>

Published on *NetworkWorld.com Community* (http://www.networkworld.com/community)

Techniques for Prolonging the Lifespan of IPv4

By *Scott Hogg*

Created *Nov 9 2011 - 5:05pm*

IPv6 proponents have long been predicting the death of IPv4 to get the industry to recognize the importance of IPv6 [1]. Although IPv4 address exhaustion has occurred many organizations are still uncertain about the next steps. It is clear that IPv4 is going to be with us for decades to come and there are strategies to prolong the lifespan of IPv4. Are these strategies worthwhile or are they distracting and confusing the industry from moving to IPv6?

For many years the networking industry has been in denial of IPv4 address depletion. It is clear that we are completely dependent on IP communications and IPv4 address depletion is imminent. If we understand the total number of things on the Earth that may be connected to the Internet we can estimate the number of IP addresses we need. For example, the news media has been running many stories lately on the fact that we are approaching 7 billion people [2]. If we only have 4.2 billion IPv4 addresses, then we can easily see that not everyone can have their very own public IPv4 address. The other statistic to keep an eye on is the worldwide literacy rate [3]. People who cannot read and write their native language will not easily be able to use the Internet. If you estimate that the world illiteracy rate is 15% that means that over 1 billion people will not be using the Internet anytime soon. I was also surprised to discover that the total number of mobile phones on the planet is approximately 5 billion [4]. The number of smartphones is rapidly increasing and each of those phones will need an IP address. Other statistics that give an idea of the world population that may use the Internet is the number of people who have access to save drinking water [5] and the number of people who have electricity to their homes. If you don't have drinking water or electricity you probably aren't interested in the Internet. Regardless, it is clear that the Internet-connected population is expanding and the number of devices that use IP is growing.

John Curran, President and CEO of ARIN [6], gave a presentation at the September Texas IPv6 Summit titled "Ready or not...IPv6 is here [7]". John Curran's presentation covered information about IPv4 address exhaustion and prescribed an action plan. The facts are that the IANA free pool was exhausted in February of 2011 and APNIC has run out of IPv4 addresses. RIPE may run out of IPv4 addresses in the next six months while ARIN has prolonged its date for IPv4 exhaustion due to extreme conservation of IPv4 addresses. LACNIC and AFRNIC will not run out of IPv4 addresses for a few years. It is clear that the world will run out of public IPv4 addresses, it is just a matter of exactly when that will happen.

The lack of public IPv4 addresses is starting to hamper innovation. If we had more IPv4 addresses, systems like sensor networks and smartgrid technologies would be growing faster. What if a company wanted to create a system that made cars function as mobile Wi-Fi hotspots with 4G uplinks? Besides the issues with distracted drivers, we simply do not have enough IPv4 addresses to make this possible. I have heard some people say that cloud computing is not taking off because of lack of public IP addresses. Some large networks are even running out of private IPv4 addresses. If there is a system that needs a large amount of IP addresses then IPv6 may be the only alternative.

We are in a very awkward time in the Internet's history because we have exhausted all the IPv4 addresses but we have not fully deployed IPv6 yet. The industry has basically waited until the last minute to migrate to IPv6. Therefore, we are all going to enjoy the last-minute pain of transitioning in a short period of time. Instead of being able to transition to IPv6 slowly and methodically we are going to do it quickly. Like Seinfeld in the episode "The Ex-Girlfriend [8]" the famous quote is "You should just do it like a Band-Aid. One motion! Right off! ..." It may sound painful to transition to IPv6 rapidly, but the longer we wait, the more pain we may experience.

