The ISP Column
A column on things Internet

Geoff Huston
April 2017

IETF 98 Report

The IETF meetings are relatively packed events lasting over a week, and it's just not possible to attend every session. Inevitably each attendee follows their own interests and participates in Working Group sessions that are relevant and interesting to them. I do much the same when I attend IETF meetings. The IETF met for IETF 98 in Chicago at the end of March, and from the various sessions I attended here are a few personal impressions that I would like to share.

IEPG

For many years on the Sunday prior to the meeting there is a meeting of the IEPG (www.iepg.org). Back in the early nineties it had some kind to role in the context of the coordination of engineering of research Internet efforts, but these days it's a handy venue for some interesting presentations that try to have some relevance to the operational Internet, as distinct from the work on standards.

It seems that the shock of the revelations of widespread state-sponsored snooping as reported in the Snowden files has had lasting and significant implications. More than one half of web traffic is now encrypted. Encryption for servers, such as web servers, relies on the use of a domain name certificate that provides a third-party attestation that a particular public key is associated with a given domain name. The server hands the client this certificate and if the client is willing to trust this certificate it will then negotiate an encrypted session with the server, starting with an initial handshake that is encrypted with this public key. Domain Name certificates have been expensive and often considered to be inaccessible, but all this changed with the introduction of Lets Encrypt a little over a year ago. A study by Giovane Moura and collaborators in the Netherlands looked at the uptake of Let's Encrypt (http://www.iepg.org/2017-03-27-ietf98/letsencrypt-moura-ietf98-iepg.pdf). Let's Encrypt has issued some 30 million active certificates, and most of these certificates are for smaller servers, yet some of the larger servers are also using these certificates. They reported that this program has been successful in reducing the cost and complexity of certificates and dramatically changing the uptake of use of encrypted traffic on the Internet.

HOMENET

The Homenet Working Group has been in operation for some years. I am still highly confused if it's aim is to simplify home networks or to seriously complicate them! There is no doubt that this working group has encompassed some very advanced concepts, including multiple segments in the home network with different security realms, and attempted to automate much of their configuration. Their work has included defining frameworks for address management and prefix configuration for home routers, managing the routing domain, undertaking name resolution, supporting service discovery and of course this concept of multi-segmented network security. They have already spun off a new routing protocol working group (Babel), and are working some advanced concepts in DNS service discovery.

But the largest debate in this Working Group at IETF 98 was not about any of these detailed technical topics. It was about a name. A recent product of this working group, RFC 7788, said that "A network-wide zone is appended to all single labels or unqualified zones in order to qualify them. ".home" is the default." However, it was subsequently pointed out that the days of simply grabbing a DNS top level domain by simply saying so in a published RFC are long gone! However, when this was reviewed by the Working Group, the proponents persisted in this approach and advocated the use of ".homenet" instead, which seems to make little in the way of progress with the substance of the problem that was originally raised. This Working Group-intended process of name allocation of a new Top Level Domain (TLD) in the root zone of the DNS wss a few million miles away from the new TLD process used by ICANN, and it did look a little strange to see the Working Group make this claim. However, by the end of the week both an Area Director and the IAB had made their pronouncements that if the any Working Group in the IETF wanted to allocated a delegated DNS label, then the least friction was via a delegation under the ".arpa" TLD. So it looks like some variant of "home.arpa" will be the new label for our home network domains.

Distributed Mobility Management

The forays of the IETF into the concept of network mobility has a long and distinguished history. While experiments were conducted into various approaches in the early 1990s, the initial specification was published in 1996, RFC 2002, "IP Mobility Support". There are numerous revisions to this original specification, including Mobile IPv6, dual stack Mobile IP and transitioning strategies. However, as far as I can tell the 7 billion or so mobile devices attached to the Internet just don't use any of this collection of technical standards. While Mobile IP envisioned a seamless environment where the mobile device could roam in and out of various cellular domains and maintain a seamless IP conversation with a static device, or even another mobile IP device, the world of mobile devices is far more prosaic. If you roam away from your selected provider's handoff-capable radio network of base stations you lose. But toiling away at Mobile IP concepts continues at the IETF, and the Distributed Mobility Management Working Group met at IETF to press on with aspects of this work. It's unclear to me to what extent this work has direct application in the Internet's mobile systems, or even if these technologies are widely used in enterprise virtual private networks.

