The
Internet Protocol *Journal*

July 2024 Volume 27, Number 2

A Quarterly Technical Publication for Internet and Intranet Professionals

In This Issue

You can download IPJ back issues and find subscription information at:
www.protocoljournal.org

ISSN 1944-1134

From The Editor

Internet access by means of *Low Earth Orbit* (LEO) satellites has become very popular in recent years, particularly in rural areas where alternative solutions are limited. We covered this technology in an article in our September 2023 issue (Volume 26, No. 2). The benefits of LEO systems include a much lower cost to launch and place the satellites into a low orbit, and a shorter *Round Trip Time* (RTT) as compared to solutions involving geosynchronous satellites. However, since LEO satellites move across the sky, a complex system of tracking and handoffs is deployed in order to provide continuous connectivity to the end user. In our first article, Geoff Huston examines the performance of Starlink from the point of view of the *Transmission Control Protocol* (TCP).

When I joined the *Network Information Center* (NIC) at SRI International in 1984, I was handed two *Request For Comments* (RFCs) describing the *Domain Name System* (DNS), and I was told that the DNS would soon be deployed across the Internet (mainly known as ARPANET and MILNET at the time). The NIC was still maintaining and publishing a host table in 1984, and it would take a couple of years before the DNS became fully operational. Our second article, also by Geoff Huston, looks at how the DNS has evolved in the last 40 years with various enhancements and extensions. The DNS is still one of the most active areas of work within the *Internet Engineering Task Force* (IETF).

As the Internet has evolved, interest by governments and intergovernmental organizations has grown to legislate and regulate various aspects of the system. These efforts, often collectively referred to as *Internet Governance*, are sometimes developed in ways that do not fully include input from the Internet technical community. One example is the *Global Digital Compact* (GDC) currently being drafted by the United Nations. In our Fragments section you will find a letter from individuals concerned about the latest GDC draft.

Publication of this journal is made possible by the generous support of our donors, supporters, and sponsors. We also depend on your feedback and suggestions. If you would like to comment on, donate to, or sponsor IPJ, please contact us at **ipj@protocoljournal.org**

—Ole J. Jacobsen, Editor and Publisher
ole@protocoljournal.org

A View of Starlink from a Transport Protocol

by Geoff Huston, APNIC

Digital communications systems always represent a collection of design trade-offs. Maximising one characteristic of a system may impair other characteristics, and various communications services may offer different performance characteristics based on the intersection of these design decisions with the physical characteristics of the communications medium. In this article I'll look at the Starlink service[0,1], and how the *Transmission Control Protocol* (TCP)—the transport-protocol workhorse of the Internet—interacts with the Starlink service.

To start, it's useful to recall a small piece of Newtonian physics from some 340 years ago[2]. On the surface of the earth, assuming that you are high enough to clear various mountains that may be in the way—and also assuming that the earth has no friction-inducing atmosphere—if you fire a projectile horizontally fast enough it will not return to the earth, but head into space. There is, however, a critical velocity where the projectile will be captured by the earth's gravity and neither fall to ground nor head out into space. That orbital velocity at the surface of the earth is some 40,320 km/sec. The orbital velocity decreases with altitude, and at an altitude of 35,786 km above the surface of the earth the orbital velocity of the projective relative to a point on the surface of the spinning earth is 0 km/sec. This altitude is of a geosynchronous equatorial orbit, where the object appears to sit at a fixed location in the sky.

Geosynchronous Services

Geosynchronous satellites were the favoured approach for the first wave of satellite-based communications services. Each satellite could "cover" an entire hemisphere. If the satellite was on the equatorial plane, then it was at a fixed location in the sky with respect to the earth, allowing the use of large antennas. These antennas could operate at a low signal-to-noise ratio, allowing the signal modulation to use a high density of discrete phase amplitude points, which lifted the capacity of the service. All these advantages have to be offset against the less-favourable aspects of this service.

Consideration of crosstalk interference between adjacent satellites in geosynchronous orbits resulted in international agreements that require a 2° spacing for geosynchronous satellites that use the same frequency, so this orbital slot is a limited resource: it is limited to just 180 spacecraft if they all use K band (18–27 GHz) radio. At any point on the earth there is an upper bound to the signal capacity that can be received (and sent) using geosynchronous services.

It is relatively expensive to place satellites into this orbit because it generally requires three-stage rockets to propel them into this high orbit.

Depending on whether the observer is on the equator directly beneath the satellite or further away from this point, a geosynchronous orbit satellite is between 35,760 and 42,664 km away, so a signal *Round-Trip Time* (RTT) to the satellite and back will be between 238 and 284 ms in terms of signal propagation time. In IP terms, a RTT will be between 477 and 569 ms, and signal encoding and decoding times must be added to that. In addition, the delay for the signal to be passed between the endpoints and the satellite earth station must also be added. In practice, a RTT of around two-thirds of a second (660 ms) for Internet paths that use geosynchronous satellite services is common.

This extended latency means that the endpoints need to use large buffers to hold a copy of all the unacknowledged data, as is required by the TCP protocol. TCP is a feedback-governed protocol that uses ACK pacing. The longer the RTT the greater the lag in feedback, and the slower the response from endpoints to congestion or to available capacity. The congestion considerations lead to the common use of large buffers in the systems that drive the satellite circuits, which can further exacerbate congestion-induced instability. In geosynchronous service contexts, the individual TCP sessions are more prone to instability and they experience longer recovery times following low events[3].

Low Earth Orbit Systems

A response to this situation is to bring the satellite closer to earth. This approach has several benefits. The spinning iron core of the earth generates a magnetic field, which traps energetic charged solar particles and redirects them through what is called the *Van Allen Belt*, thus deflecting solar radiation. Not only does this deflection allow the earth to retain its atmosphere, but it also protects the electronics of orbiting satellites that use an orbital altitude below 2,000 km or so from the worst effects of solar radiation. It's far cheaper to launch satellites into a *Low Earth Orbit* (LEO), and these days SpaceX can do so using reusable rocket boosters. The reduced distance between the earth and the orbiting satellite reduces the latency in sending a signal to the satellite and back, which can improve the efficiency of the end-to-end packet-transport protocols using such satellite circuits.

