, then it breaks everything badly. We are already address sharing at the level of sharing a single address per household, and households are these days buying more connected devices of various sorts, not less. So attempting to share that single address across more than one household is a at best a short term stop gap measure, and is not a sustainable option that is an alternative to IPv6.

So we need to transition to IPv6, and we need to do so within the parameters of an impossibly short time in which to do so.

This all sounds like a terrible problem. Was this whole global "experiment" with the Internet all one big mistake? Should we have looked elsewhere for a networking technology for computers back in the 1990's?

The IP address issue is, for me at any rate, a fascinating issue. At the time when researchers were working in the specifications for the Internet Protocol in the 1970's they decided to use fixed length 32 bit fields of the interface identifier addresses in the protocol. This was a radical decision at the time. Contemporary network protocols, such as DECnet Phase III used 16 bit address lengths, and 8 bit addresses were also very common at the time. After all, computers were so big and expensive who could possibly afford more than 256 unique devices in a single network? 8 bits for addresses was surely enough! Using 32 bits in the address field was not an easy decision to make, as there was the constant pressure to reduce the packet headers in order to leave more room for the data payload, so to reserve such a massive amount of space in the address fields of the protocol header to allow two 32 bit address fields was a very bold decision.

However, it was a decision that has proved to be very robust. TCP/IP has sustained the Internet from a mere handful of warehouse-sized computers running at mere kilobits per second to today, where there are probably in excess of 3 billion devices that connect to the internet in one way or another, at speeds that range from a few hundred bits per second to a massive 100Gbps - all talking one single protocol that was invented over 30 years ago. IP has been able to demonstrate a scale factor 1 billion! In my mind that demonstrates a level of engineering foresight that is truly phenomenal! So in some sense the underlying observation here is not that IPv4 is running out of addresses today, but that it has been able to get to today at all!

Given that IPv4 has been able to scale by a factor of 1 billion, then if we can make IPv6 scale by a further factor of 1 billion from today then we will have done well.

Disclaimer

The above views do not necessarily represent the views or positions of the Asia Pacific Network Information Centre, nor those of the Internet Society.

Author

Geoff Huston B.Sc., M.Sc., is the Chief Scientist at APNIC, the Regional Internet Registry serving the Asia Pacific region. He has been closely involved with the development of the Internet for many years, particularly within Australia, where he was responsible for the initial build of the Internet within the Australian academic and research sector. He is author of a number of Internet-related books, and was a member of the Internet Architecture Board from 1999 until 2005, and served on the Board of Trustees of the Internet Society from 1992 until 2001.

www.potaroo.net