Congestion Control

As well as Working Groups that are chartered to produce standards specifications, there are also a number of Research Groups who meet at the IETF. The issue of session congestion control is a longstanding issue for IP networks. The original Internet architecture conceived of a stateless packet forwarding system whose interior elements consisted of simple buffered switches, each of which could directly forward a packet, or store it in a local buffer for subsequent forwarding, or drop the packet completely. It was left to the end systems to regulate their packet sending rates, using the signals of latency extension and packet drop as indicators of incipient path congestion as the input to this packet flow regulation function. These days nothing quite so naïve works very well. The interior of all networks is filled with various forms of flow regulators, prioritisation systems, traffic policy enforcement systems and similar. In the worst case these network systems can work in direct opposition to the host systems, with the result that the end-to-end can be coerced into pumping packets into a congestion-ridden abyss. However, at the other end of the spectrum the information flow between network and host can be used to mutual advantage. A presentation at this session was on the topic of "Throughput Guidance" where middleware informs the data sender, via an inserted TCP option in an ACK packet, of the sustainable bitrate for the session.

For many years, the operating system provided applications with a single TCP control protocol. We are now in an environment when an application may select a particular flow control algorithm based on the nature of the application and its anticipated data flow characteristics and derived information about the nature of the end-to-end path. Countering this is the issue of ossification where network middleware interacts badly with all but the well established control protocols and where APIs cannot be readily updated to support new transport options.

The session also included an update on Google's recently announced BBR protocol, looking at results that compare BBR and Cubic on an LTE cellular network, and results that look at path contention between BBR and a Reno flow compare to CUBIC and a Reno flow.

DNSOPS

The DNS remains a subject of intense study in the IETF, and much of this work is now considered in the DNSOPS Working Group.

In this session, there was a novel use of cryptography to improve upon the current model of obfuscated denial of existence in DNSSEC. The initial problem was that a number of registries, notably Verisign's operation of the .com and .net registries, did not want to allow third parties to enumerate the contents of these two zones (exactly why they felt that zone enumeration was detrimental to their commercial interest as a registry is probably the subject of an entire article in its own right – but let's not go there right now!). The result was NSEC3 (RFC5155) where the range field of the NSEC record was changed to be a range of hashed values. If the hashed value of the requested label sat between the two hash values in an NSEC3 record than the label was provably absent from the zone. However, hash functions can be reversed and there are now kits (https://github.com/anonion0/nsec3map is one such) that can be used to enumerate NSEC3 signed zones. DNSOPS was presented with a new algorithm, NSEC5, that replaces the simple symmetric hash function with a public/private key par hashing function.

The entries in the zone are sorted into a hash order using the private key to generate the hash values. Responses to queries for non-existent names require the server to generate a "proof" based on the query name and the private hash key. The returned response includes both the range indicator and this "proof. The client can validate the response by using the public part of the hash key to confirm that the query name and the proof are related. It's certainly a novel approach, but it's unclear to what extent the larger computation load in using asymmetric cryptography is viable for large scale zone servers of signed zones.

An interesting, and potentially very useful proposal, was to use an EDNS0 option to pass a richer set of error codes than are used in the RCODE set. The basic motivation is that the choice of SERVFAIL as an indicator from a recursive resolver to a client that the requested query failed DNSSEC validation has had the unfortunate result of increasing the query rates for DNSSEC-broken domains. This option would allow a recursive resolver to indicate that there is no answer because of a particular DNSSEC validation outcome, allowing the client to cease the query at this point if it so chooses.

NTP

NTP continues to meet at IETF meetings, and the work these days appears to be concentrated in the area of securing the protocol.

As with the DNS, one of the challenges is to secure the UDP packet exchange for NTP sessions. The DTLS protocol, an adaptation for TLS to run over UDP, appears to be the obvious choice, but the interactions of the stateful TLS protocol with a fragmenting UDP datagram transport service is proving to be a challenge. It appears that another option is to use TCP to perform the key exchange between NTP peers and then use NTP packets with NTS extensions for time operation.

Another interesting proposal was to reduce the amount of information leakage inherent in the REFID field while preserving its loop detection properties. An interesting twist in the draft was to also encode a server's leap second handling mode (smear or jump) into the REFID so that clients would understand the particular shift of time during the leap second event used by each server.

Inter-Domain Routing

The inter-domain routing protocol, BGP4, continues to be the subject of much of the IETF's more esoteric tweaking efforts! This time some 14 agenda items were included in a 90 minute meeting of the working group.

The recent work on simple large communities for BGP encountered cases of vendors squatting on unallocated code points for attributes. It's not a tightly constrained space and a First Come First Served registration policy would seem like a good way to stop squatting, but of course this is the IETF and adding process is indeed an institutional forte. So, the current proposal is to engage all the chairs of BGP-related working groups as an expert review of code point allocations. Sometimes it's hard to understand the relationship between the original problem and the proposed solution!