This group of orbital altitudes, from some 160 to 2,000 km, are collectively termed LEOs[4]. The objective is to keep the orbit of the satellite high enough to prevent its slowing down by grazing the denser parts of the earth's ionosphere, but not so high that it loses the radiation protection afforded by the Inner Van Allen belt. At a height of 550 km, the minimum signal propagation delay to reach the satellite and return to the surface of the earth is just 3.7 ms.

But all of these facts come with some different issues. At a height of 550 km, an orbiting satellite can be seen from only a small part of the earth. If the minimum effective elevation to establish communication is 25 degrees of elevation above the horizon, then the footprint of the satellite is a circle with a radius of 940 km, or a circle of area 2M km^2.

To provide continuous service to any point on the surface of the earth (510.1M km^2), the number of orbiting satellites must be a minimum of 500. This reality implies that a satellite-based service is not a simple case of sending a signal to a fixed point in the sky and having that single satellite mirror that signal down to some outer earth location. A continuous LEO satellite service must use a continual sequence of satellites as they pass overhead and switch the circuit path across to successive satellites as they come into view.

At this altitude, the satellite orbits with a relative speed of 27,000 km/hour and it passes across the sky from horizon to horizon in less than 5 minutes. Some implications for the design of the radio component of the service are evident. The satellites are close enough that there is no need to use larger dish antennas that require some mechanised steering arrangement, but this situation itself it not without its downsides. An individual signal carrier might be initially received as a weak signal (in relative terms), increase in strength as the satellite transponder and the earth antenna move into alignment, and weaken again as the satellite moves on. Starlink's services use a phased-array arrangement with a grid of smaller antennas on a planar surface, which allows the antennas to be electronically steered by altering the phase difference between each of the antennas in the grid. Even so, this arrangement is relatively coarse, so the signal quality is not consistent, implying a constantly variable signal-to-noise ratio as the phased-array antenna tracks each satellite.

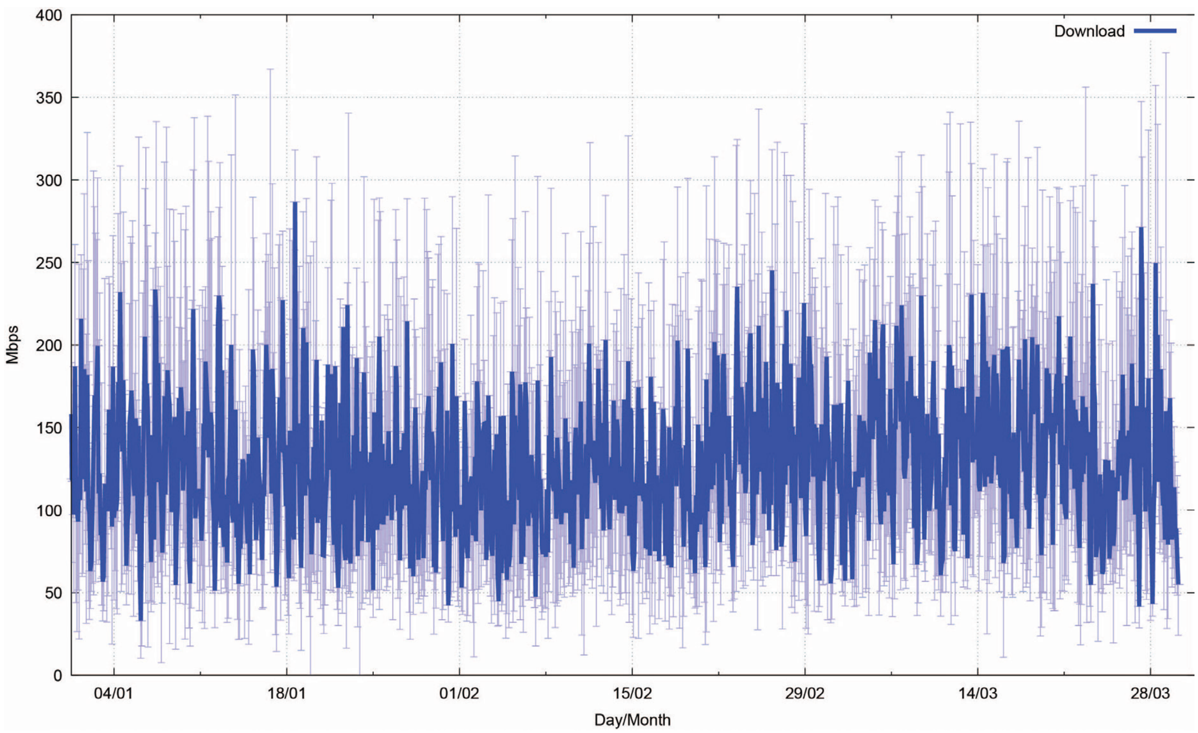
The modulation of this signal uses adaptive phase amplitude modulation, and as the signal-to-noise level improves, the modulator can use a larger number of discrete code points in this phase amplitude space, thus increasing the effective capacity of the service even while using a constant-frequency carrier signal. The implication is that if the satellite service attempts to always operate at peak efficiency, then it must constantly adapt its signal modulation to take advantage of the instantaneous signal-to-noise ratio, which results in a constantly varying service capacity.

Now we have four major contributory factors for variability of the capacity of the Starlink service, namely the variance in signal modulation capability, which is a direct outcome of the variable signal-to-noise ratio of the signal, the variance in the satellite path latency due to the relative motion of the satellite and the earth antennas, and the need to perform satellite switching constantly, and the variability induced by sharing the common medium with other users.

One way to see how this variability affects the service characteristics is to use a capacity measurement tool to measure the service capacity regularly. The results of such regularity of testing are shown in Figure 1. Here the test is a Speedtest measurement test[5], performed on a 4-hourly basis for the period January 2024 through March 2024.

The service appears to have a median value of around 120 Mbps of download capacity, with individual measurements reading as high as 370 Mbps and as low as 10 Mbps, and 15 Mbps of upload capacity, with variance of between 5 and 50 Mbps.