Geoff Huston, Chief Scientist, Asia Pacific Network Information Centre (APNIC [9]), has been writing and presenting on some insightful thoughts on the transition of IP protocols. The March 2011 The Internet Protocol Journal [10] (Volume 14, Number 1 [11]) had an article on World IPv6 Day [12] and its contents are almost completely IPv6-related. Geoff Huston wrote two articles "Transitional Myths" and "Transitioning Protocols" about all the methods of transitioning from IPv4 to IPv6. Recently, Geoff Huston posted a similar article about the transition to IPv6 titled "IPv6 Transitional Uncertainties [13]". Geoff Huston wrote about a variety of these techniques in his The Internet Protocol Journal, Volume 13, No.2 [14] titled "NAT++: Address Sharing in IPv4". Geoff Huston also gave a presentation at NANOG53 [15] titled "Keynote: A Progress Report on IPv4 Address Exhaustion". These articles provide an objective look at where we are heading, the dangers of inaction, and the issues facing adoption of IPv6.

Even though there is a large portion of the Internet backbone that supports IPv6 and most of the end-user computers run dual-protocol-capable operating systems, there are few broadband ISPs providing native IPv6 services to their subscribers and there are few content providers offering their content over IPv6. It is apparent that we will have IPv4-only systems in our environments for decades to come. Due to the issues facing IPv6 adoption and our dependency on IPv4, the industry has come up with many gyrations to prolong the lifespan of IPv4 and slowly move to IPv6.

IPv4 Address Efficiency

One strategy of prolonging IPv4 is to continue to use IPv4 but increase the efficient use of those precious addresses. Many large ISPs are spending considerable time with IP address reclamation projects to find blocks of unused IPv4 address blocks and repurpose them in the network. Enterprises continue to break up their IPv4 blocks into smaller and smaller subnets and increase their use of NAT/PAT. Many organizations feel that "we have plenty of IPv4 addresses for our needs" and therefore, IPv6 holds no interest or provides them no benefit. The question is; do these organizations have enough IPv4 addresses to sustain their businesses for the next 20 years?

You could always just purchase/lease more IPv4 addresses from your service provider. You could purchase IPv4 addresses from an organization willing to sell some of theirs and perform an "address transfer". For example, ARIN provides a process [16] for performing an address transfer. ARIN also approves several organization to perform address transfers, also calls Specialized Transfer Listing Service [17] (STLS).

In the meantime there are plenty of people trying to predict the future. There was an interesting presentation given at NANOG53 [18] titled "Economics of IPv4 Address Markets on IPv6 Deployment", by Andrew Dul, Cascadeo Corp. This presentation gave several economic models as to the depletion of IPv4 addresses and the adoption of IPv6. This presentation mentions the Hotelling's Rule [19] which predicts the price and timing for the exhaustion of a finite resource like public IPv4 addresses. It is clear that over time the cost of running an IPv4 network will increase [20]. Putting forth effort to prolong the lifespan of IPv4 will add to these costs.

Organizations may also re-arrange their use of public IPv4 addresses. IPv4 address reclamation activities are just "rearranging deck chairs on the Titanic". Time would be better spent furthering the deployment of IPv6. However, the reality is that organizations will be maintaining IPv4 and IPv6 networks simultaneously for many years so they have no choice but to increase the efficiency of IPv4. Because we have waited until the last possible minute to deploy IPv6 we are going to come with all kinds of creative ways to avoid having to continue to use IPv4 with the least amount of effort.

Carrier Grade NAT (CGN), Large Scale NAT (LSN), NAT444

Another technique to prolong the lifespan of IPv4 is to perform multiple layers of Network Address Translation (NAT)/Port Address Translation (PAT). This is a technique [21] where there is one instance of NAT44 (translating an IPv4 address into another IPv4 address) is used by the broadband internet access subscriber CPE device at their location. The subscriber has private IPv4 [22] addresses inside their home and get a private IPv4 address for the external interface of their broadband access device. The service provider will use private IPv4 addresses in their core and use a CGN/LSN/NAT444 [23] device in their core to NAT/PAT many private-addressed customers to a smaller pool of public IPv4 addresses. This technique is sometimes referred to NAT444 because there are 2 levels of NAT/PAT being performed.