Getting rid of features in BGP is proving tricky. The concept of route aggregation, where a BGP speaker would take a collection of individual prefix advertisements and substitute a single aggregate announcement, was seen as a major component of the shift away from class-based routing in the 1990s, and the notions of AS SETs, AS_CONFED_SETs and the ATOMIC_AGGREGATE attribute were introduced to support this form of proxy aggregation of route advertisements. It seems that all this has fallen foul of the long standing efforts to secure the inter domain routing space, and are now being removed from the standard specification. But don't hold your breath about seeing AS Sets disappear anytime seen from BGP. There are still a few hundred instances in the global IPv4 routing tables, and while their use today is largely unnecessary, eradicating all traces of their use seems to be beyond us. Once standards and common practice diverge, it's not the common practice that is the loser here – it's the standards body that is perceived to be losing credibility.

BGP is a protocol that talks about connectivity. Most of the time the BGP control conversation and the connectivity that BGP manages occurs over the same infrastructure, so a break in data plane connectivity is also a break in control plane connectivity. But when the two are not the same, as in the case of route servers, then a break in next hop adjacency (data plane) may not be visible to the route server (control plane). The proposed solution: add more knobs to BGP of course! This one is a proposal to include next-hop reachability into BGP so that clients of a route server can inform the server when Next Hop reachability is impaired from their perspective.

Difficult problems never go away, and the problem of route leaks can be an exceptionally difficult problem. In the case of a route leak you cannot fault BGP itself. BGP is acting as BGP should, propagating routing information across the routing space. The failure is one of policy, where the routes being propagated are counter to the routing policies of a network. So let's add policies to a BGP speaker. The long standing approach has been by using registries of route policies. Individual networks publish their routing policies via RPSL, the Routing Policy Specification Language, and other networks download these policies and construct route filters from them. Over the decades a lot of work has been put into routing registries, but it has not been all that successful. One of the impediments here is that RPSL is a complex language, and operators have a strange love-hate relationship with complexity! The latest effort is to swing the solution pendulum to the other end of the spectrum and simply label each eBGP peering relationship as either a peer-to-peer or a customer-to-provider. Crude, but it is also extremely simple. Route leaks of the nature where a customer leaks the routes learned from one provider to another can be readily detected and prevented. Of course, other forms of route policy, such a geographically limited routing ("please, don't transit the US with my packets" for example), or avoidance of explicit transit operators ("please don't pass my packets through transit provider X"), cannot be so readily expressed with such a simple policy language. However, there is the hope that if complex RPSL solutions have not really worked, maybe incredibly simple ones might achieve most of what we want.

Any protocol design is the outcome of a set of design trade-offs. When given a couple of possible paths it's a typical response to pick one. This approach is consistent with Occam's razor in a search for approaches that as simple as they need to be, but no simpler. But this often leaves a number of paths not taken on the floor. In routing, one of these paths not taken was source-based routing. Packets are passed into the network with a description of the intended destination and it's up to the network to determine how to get to the specified destination. With source routing, or even segment routing, the source provides the network with some details of the path to be followed. For a dedicated IETF attendees this represents a bonanza of new specifications to be written of course, and BGP is certainly part of this. “BGP signalling of IPv6-Segment-Routing-Based VPNs” is as good as any as an example of this work of picking up these originally discarded design choices and re-animating them with new protocol knobs and new features.

Multi-Path TCP

I happen to love the ideas behind Multi-Path TCP. The idea that you can exploit path diversity between two points and splay a single TCP flow across such paths introducing interesting possibilities. Conventionally it has been used in mobile devices, and the MP-TCP operation has generally used WiFi and cellular radio as its media.

A presentation at the MP-TCP Working Group showed a very different application. Some NASA airborne applications use Iridium modems to communicate with “on-board payloads” in planes. The conventional approach has been to set up a number of concurrent channels and bond them together into a single logical link using the standard layer 2 bonding protocol, Multi-Link PPP. They have noticed poor TCP performance and a breakdown in MLPPP performance when more than 4 channels are used. Iridium can be slow (2.4Kbps per channel), clunky (channels drop out intermittently) and the RTTs are very long (4 seconds for a 500 byte packet). Using 4 channels with MLPPP they get a degraded mode when one of more channels are interrupted some 25% of the time during a 13 hour flight. Their configuration was changed by replacing the MLPPP unites with MPTCP. The reported results are impressive, allowing long lived MPTCP sessions to operate with dynamic adaptation to the amount of available transmission resources. The observation in this case is that a single TCP flow control that is imposed across a dynamically varying layer 2-aggregated substrate performs far worse than an approach that places a flow control state across each component element of the substrate. This is most impressive work that describes a novel, but very useful, application of MP-TCP.

QUIC

Google have taken their QUIC work to the IETF, and the reaction is entirely predictable.

QUIC is seen as a way of avoiding the rampant volumes of intrusive network middleware unit that all share an overwhelming urge to manipulate the sessions that fall within their clutches. The result is that the end-to-end flow control us now not only having to sense and react to perceptions of the conditions of the network path, but also try and compensate for arbitrary manipulation of its own flow control parameters in flight. QUIC can be seen as taking the ultimate response to intrusive middleware by simply taking everything off the table.