Figure 1: Starlink Performance



In Internet terms, *ping*[6] is a very old tool. However, at the same time it is very useful which probably explains its longevity. Figure 2 shows a plot of a continuous (flood) *ping* across a Starlink connection from the customer-side terminal to the first IP endpoint behind the Starlink earth station.

The first major characteristic of this data is that the minimum latency changes every 15 seconds. It appears that this change correlates to the user's being assigned to a different satellite, which implies that the user equipment "tracks" each spacecraft for 15-second intervals. This period corresponds to a tracking angle of 11 degrees of arc.

The second characteristic is that loss events are seen to occur at times of switchover between satellites (as shown in Figure 3), as well as occurring less frequently as a result of obstruction, signal quality, or congestion.

Figure 2: Starlink Ping Profile

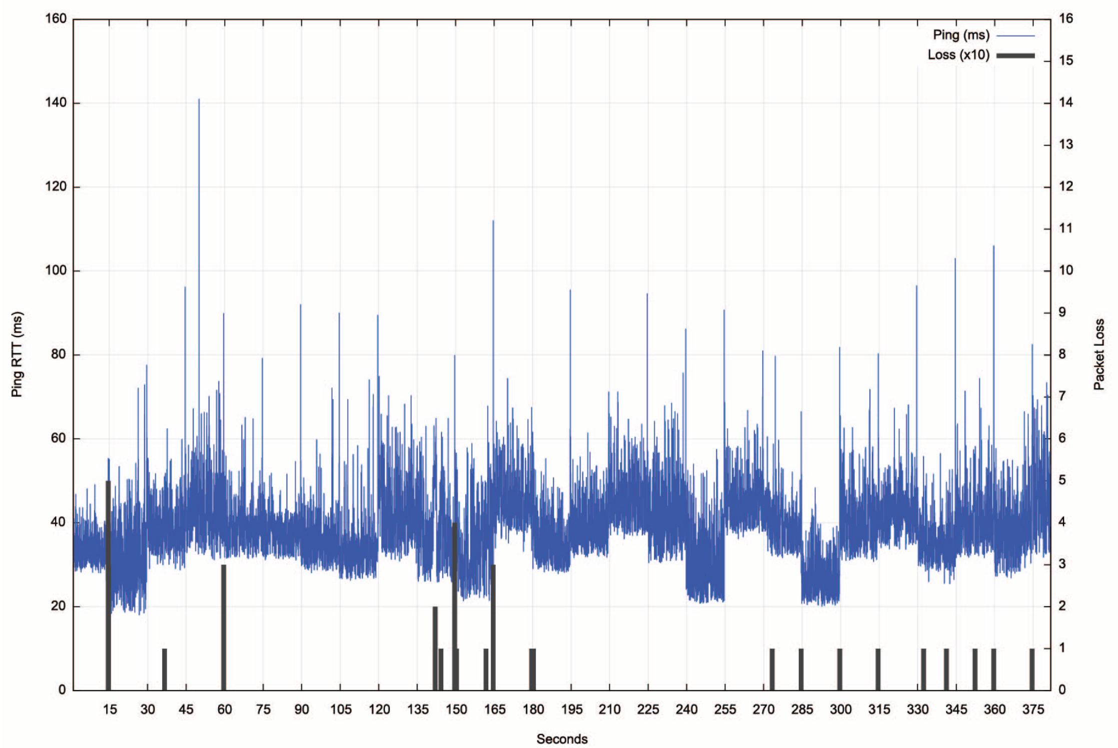
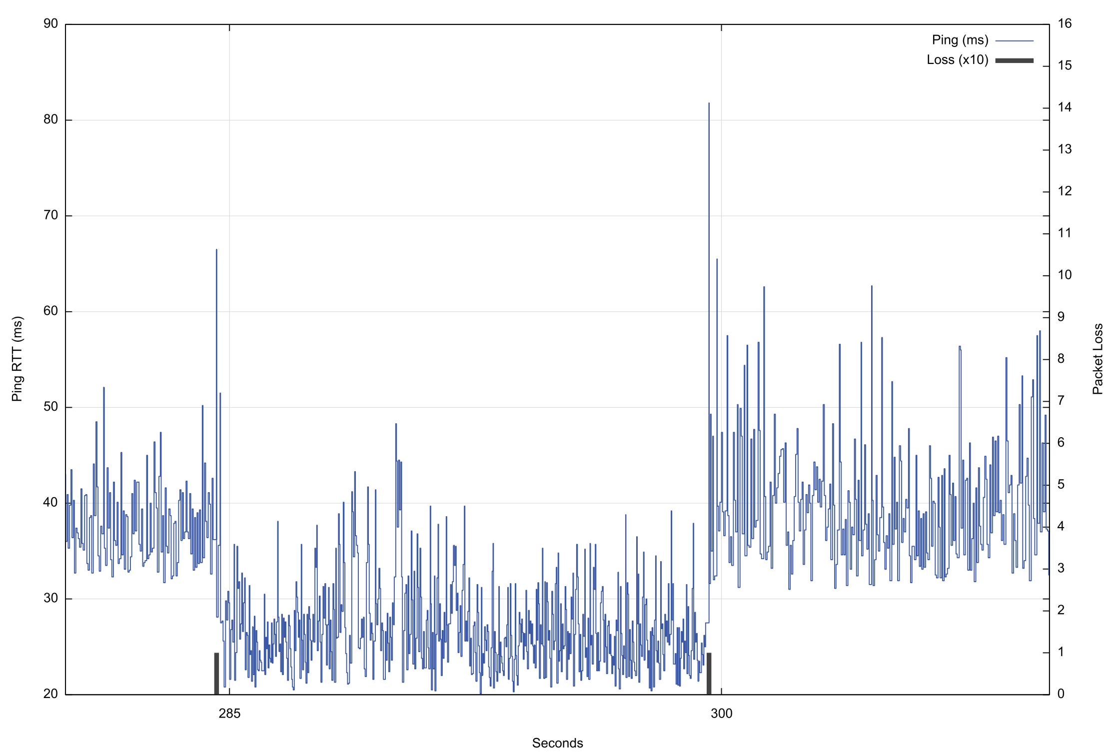