This method works, but there are issues with some applications. Depending on the distance between the subscriber's home and the location of the CGN/LSN device within the service provider's network this could increase latency for all IPv4 communications for that subscriber. The subscriber may experience connectivity problems if the performance of the CGN/LSN system is limited and cannot keep up with the total number of connections per second.

Service providers recognize that deploying CGN/LSN devices at several points in their network is easier than migrating their core networks to be dual-protocol enabled. However, there is capital costs and operational costs to deploying and maintaining CGN/LSN systems. The CGN/LSN system can be a single point of failure that can negatively impact the subscribers Internet experience. CGNs/LSNs further increase the amount of anonymity of attackers on the Internet and make forensics and reputation filtering [24] difficult, if not impossible. Many agree that CGN/LSN is not an optimal solution but service providers have been slow to deploy IPv6 so they are in a tough situation trying to preserve their IPv4 address allocations for years to come.

Protocol Translation (NAT-PT, NAT64/DNS64)

Another extreme technique is to continue to use IPv4 inside of your organizations and translate those packet's source/destination addresses to IPv6 addresses as those packets leave an enterprise destined for the Internet. You probably have enough RFC 1918 [25] private IPv4 address space to sustain your internal networks for years to come. From the Internet perspective it looks like your organization is progressive in their use of IPv6. However, the problem is that, currently, there is little Internet content that is IPv6-reachable.

I liken this technique to trying to preserve the longevity of your Betamax [26] video recorder. Even though the rest of the world has transitioned to using DVDs, BlueRay, and DVRs, you cling to your recorder to maximize your investment in that technology. However, someday, your recorder may break or you can't buy tapes anymore.

The reverse of this approach is the other extreme where an enterprise has fully transitioned to IPv6 internally and use their limited public IPv4 addresses on the outside of a protocol translator. This would require a system at the Internet perimeter to translate the internal IPv6 packets to external IPv4 packets for use on the Internet. It is not likely that an organization can easily fully deploy IPv6 within their enterprise today and disable all use of IPv4.

Protocol translation causes issues that are less than ideal and cause problems for many applications. Applications that have embedded IP addresses may not work as desired. Applications that use FTP, SNMP, SCTP, multicast, or fragmentation of packets can experience problems traversing a protocol translator. Stateless Network Address Translation - Protocol Translation (NAT-PT [27]) also causes problems for DNS and IPsec. These are some of the reasons why NAT-PT is now deprecated by RFC 4966 [28].

An organization could use NAT64/DNS64 [29] at their Internet perimeter. However, for the DNS64 proxy function to make this work all communications must start with a DNS query. That may not work for all applications inside an enterprise. Therefore, protocol translation is considered a last-resort IPv6 transition technique.

Dual Stack Lite (DS-Lite)

Dual Stack Lite (DS-Lite [30]) is yet another technique for prolonging the lifespan of IPv4. DS-Lite is an IPv6 transition mechanism that is implemented within the service provider's infrastructure. It is comprised of a centralized Address Family Transition Router (AFTR) element function and a Basic Bridging BroadBand (B4) element that is integrated into the subscriber's CPE device or as a software agent on their computer (cleaver names huh?). DS-Lite encapsulates the end-user's IPv4 packets inside IPv6 packets for transport across the service-provider's IPv6-enabled core network. DS-Lite is a tunneling technique and effectively reduces the IPv4 MTU by 4 bytes to 1460 bytes. DS-Lite is standardized with the IETF RFC 6333 [31] titled "Dual-Stack Lite Broadband Deployments Following IPv4 Exhaustion".

DS-Lite improves upon multiple layers of NAT because it uses on a single centralized NAT function within the AFTR. There is no NAT being performed by the subscriber CPE B4 element. That single AFTR NAT function is stateful and uniquely differentiates each client IPv4 session based on the unique global IPv6 addresses used for encapsulating the subscriber's IPv4 packets. In this way, each subscriber CPE devices can use the same overlapping IPv4 address space (192.0.0.0/29) but their packets are identified by the unique IPv6 addresses used for the encapsulation.