In essence, QUIC uses a UDP packet exchange between the end points, so there is no visible TCP for the network middleware to play with. QUIC encrypts most of its UDP payload, so even if middleware understood pre-encrypted QUIC packet formats, the result of encryption is that nothing is visible anymore. Within this encrypted payload is a fully function application level implementation of an end-to-end session, a TCP-derived transport protocol, and a port of HTTP/2 that selectively refined this protocol to match QUIC’s strengths.

The QUIC session at the IETF appeared to have two major groups present. One set of comments was along the lines of “let’s make it better”, seeking to improve upon the QUIC model. But another discernible theme of comments appeared to be consistent with the desires of middleware makers, attempting to coerce parts of QUIC back out from its encryption shield so as to allow middleware introspection and manipulation, one suspects.

We have come a long way from the model of “dumb application, smart tools in the protocol stack, dumb network", which typified the original IP architecture. Applications are folding in their own communications requirements and using only the most basic of protocol stack services. Why? To hide from a network that is no longer “dumb” and instead is perceived as being potentially malevolent!

It will be interesting to watch over the next few months whether Google's decision to take QUIC to the IETF will turn out to be a positive experience that improves both the protocol and its acceptance in the Internet, or whether the search for an acceptable level of consensus between a set of diametrically opposed viewpoints compromises the standard specification of QUIC to the extent that no-one, even Google, will want to use the IETF's product!

6MAN

One of the more amusing items of the agenda of this working group is the effort to bring the IPv6 specification to a "full standard".

Years ago, the IETF used to pride itself on being nimble and able to publish a standard specification in a fraction of time that was taken by other international technical standards bodies. But that was then and this is now. Now it has taken 22 years for the IPv6 specification to not be published as a Full Standard.

Now, in some ways this is a "Good Thing", as there are still tweaks and changes being made to aspects of the IPv6 specification, such as RFC8106, a recent document that updates the inclusion of DNS configuration data into Router Advertisements in IPv6. Such changes and tweaks are illustrative of a protocol that is being used, studied and adapted, which is good. But at the same time the fact that the core specification for IPv6 are not a full standard gives the protocol an air of incompleteness and a subtext of "not yet ready for use".

The IETF has reacted to this by attempting to advance updated version of the "core" IPv6 protocols to full standard. This set is updated versions of RFC2460 (IPv6 specification), RFC4291 (IPv6 Addressing Architecture), RFC1981 (Path MTU discovery), RFC4443 (ICMPv6) and RFC3595 (Flow Labels in MIBs). This move has generated some considerable controversy. In particular, the use of IPv6 Extension Headers is a fraught issue. Can intermediate systems (routers) add extension headers to an IPv6 packet, or is it only an option open to the packet source? It seems strange to me that this is such a controversial topic, given that there is a lot of network equipment in use out there that drops any IPv6 packet that contains an Extension Header (RFC7872). Arguing in favour of having intermediate systems adding an additional packet element in the nature of an Extension Header that increases its risk of being subsequently dropped seems rather strange, but then again this is an IPv6 working group, so a certain suspension of belief seems appropriate on all matters! The case against standardisation in this respect is that there is a realm of use of IPv6 header insertion, particularly in private use networks it seems somewhat anomalous to publish an Internet Standard that deliberately precludes what is reported to be common practice. The deeper dichotomy is about the views that cast a full Internet Standard as a "big deal", and that action should reflect a level of technical maturity that precludes continual further tinkering, and the opposed view, where the needs for further adaptation are constant in a "live" and active protocol, and we should not imbue a full Standard classification with an inappropriate and limiting assumption of stability and permanence. Standards can and do change, and we should be able to keep on changing. In this case, the matter was further complicated by the process question raised by an Area Director taking an editor's pen to a draft under IETF Last Call, and the 6Man Working Group got itself rather wound up on this topic!

A related issue concerns the 64+64 bit address architecture in RFC4291bis. There is simply no agreement over this fixed boundary architecture, and on one side is the view that fixed boundaries in address architectures are completely without technical merit and networks should be able to set this dynamically, as is the case with IPv4. The other side of the conversation points to the benefits in a clear boundary between the network address component and the interface address part, and a 64 bit boundary is a Fine Place to make the distinction between network and host addresses. Auto-configuration tools, such as SLAAC, clearly benefit from this fixed boundary in the address architecture.

Author

Geoff Huston B.Sc., M.Sc., is the Chief Scientist at APNIC, the Regional Internet Registry serving the Asia Pacific region.

www.potaroo.net

Disclaimer

The above views do not necessarily represent the views or positions of the Asia Pacific Network Information Centre.