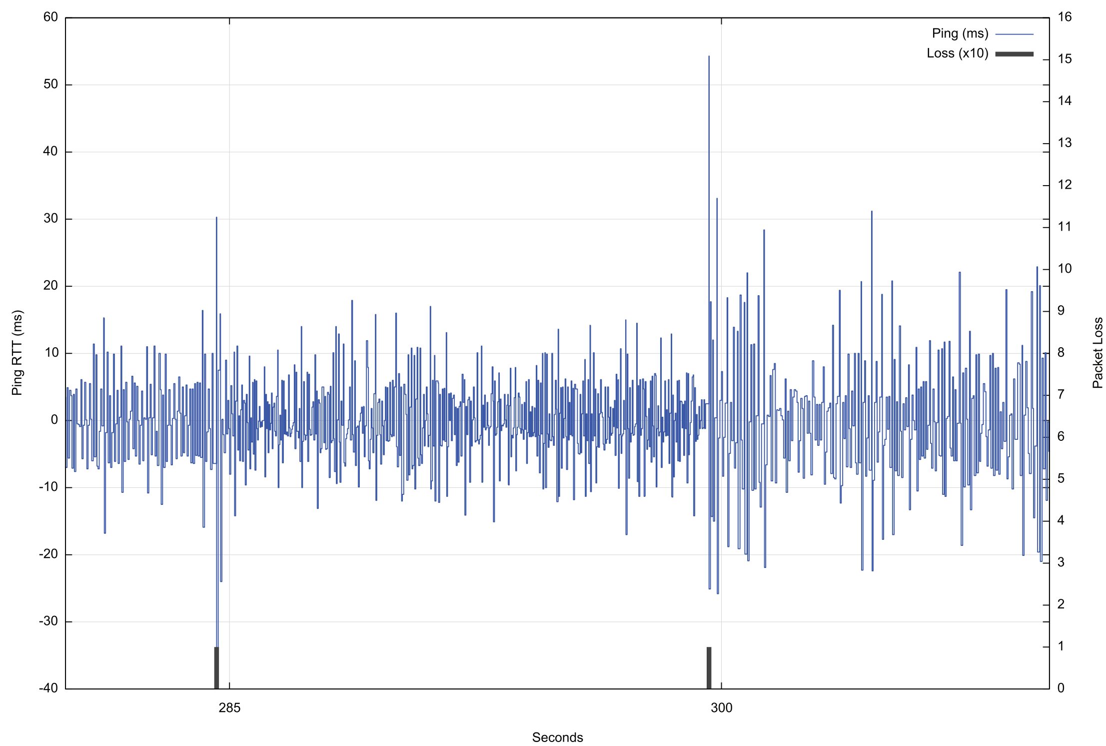
Figure 3: Starlink Ping Profile Showing Satellite Handover



The third characteristic is a major shift in latency when the user is assigned to a different spacecraft. The worst case in this data set is a shift from a minimum of 20 ms to a minimum of 40 ms.

Finally, within each satellite tracking interval the latency variation is relatively high. The average variation of jitter over successive RTT intervals is 6.7 ms. The latency spikes at handover impose an additional 30 to 50 ms, indicating the presence of deep buffers in the system to accommodate the transient issues associated with satellite handover (Figure 4).

Figure 4: Starlink Ping Profile Showing Latency Variance



The overall packet-loss rate when measured using 1-second paced *pings* over an extended period is a little over 1% as a long-term average loss rate.

TCP Protocol Performance

TCP[7] is an instance of a sliding window positive acknowledgement protocol. The sender maintains a local copy of all data that has been passed into the communications systems and discards that data only when it has received a positive acknowledgement from the receiver.

Variants to TCP are based on the variations in the sender's control of the rate of passing data into the network and variations in the response to data loss. The classic version of TCP is one that uses a linear inflation of the sending window size while there is no loss, and halves the window in response to loss.

The algorithm is called the RENO TCP control algorithm. Its use in today's Internet has been largely supplanted by the CUBIC TCP control algorithm[8], which uses a varying window inflation rate that attempts to stabilise the sending rate at a level just below a level that causes the buildup of network queues, which ultimately leads to packet loss.

In general terms, there is a small set of common assumptions about the characteristics of the network path for such control algorithms:

- There is a *stable* maximal capacity of the path, where the term stability describes a situation where the available path capacity is relatively constant across a number of RTT intervals.
- The amount of *jitter* (variation in end-to-end delay) is low in proportion to the RTT.
- The average packet-loss rate is low. In the case of congestion-based loss, the TCP control algorithm generally interprets packet loss as a sign that the network buffers have filled and the loss is an indication of buffer overflow.

Obviously, as we've noted, the first two conditions do not hold for end-to-end paths that include a Starlink component. The loss profile is also different. There is the potential for congestion-induced packet loss, as is the case in any non-synchronous packet-switched medium, but an additional loss component can occur during satellite handover, and other impairments can further affect the radio signal.

TCP tends to react to such environments by using conservative choices.

The RTT estimate is a smoothed average value of RTT measurements to which is added the mean deviation of individual measurements from this average. For Starlink, with its relatively high level of individual variance in RTT measurements, this estimate means that the TCP sender may operate with a RTT estimate that is too high, which in turn will result in a sending rate that is lower than the available end-to-end capacity of the system.

The occurrence of non-congestion-based loss can also detract from TCP performance. Conventionally, loss will cause the sender to quickly reduce its sending window on the basis that if this loss is caused by network buffer overflow, then the sender needs to allow these buffers to drain. The sender will then resume sending at a lower rate, which should restore coherency of the feedback control loop.