DS-Lite can be considered a late-stage IPv6 transition mechanism to help service providers continue to conserve IPv4 address space. One of the dependencies for a service provider to implement DS-Lite is that it requires the ISP core network to be IPv6-enabled. While some service providers have implemented IPv6 [32] in their core networks today, there are still many service providers that are still working on that deployment.

Address Plus Port (A+P)

Another technique to prolong the lifespan of IPv4 involves changing how we use 32-bit IPv4 addresses in combination with 16-bit port numbers. The technique called Address plus Port (A+P) borrows bits from the port number and uses them to augment the IPv4 address yielding more public IPv4 addresses. This is an address-sharing technique that reduces the port number range and allows multiple hosts to statefully share a single public IPv4 address. Each node is assigned the shared IPv4 address and a port-range that its applications may use. A+P is standardized in IETF RFC 6346 [33] titled "The Address plus Port (A+P) Approach to the IPv4 Address Shortage".

One of the advantages of this technique is that it avoids the problems of a centralized CGN/LSN solution in the service provider's environment. A+P also helps to maintain the end-to-end nature of TCP/IP and allows for forensics and tracebacks.

The downside is that applications may need to be re-written so that they use the restricted range of source TCP/UDP port numbers properly. There are several implementations of A+P available today (RFC 6346 Section 3.4 [34]). There needs to be a signaling mechanism to communicate to the end node which IPv4 address it should use and the port range it should use. One proposed port-range signaling technique is "dIVI-pd: Dual-Stateless IPv4/IPv6 Translation with Prefix Delegation [35]". The other problem is that A+P only works for applications that use TCP/UDP port numbers. For example, ICMP will not function properly because it does not use a port number to borrow bits from to use to uniquely distinguish the source address.

IPv4 Address Sharing

One idea that I have thought of before is the idea of performing even greater address sharing techniques. What if your device only had an IPv4 address when it had something to send to/from the Internet? This would be like having a very short-term IPv4 address lease. Other times when your device was "dormant" then it would release its IP address for use by another end-user device. Think of this as DHCP with a lease-time of 60ms. This reminds me of the video that Cisco created titled "IPv6 - Are you Ready? [36]"

You can think of this as an IPv4 address equivalent of a vacation time share. You only have an IPv4 address when you have something to send (like interesting traffic). This is the ultimate combination of time domain multiple-access, packet-based networking, and statistical multiplexing. This might work for mobile devices that do not need an IP address while its owner sleeps. This might be a workable solution so long as this temporary IPv4 address leasing is done quickly enough to not impact end-user performance. It could be too much for a system like this to work with Dynamic DNS [37] (DynDNS). Furthermore, this is not an ideal solution for a service that needs a FQDN that resolves to a fixed IP address that doesn't change over time.

Locator ID Separation Protocol (LISP)

As the Internet strives to drive greater efficiency of the use of IPv4 addresses the size of blocks of IPv4 addresses advertised to the Internet will get smaller. The problem with IPv4 that causes this issue is that IPv4 uses a single 32-bit namespace where the address represents the location of the node on the network and its point of attachment. This leads to deaggregation of the addressing space as the Internet becomes more densely populated. This fragmentation of the IPv4 address space will cause the Default Free Zone [38] (DFZ) Internet routing tables to dramatically grow in size. Today the Internet is closing in on 400,000 IPv4 prefixes, but over the next 10 years this could grow dramatically. In order to prolong the lifespan of IPv4 we need a technique that can help us with the scalability of the Internet routing tables.

One technique that can ease this growth is Locator/ID Separation Protocol [39] (LISP). LISP is a network architecture that splits the namespace into two sections: one used for the routing (Routing Locator (RLOC)) and one used for the end-node (Endpoint Identifier (EID)). This technique separates the topology location on the Internet from the identifier of the single node on the Internet, hence the name.

