October 2016

Geoff Huston

NANOG 68

NANOG held its 68th meeting in Dallas in October. Here's what I found memorable and/or noteworthy from this meeting.

The meeting opened with Scott Bradner and a history of the IANA. Given that the arrangements with the US Government exercising some level of oversight on the IANA function lapsed on 1 October this year, then this brief history of the IANA up to this point seemed to be appropriate (I've separately written about his presentation - it can be found at http://bit.ly/2dxSM5V).

It seems that the distributed nature of the Internet is now waning and these days we are evolving into a data centre-centric Internet. The conversations that used to happen within an ISP about POP design and inter-POP flows have been replaced with conversations about Top of Rack, North/South and East/West flows and routing management within this data centre environment. In the same way that many ISPs shifted from carrying a full internal route set on OSPF to using iBGP, a presentation by Chris Woodfield of Twitter described a similar shift in Data Centre routing that is happening today. As Chris puts it: "BGP restores a "flat" routing model that is far more scalable than OSPF, with policy control point at every peer. It's easier to see paths, and diagnose issues." In this case they used the 94 million private 32-bit ASN space to perform automated ASN numbering, and also used a variety of automation approaches to take what could've been a daunting manual process into one that involved a set of discrete automated steps. It's like the evolution of Internet routing replayed in the context of an individual content distribution network! At the same time MAC-level switching just won't go away and the wide area form of Layer 2 VPN is now possible using VxLAN BGP EVPN to scale a simple L2 VPN across multiple sites. The objective? Allowing a content distribution network to host multiple tenants who each see a simple L2 or L3 network that connects the tenant's assets. Again it seems that it's now a data centre-centric Internet in terms of the current intense levels of attention from vendors and service providers.

It also is clear now that the dev-ops approach is finally getting traction and the world of human network operations centres driving a network of devices with ASCII CLIs and spreadsheets of managed inventory is, thankfully, now over. In place of myriads of ad-hoc expect scripts and operations centre process scripting we are using any one of a number of automation suites to hand the entire role over to automated control systems. All that's left is the tough question: Which one should you use? A presentation from Cloudflare justified their choice of SALT and NAPALM as the foundation of their automation environment. SALT allowed for a uniform approach to manage both servers and network devices and NAPALM has extensive multi-vendor capability that creates a vendor agnostic set of abstractions. Cloudflare manage both the network itself, their servers and the constellation of monitoring probes, through this single automation framework. Much of their network management tasks, including detection, isolation and remediation are now fully automated as a result. These days it seems that the traditional NOC is just a set of scripts in a devops framework!

Continuing on this view of the Internet as a massive CDN, where the critical points are the data centres, was a presentation from Facebook on their use of Identifier-Locator split Addressing. It's one of those unintended consequences that came from work in the routing community some 15 or so years

back. At that time there was considerable concern over the routing system growing faster than Moore's Law could handle, and we were looking for ways to slow the growth impetus of the routing system. One approach was the so-called Identifier/Locator split. It was noted that the Internet used overloaded semantics for an IP address, as it carried both the location of a device and its identity (this was clearly a pre-NAT world at the time!). A number of protocols were developed that addressed attempting to cleave apart these two concepts, one of which was ILNP. (Another was LISP, a ID/LOC split routing environment that gained greater prominence at the time, perhaps due to a strong push from Cisco). Facebook has turned to ILNP to use an internal IPv6 architecture that revives the old IPv6 8+8 split and couples it with ILNP. This allows them to define "container" applications where the application itself may move across standard Linux hosts yet still maintain a coherent identity. The application's process has a unique IPv6 address, where the first 64 bits are the routed location of the processing engine and the lower 64 bits are the process' unique identity across the entire Facebook processing context. This provides a simple IPv6 internal address architecture in Facebook's data centre model: every server has a unique /64 route, which is summarised into a /54 at the top of rack, which is summarised into a /46 in a multi-rack pod, which is summarised to an individual /32 for the data centre. Chef (yes, another devops tool) applies this /64 to every server host. Processes receive a unique 64-bit address based on a UUID64 plus some Facebook identity values added into the value. This leads to a number of possibilities, particularly when combined with eBPF extended packet filters and XDP, the Linux host's express data path. At one level of abstraction this is using the internet's own networking technologies to stitch together a collection of separate systems in a closed distributed environment to form a single data-centre scale virtual mainframe again! Yes, mainframe-style massive consolidated computing engines live once more, but it's not quite life as we knew it!