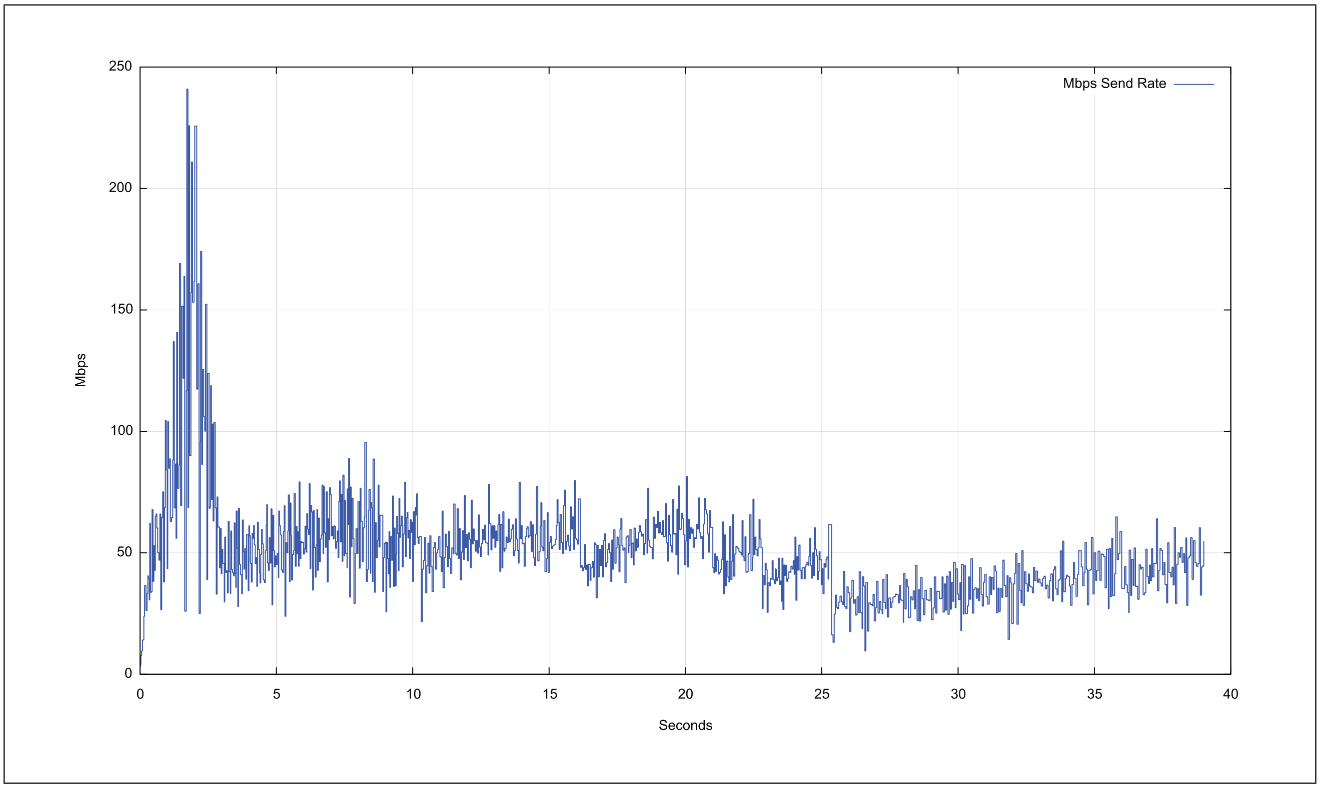
How does this mechanism work in practice?

Figure 5 shows a detailed view of a TCP CUBIC session over a Starlink circuit. The initial 2 seconds show the *slow start* TCP sending rate inflation, where the sending window doubles in size for each RTT interval, reaching a peak of 250 Mbps in 2 seconds. The sender then switches to a rapid reduction of the sending window in the next second, dropping to 50 Mbps within 1 second.

At this point the CUBIC congestion-avoidance phase appears to kick in, and the sending rate increases to 70 Mbps over the ensuing 5 seconds. A single loss event occurs that causes the sending rate to drop back to 40 Mbps in second 8. The remainder of the trace shows this same behaviour of slow sending rate inflation and intermittent rate reductions that are typical of CUBIC.

This CUBIC session managed an average transfer rate of some 45 Mbps, which is well below the peak circuit capacity of 250 Mps.

