IPV4 Doing more with Less

Participants:

You

Fred Baker – Cisco Systems
Randy Bush – IIJ
Joel Jaeggli – Zynga
Charles Lee -Addrex

Genesis of the BOF Topic

```
On 12/9/11 10:37 , Franck Martin wrote:
I just had a personal email from a brand new ISP in the
Asia-Pacific area desperately looking for enough IPv4
to be able to run their business the way they would like…

This is just a data point.
```

- Apologies to Franck Martin, I found the post and the resulting thread interesting.
- We run networks that are going to have to support IPV4 for a while. For the most part I imagine we'll continue to run them in the absence of registries able to service requests for additional address space on the basis of need.

The Speakers

- Are here to address the topic in a variety of ways.
- Are comprised of:
  - Network equipment vendors
  - Transit network operators
  - Internet content providers
  - IP address brokers
  - You.

Agenda

- Fred Baker
- Randy Bush
- Charles Lee
- Joel Jaeggli
- Open Microphone
- Please:
  - Avail yourself of the microphone, this is a dialog not a lecture.
  - If you have a topic that you want surfaced please bring it to the mic and inject it where it seems most appropiate.

Internet Content Provider vantage point.

- Disclaimer:
  - These are my opinions, they are not that of my employer. Nor for that matter, are they the operational practices of my employer. I've been in the industry for 19 years at this point and I'm entitled to my own opinions.

Consumption

- ICPs do not (in general) naturally soak up as much IPV4 address space as consumer facing ISPs in proportion to the size of their customer base.
  - However:
    - scale drives consumption
    - state accumulates in uncomfortable places
    - While the default position today is to build the service in the cloud thereby externalizing (among other things) the address consumption, outgrowing your cloud much like outgrowing a PA assignment can be an uncomfortable position to be in.

Consumption II

- In a given year, 33-40% of ARIN prefix assignment requests are the first trip to the address well.
- Between 1.5% and 4% of the address space assigned in a given year is going to those folks
  - http://lists.arin.net/pipermail/arin-ppml/2012-January/024053.html
- While everyone dependent on the availability of more IPV4 as engine of business growth will be materially worse off when the well runs dry. The guys who have none now, will be entirely reliant on secondary and tertiary mechanisms.

Securing Address-Space

- Can still go to the RIRs for now.
- PA space
  - Have now seen/heard of existing customers getting dropped in order recover their address space.
- Transfer markets appear to be taking shape
- Going to have to make renumbering managable in your plans cause otherwise it's going to be a major risk item.

Securing Address space

- Addressing requirements:
  - It's not just box count or load balancer ip's that show up in address requests.
  - Outgoing NAT pools
  - De/Re-aggregation for regional DCs
  - Mechanics of rejiggering prefix usage to move DOS targets around or peel them off to third parties.

REST APIs (and DNS) and the grim meathook future of NAT Overload

- Everyone is used to overload NAT in some contexts (home routers for example), but when you deploy a large number of hosts in a datacenter talking to a small number of destinations such as a partner's api, bad things happen.
- ICPs as a general rule have lot more servers than IPV4 addresses.
- If a destination address receives a few hundred connections per second has a few hundred outstanding connections and 4000 closed ports in time-wait, the probability of a naive nat implementation reusing a port before it's closed and clobbering a new outgoing connection as a result is greater than 1 in 16.

Nat Overload

- You can:
  - Use a non-overload nat pool for a supply of outgoing port numbers.
    - Soaks up 16 ip addresses for every million port numbers
  - Not use brute force to solve an engineering problem
    - Bring the app in-house or into a connected private cloud so it doesn't traverse the NAT.
    - Encourage the app to use connection pooling/resuse radically reducing the connection rate.
    - Invert the relationship by proxying API requests through your own load balancers reusing ip addresses used for incoming connections.

Nat Overload

- Economies of scale, performance improvements, faster servers contribute to a steady rise in port demand from a given server count with the occasional drastic improvement, when the network engineers finally get through to the software engineers.
- While other large scale NAT applications face this problem with particularly hot destinations port controlled or deterministic NAT mappings may help (if you have enough Ips)
- Combination of overload + pool (if available) can ameliorate if not entirely alleviate this problem while creating others.
  - FTP PASV is toast when the same ip doesn't get used for the transfer connection.

NAT scale.

- Turns out, that by now, vendors can build boxes that can do 4 million nat sessions per second per box so it's not actually the volume that's the problem.
- When the scale gets beyond the boxes that exist or are economically feasible, one turns to the routing tools used for divvying up traffic to load-balancer frontends
  - Layer-3 ECMP
  - PBR/FBF
  - With the attendant liabilities and compromises associated with each one.

What can IPV6 do for you today?

- So far, most visible efforts I've seen the ICP space are the user accessible versions of services made available via IPV6.
  - There are others.
  - In the scheme of things making a service available to a small number of customers is not really a particularly hard challenge.
  - In the absence of a big inflow of customers it's mostly about readiness and developing operational competency.

IPV6 today

- Exposing public APIs via V6 has a lot of potential upside.
- Even if the backend isn't fully V6 enabled stateless nat46 could be used to move that workload away from IP4 NAT
- Pools of source port numbers become a lot easier to manage when an address per internal host binding can exist.

IPV6 today, outgoing connections

- I imagine, that for the most part internal to outgoing connections will continue to pass through a stateful device even as reliance on the ipv4 NAT declines.
- Doing so maintains the posture associated with unsolicited incoming connections to internal resources.
- Protects against certain kinds of DOS
  - http://tools.ietf.org/html/draft-ietf-v6ops-v6nd-problems-04
  - Otherwise has the same considerations as any other firewall application.

Cloud

- Cloud providers today don't seem entirely prepared to handle IPV6 internally.
  - Some can do it externally.
  - Very large Datacenters + multi-tenant + private cloud interconnects = VRF + more NAT

On that cheery note...

- We've been working under conditions of IPV4 scarcity for most of the commercial life of the internet.
- It's gonna get moreso, before it gets better.

We should be concerned about the costs borne by new entrants, the innovation disruption cycle that allowed the internet to thrive may find it more challenging and capital intensive then previously.

Thanks!