Aside from the data centre technology presentations there were a number of presentations dealing with the state of security and the issues of attacks and defence. This is well trodden ground and while there were no particularly new insights from this meeting, the overall picture remains very disturbing. The increasing population of poorly programmed and poorly maintained devices on today's Internet provides a continuing rich fodder for various forms of attack, and the release of a number of attack codes has made this issue significantly worse. When each enlisted device is sending just a couple of packets a second, if a million such devices can be enlisted in the attack the cumulative volumes become overwhelming to most of our deployed defences. This is not a pretty picture. It always seems so anomalous when I see various pundits loudly applauding the onset of the Internet of Things and the supposed benefits that such an onslaught of such devices will bring to our lives (and someone's pockets no doubt), and at the same time see these same poorly programmed and poorly maintained devices being coopted into a massive zombie army of attackniks. We can't have both futures, in that the Internet as we know it cannot survive in the face of the tremendous numbers of sub-standard devices being fielded in the consumer market. So either the device pundits need to do a far better job of quality control and assume complete responsibility for the massive damage poor quality devices cause to the rest of us, or we surrender the current model of the Internet and focus on building a fortress architecture that leaves the open Internet to become a toxic wasteland of uncontrolled and uncontrollable bot warfare. The latter outcome is looking more and more likely.

As well as brute force attack there is also the problem of data exfiltration. It's always a difficult decision whether to go public or not with a security vulnerability. Publicising it turns the tool over to the hands of an unknown set of adversaries without knowing whether the potential victims are also aware of the problem, while remaining silent carries the risk that the exploitation will be discovered in any case. In this particular case the exploit was covert data exfiltration using Javascript primitives, and Rackspace described a method of encoding the data in Base64 and then jamming this into a set of DNS queries as terminal QNAMES and sending the data to an accomplice authoritative name server. They devised a command and control system by encoding directives in the answers to MX queries. No doubt this use of the DNS as a covert communications channel is just one of many such exploits, but it underlines a particular lesson on system defence - disable everything you are not using, and for what you are using, ensure that you control its actions and access!

In the access ISP world Comcast reported on an experiment they conducted as a followup to the buffer bloat work. To briefly recap, "buffer bloat" was seen as a symptom of poor performance in access networks, where the excessive provision of buffer space in access routers became a source of delay. Comcast tested the Docsis 3.1 modems with three buffer sizes on the upstream channel and found that the smaller buffers tended to generate better performance. I suspect that there is much more going on there than just buffers - the issue is the interaction between the end-to-end flow control algorithm and the network's buffers is the root of protocol performance. Reno-style AIMD flow algorithms tend to operate in a mode that fills buffers while it performs a linear increase in the send window and supposedly drains the buffers immediately following rate halving. Sender pacing and delay sensitivity can be used instead to try to calibrate the controlled data flow to the onset of buffering and efforts like CUBIC and, more recently, BBR appear to head down this direction of attempting to oscillate the flow parameters around the onset of buffering, rather than pushing the data flow into the buffer space to the point of buffer overflow and packet loss.

The final presentation I want to comment on is one on BGP Large Communities. The exhaustion of the 16 bit ASN pool happened many years ago for mostly ISPs, and we have been using 32 bit ASNs for some time. But the problem has been that the size of the BGP community attributes has not kept pace. 32-bit community values allowed a network operator to define a target AS and a 16 bit policy setting. But of course the same capability requires 64 bits for the larger ASNs. It has been some time in coming, and not for want of trying (RFC5668, 4-Octet Extended Communities, the Flexible Community Attribute proposal and the Wide BGP community attribute) but finally there is a proposal to use 96 bit communities in BGP in a manner that is consistent with the use of the original Extended Community Attribute. This Large Community would allow 2 AS values and a 32-bit value setting. The only minor problem is that the draft has progressed to the point of a provisional IANA code allocation of attribute code 30 for these large community attributes and only now has it emerged that the vendor Huawei has been squatting on this same code value in its deployed equipment. Such blatant disregard for the standards progress by a vendor is to be deplored, but at the same time expediency demands that we work around such clashes and draw a different number from the BGP attribute number space. (https://tools.ietf.org/html/draft-ietf-idr-large-community-02).

This is by no means all that happened at NANOG 68. The presentations, both as slide packs and the YouTube videos can both be found at https://www.nanog.org/meetings/nanog68/agenda.

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 building the Internet within the Australian academic and research sector in the early 1990's. 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 and chaired a number of IETF Working Groups. He has worked as an Internet researcher, as an ISP systems architect and a network operator at various times.

www.potaroo.net

Disclaimer