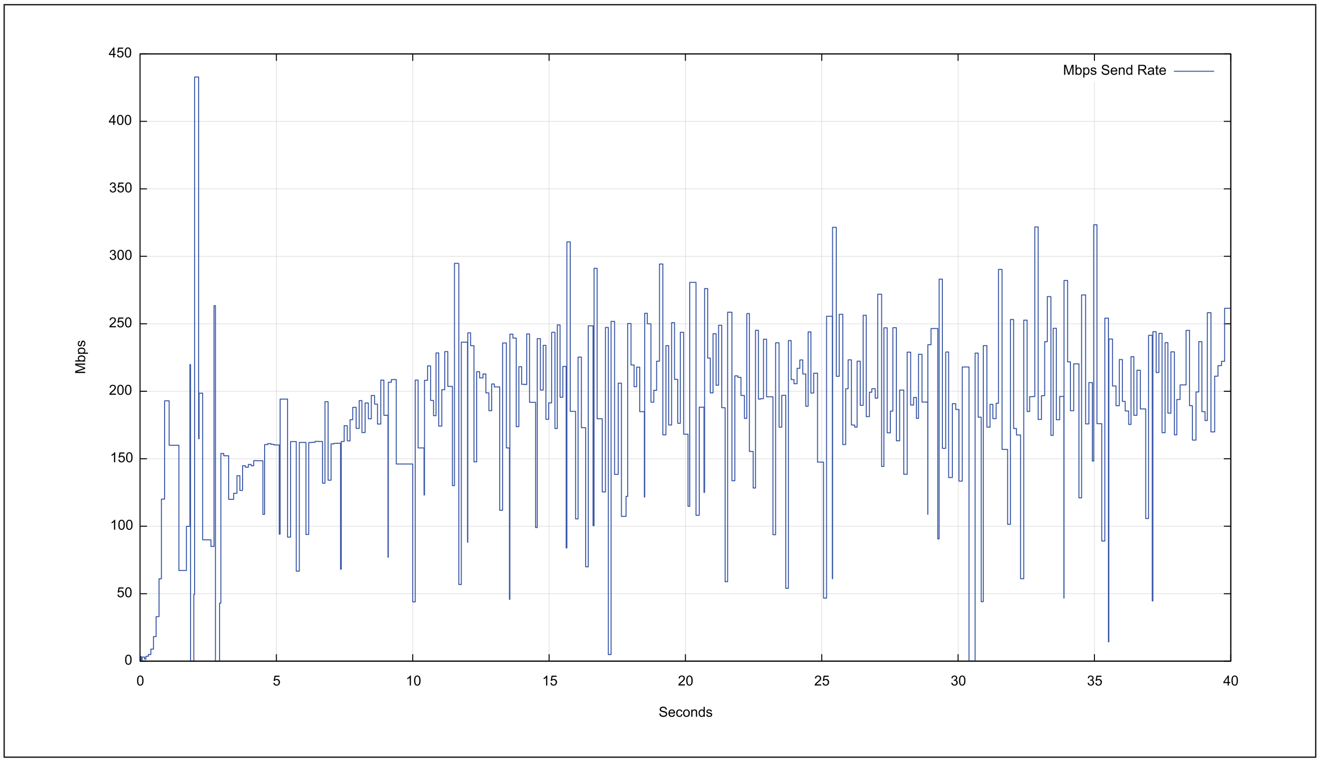
Figure 5: TCP CUBIC Over Starlink



Starlink is a shared medium, and the performance of the system in local times of light use (off peak) is significantly different from that of performance in peak times. Figure 6 shows the CUBIC performance profile during an off-peak time.

The difference between the achievable throughput between peak and off-peak times is quite significant, with the off-peak performance reaching a throughput level some 3 to 4 times greater than the peak-load performance. The slow-start phase increased the throughput to some 200 Mbps within the first second. The flow then oscillated for a second, then started a more stable congestion-avoidance behaviour by second 4. The CUBIC window inflation behaviour is visible up to second 12 and then the flow oscillates around some 200 Mbps of throughput.

Figure 6: TCP CUBIC Over Starlink – Off-Peak



Is the difference between these two profiles in Figures 5 and 6 a result of active flow management by Starlink equipment, or the result of the way in which CUBIC reaches a flow equilibrium with other concurrent flows?

We can attempt to answer this question by using a different TCP control protocol that has a completely different response to contention with other concurrent flows.

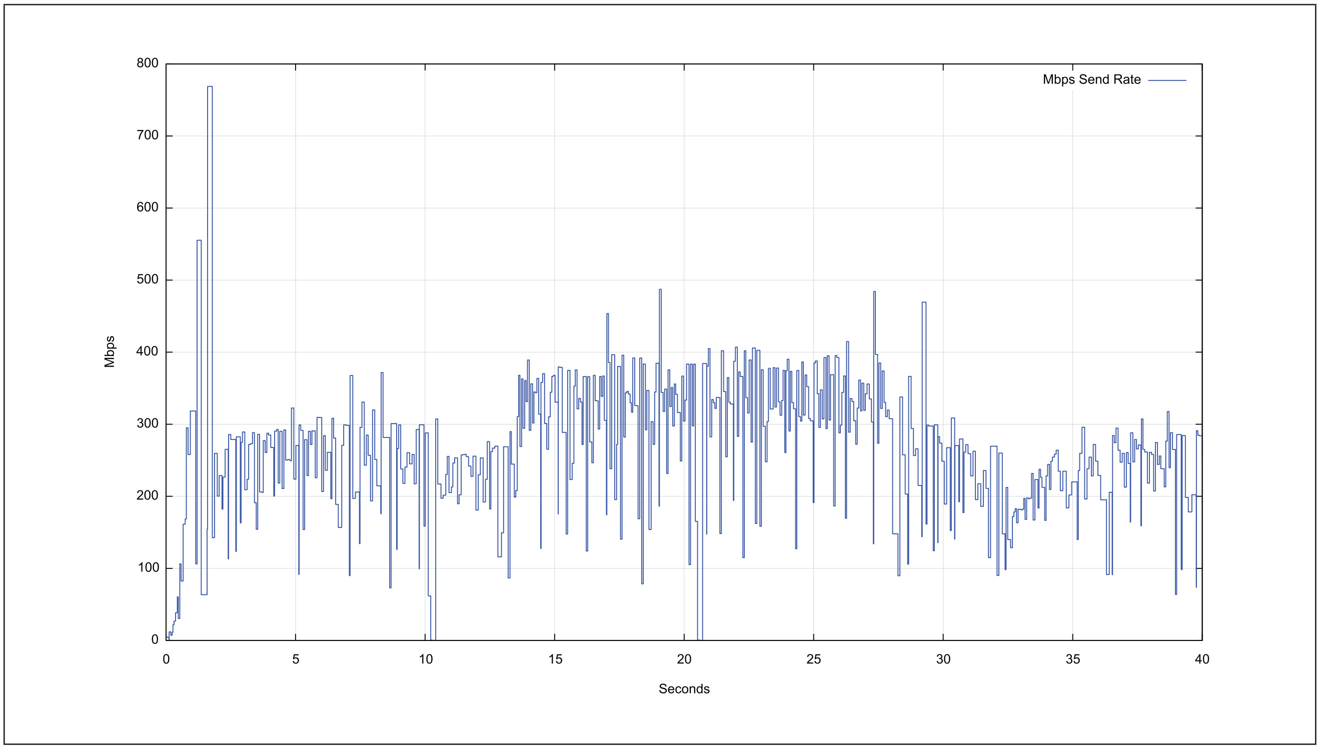
The *Bottleneck Bandwidth and Round-trip propagation time* (BBR)[9] is a TCP congestion-control algorithm developed at Google a decade ago. BBR attempts to position the TCP flow at the onset of network queue formation rather than oscillating between full and empty queue states (as is the case in loss-based congestion-control algorithms).