The other aspect of LISP is that it is a technique for mapping and encapsulating (map and encap) packets on the Internet. The RLOC is the IP address of the LISP router and the EID packets are encapsulated in the RLOC's packets as they are sent to the Internet. Therefore, you can think of LISP as an "over-the-top" tunneling method adding a 32-byte UDP port 4341 LISP header. There is also a mapping service that uses UDP 4342 packets, and similar to DNS, that will resolve the EIDs to locators defined in the mapping database.

One of the advantages to LISP is that it does not require any modifications to the end-systems. Only the routers are aware of these tunnels and the name mapping/lookups. As such, LISP is considered an "over the top" technology that can operate on an IPv4 or IPv6 network. The LISP IETF working group [40] has been working on this architecture for many years. There are many drafts of the standard and the various components of the protocol but there are no RFCs yet.

Another advantage for LISP is that it has broad vendor support. Cisco [41] has been supporting LISP extensively. Facebook [42] has been using LISP [43] and has used it for IPv6 [44]. LISP is implemented in FreeBSD with OpenLISP [45]. Other companies like Qualcomm, VeriSign, Microsoft, Verizon and Wells Fargo are experimenting with LISP. Furthermore, LISP is already running on the Internet and there is a functional LISP beta network [46].

Imagine if the effort that is being put into LISP were being put into helping the world migrate to IPv6. Would that work have resulted in a bigger long-term impact over simply development of a tunneling technique?

Host Identity Protocol (HIP)

Another "two-space" technique is the Host Identity Protocol [47] (HIP). This technique inserts a shim between the IP header and the transport header that contains the "Host Identifier" and "Location" of the end-system. HIP is an IETF experimental standard defined in RFC 4423 [48] "Host Identity Protocol (HIP) Architecture" and RFC 5201 [49] "Host Identity Protocol".

The down-side to this approach is that it requires changing the IP stack within the operating system and applications and changes to DNS to add new resource records. Applications need to be modified to use the HIP names space rather than IPv4 or IPv6 addresses. The other limitation is that these Host IDs are cryptographic public keys so we need a global PKI to make this functional on the Internet. The industry already had to endure a decade of adding IPv6 to operating systems so another stack modification seems daunting at best.

Conclusions

Regardless of whether these techniques are a good idea or not, it will take a lot of time before these solutions are viable. The CGN/LSN techniques required stable vendor products and service provider deployments. The IP stack shim techniques require software to be integrated into all the popular operating systems and network devices across many manufacturers in an interoperable way. Just look at how much time it has taken to get IPv6 capabilities into the broad range of systems available today. Even if someone came up with a great idea today it would take several years before it was available for ubiquitous deployment on the global Internet. If you think the Internet is in bad shape now, we just can't wait 5-10 years from now for the hot-new-idea to be ready for deployment.

Time is wasting while the industry works on techniques to prolong IPv4. While we argue and try to look at alternatives, time is passing by and IPv4 addresses are growing increasingly scarce. Any of the other techniques that we can come up with here at the last minute are just "too little, too late". Many people would like to "do the right thing" and help preserve the Internet for everyone to use. Some people want to create an economic advantage for themselves. Others are lazy and have not learned about IPv6 and these alternatives enough to help their organizations make an informed decision on the topic. Many people seem to be delaying the decision as they try to see how IPv4 address exhaustion applies to them or how they can make money in these market conditions. This seems to be distracting us from the only alternative that will result in a unified and interoperable Internet.

Fall is coming and the weather is changing. In Colorado, where I live, summer basically gives way directly to winter. A few weeks ago the temperature was 80(f)/27(c) and last week it snowed. With winter fast approaching it is important to "Be Prepared" (Boy Scout motto [50]) and not wait until the last minute. It is far better to be knowledgeable about these alternatives rather than be clueless. The same is true for planning your organization's Internet Protocol version 6 (IPv6) transition. Today is a great day to get started on your transition strategy. Whether we like the idea of transition to IPv6 or not, we may be in a situation to "love the one you're with" and IPv6 is the only possibility at this point in time. However, now our backs are against the wall and we have no choice but to jump directly to dual-stack [51].

Scott

IPv6 General discussions A+P Address plus Port CGN DS-Lite HIP IPv6 LISP LSN NAT444

Source URL: http://www.networkworld.com/community/blog/techniques-prolonging-lifespan-ipv4

Links:
[1] http://en.wikipedia.org/wiki/Ipv6
[2] http://en.wikipedia.org/wiki/World_population
[3] http://en.wikipedia.org/wiki/Illiteracy
[4] http://www.cbsnews.com/stories/2010/02/15/business/main6209772.shtml
[5] http://en.wikipedia.org/wiki/Drinking_water
[6] https://www.arin.net/
[7] http://www.txv6tf.org/?page_id=593
[8] http://www.seinfeldscripts.com/TheExGirlfriend.htm
[9] http://www.apnic.net/
[10] http://www.cisco.com/ipj
[11] http://www.cisco.com/web/about/ac123/ac147/archived_issues/ipj_14-1/ipj_14-1.pdf
[12] http://www.worldipv6day.org/
[13] http://www.circleid.com/posts/ipv6_transitional_uncertainties/
[14] http://www.cisco.com/web/about/ac123/ac147/archived_issues/ipj_13-2/132_address.html
[15] http://www.nanog.org/meetings/nanog53/presentations/Wednesday/Huston.pdf
[16] https://www.arin.net/policy/nrpm.html#eight
[17] https://www.arin.net/resources/request/transfers.html
[18] http://www.nanog.org/meetings/nanog53/presentations/Wednesday/Dul.pdf
[19] http://en.wikipedia.org/wiki/Hotelling's_rule
[20] http://www.networkworld.com/community/blog/cost-running-ipv4-network-will-increase
[21] http://www.networkworld.com/community/node/45776
[22] http://www.networkworld.com/community/node/45002
[23] http://www.networkworld.com/community/blog/can-large-scale-nat-save-ipv4
[24] http://www.networkworld.com/community/blog/ipv4-reputation-filtering-not-long-term-solut
[25] http://www.ietf.org/rfc/rfc1918.txt
[26] http://en.wikipedia.org/wiki/Betamax
[27] http://www.ietf.org/rfc/rfc2766.txt
[28] http://www.ietf.org/rfc/rfc4966.txt
[29] http://www.networkworld.com/community/blog/testing-nat64-and-dns64
[30] http://www.networkworld.com/community/node/46600
[31] http://tools.ietf.org/html/rfc6333
[32] http://www.networkworld.com/community/blog/tier-1-ipv4-tier-1-ipv6
[33] http://tools.ietf.org/html/rfc6346
[34] http://tools.ietf.org/html/rfc6346#section-3.4
[35] http://tools.ietf.org/html/draft-xli-behave-divi-pd-01
[36] http://www.youtube.com/watch?v=eYffYT2y-Iw
[37] http://en.wikipedia.org/wiki/Dynamic_dns
[38] http://en.wikipedia.org/wiki/Default-free_zone
[39] http://en.wikipedia.org/wiki/Locator/Identifier_Separation_Protocol
[40] http://datatracker.ietf.org/wg/lisp/
[41] http://lisp4.cisco.com/
[42] http://www.nanog.org/meetings/nanog50/abstracts.php?pt=MTYzMyZuYW5vZzUw&nm=nanog50
[43] http://www.lisp4.facebook.com
[44] http://www.lisp6.facebook.com
[45] http://gforge.info.ucl.ac.be/projects/openlisp/
[46] http://www.lisp4.net/
[47] http://en.wikipedia.org/wiki/Host_Identity_Protocol
[48] http://www.ietf.org/rfc/rfc4423.txt
[49] http://www.ietf.org/rfc/rfc5201.txt
[50] http://en.wikipedia.org/wiki/Scout_Motto
[51] http://www.networkworld.com/news/tech/2007/090507-tech-uodate.html