Briefly, BBR makes an initial estimate of the delay-bandwidth product of the network path, and then drives the sender to send at this rate for 6 successive RTT intervals. It performs repair for dropped packets without adjusting its sending rate. The 7th RTT interval sees the sending rate increase by 25%, and the end-to-end delay is carefully measured in this interval. The final RTT interval in the cycle sees the sending rate drop by 25% from the original rate, intended to drain any network queues that may have formed in the previous RTT interval. If the end-to-end delay increases in the inflate interval, the original sending rate is maintained.

If the increased sending window does not impact the end-to-end delay, it indicates that the network path has further capacity and the delay-bandwidth estimate is increased for the next 8-RTT cycle. (There have been a couple of subsequent revisions to the BBR protocol, but in this case, I'm using the original (v1) version of BBR.)

Figure 7 shows the results of a Starlink performance test using BBR.

Figure 7: TCP BBR Over Starlink



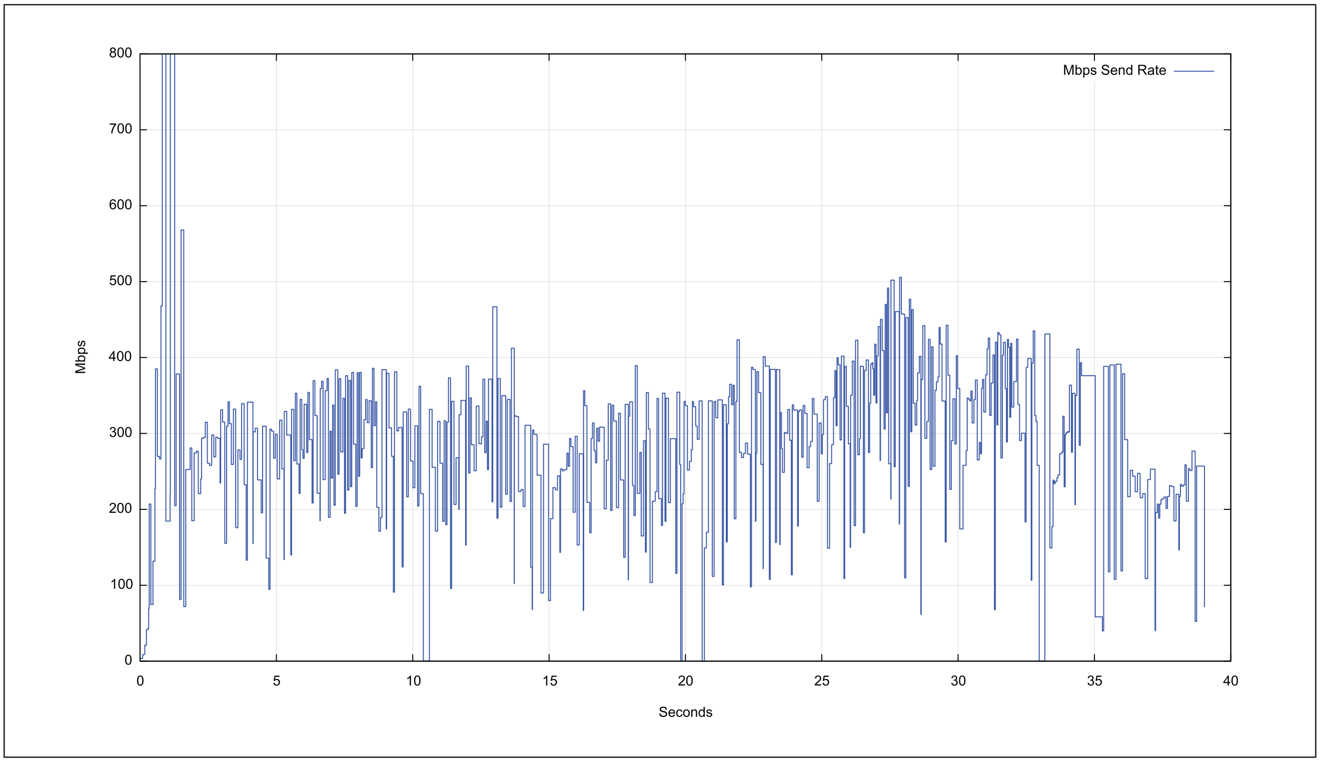
In this case, BBR has made an initial estimate of some 250 Mbps for the path bandwidth. This estimate appears to have been revised at second 14 to 350 Mbps, and then dropped to 200 Mbps 15 seconds later for the final 10 seconds of this test. It is likely that these changes are the result of BBR responding to satellite handover in Starlink.

The same BBR test was performed in an off-peak time and had a very similar outcome (Figure 8 on the following page).

If BBR is sensitive to changes in latency, and latency is so variable in Starlink, then why does BBR perform so well?

I suspect that here BBR is not taking a single latency measurement, but measuring the RTT for all packets that are sent in this 7th RTT interval, and using the minimum RTT value as the "loaded" RTT value to determine whether to perform a send-rate adjustment. As long as the minimum RTT levels are consistent, and they—as shown in Figure 3—are consistent across each 15-second scheduling interval, then BBR will set its sending rate close to the maximum sending rate that Starlink supports.

Figure 8: TCP BBR Over Starlink – Off-Peak



Protocol Tuning for Starlink

Could you tune a variant of TCP to optimise its performance over a path that includes a Starlink component?

A promising approach would appear to be a variant of BBR. The reason for the choice of BBR is its ability to maintain its sending rate in the face of individual packet-loss events. Starlink performs a satellite handover at regular 15-second intervals. If the regular sending-rate inflation in BBR occurs at the same time as scheduled satellite handover, the BBR sender could defer its rate inflation, maintaining its current sending rate across the scheduled handover.

The issue with BBR is that, for version 1 of this protocol, it is quite aggressive in claiming network resources, and this aggression can starve other concurrent sessions of capacity. One possible response is to use the same 15-second satellite handover timer with version 3 of the BBR protocol, which is intended to be less aggressive when working with concurrent data flows.

In theory, it would be possible to adjust CUBIC in a similar manner, performing a lost packet repair using *Selective Acknowledgement* (SACK)[10] if the packet loss occurred at the time of a scheduled satellite handover. While CUBIC is a fairer protocol with respect to sharing the path capacity with other concurrent sessions, it tends to react conservatively when faced with high jitter paths. Even with some sensitivity to scheduled satellite handovers, CUBIC is still prone to reduced efficiency in the use of network resources.

References and Resources

[0] Dan York and Geoff Huston, "Low Earth Orbit Satellite Systems for Internet Access," *The Internet Protocol Journal*, Volume 26, No. 2, September 2023.

[1] Starlink: **https://www.starlink.com**

[2] Isaac Newton, *Philosophiæ Naturalis Principia Mathematica*, July 1687.

[3] Mark Allman, Daniel R. Glover, and Luis A. Sanchez, "Enhancing TCP Over Satellite Channels Using Standard Mechanisms," RFC 2488, January 1999.

[4] Wikipedia, Low Earth Orbit: **https://en.wikipedia.org/wiki/Low_Earth_orbit**

[5] Speedtest: **https://www.speedtest.net**

[6] Ping network utility: **https://en.wikipedia.org/wiki/Ping_(networking_utility)**

[7] Jon Postel, "Transmission Control Protocol," RFC 793, September 1981.

[8] Sangtae Ha, Injong Rhee, and Lisong Xu, "CUBIC: a new TCP-friendly high-speed TCP variant," *ACM SIGOPS Operating Systems Review*, Volume 42, No. 5, July 2008.

[9] Neal Cardwell, Yuchung Cheng, C. Stephen Gunn, Soheil Hassas Yeganeh, and Van Jacobson, "BBR: Congestion-Based Congestion Control: Measuring bottleneck bandwidth and round-trip propagation time," *ACM Queue*, Volume 14, No. 5, October 2016.

[10] Sally Floyd, Jamshid Mahdavi, Matt Mathis, and Matthew Podolsky, "An Extension to the Selective Acknowledgement (SACK) Option for TCP," RFC 2883, July 2000.

[11] Josh Fomon, "New Speedtest Data Shows Starlink Users Love Their Provider," *Ookla Insights Articles*, May 8, 2023.

[12] Kevin Hurler, "Starlink Is Now Connecting Remote Antarctic Research Camps to the Internet," *Gizmodo*, January 23, 2023.

[13] "Perspectives on LEO Satellites: Using Low Earth Orbit Satellites for Internet Access," Internet Society, 2022.

[14] Ulrich Speidel, "Satellite still a necessity for many Pacific Islands," *APNIC Blog*, September 18, 2018.

GEOFF HUSTON AM, 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 1990s. He is author of numerous Internet-related books and was a member of the Internet Architecture Board from 1999 until 2005. He served on the Board of Trustees of the Internet Society from 1992 until 2001. At various times Geoff has worked as an Internet researcher, an ISP systems architect, and a network operator. E-mail: **gih@apnic.net**

DNS Evolution

by Geoff Huston, APNIC

The *Domain Name System* (DNS) is a crucial part of today's Internet. With the fracturing of network address space as a byproduct of IPv4 address rundown and the protracted IPv6 transition, the namespace of the Internet is now the defining attribute that makes it one network. However, the DNS is not a rigid and unchanging technology. It has changed considerably over the lifetime of the Internet, and here I'd like to look at what has changed and what has remained the same.

The Early DNS

The early Internet architecture used names as a convenient alias for an IP address. Each host used a local list of name and address pairs, and an application would look up the name in this file (**`hosts.txt`**) and use the associated address in the subsequent packet exchange. In many ways, this file was a direct analogy to the telephone directory in a telephone network.

This simple framework has one major drawback: *scalability*. As the number of connected hosts on the network increased, the burden of distributing updated copies of the name file increased and the task of maintaining loose coherence across all these local copies of this file became more challenging. The document IEN 61[1], describing an *Internet Name Server*, was released in 1978, and it appears to be a basic predecessor of today's DNS.

Some five years later, in 1983, RFC 882[2] defined a hierarchical namespace using a tree-structure name hierarchy. It also defined a name server as a service that holds information about a part of the name hierarchy, and also refers to other name servers that hold information about lower parts of the name hierarchy. The document also defined a resolver that can resolve names into their stored attributes by following referrals to find the appropriate name server to query, and then obtaining this information from the server. RFC 883[3] defined the DNS query and response protocol, a simple stateless protocol.

And that's about it.

Most of what is in today's DNS was defined in these early specifications, and what we've been doing over the intervening forty years has been filling in the details. The DNS has not really changed in any substantive manner over the intervening period.

Evolutionary Pressures

However, I think that such a perspective ignores a large body of refinement in the DNS world that has occurred. The DNS is by no means perfect; it can be extremely slow to resolve a name, and even slower to incorporate changes into the distributed data framework.

The resolution of DNS queries pays scant regard to concerns about user privacy, and any party who can observe a user's DNS query stream can readily piece together an accurate picture of the user's activities. The distributed stateless method used to resolve names is prone to various efforts to eavesdrop DNS transactions and manipulate the information being provided in DNS responses. The DNS cannot easily protect itself from disruptive attack and has been regularly used in highly effective denial-of-service attacks. It's also insecure, in that a client cannot verify the authenticity and currency of a response.

The operation of the DNS in resolving a name can be extremely opaque. The use of parallel servers and resolvers to improve the resilience of the DNS creates combinatorial explosion in the number of paths that can be used to navigate through the distributed data structure. It is not possible to tell in advance which servers may be used in the resolution of a query, or the number of additional queries a single original query may trigger. Given that resolvers can respond directly to a query with a locally cached response, it is not possible to tell in advance where the response will come from, or if the response is authentic.

For a common and fundamental service that every user not only uses, but implicitly relies upon, the DNS in practice is far from a paragon of sound operational engineering.

The evolutionary efforts have been intended to remedy some of these shortcomings, with goals to improve the speed of DNS resolution, improve aspects of privacy of DNS transactions, increase the level of trust in DNS responses, and resist efforts to subvert the integrity of DNS name-resolution transactions.

DNS Privacy

The DNS is not what you might call a discrete protocol. By default, queries are made in the clear. The IP addresses of the querier, the server being queried, and the name being queried are visible to any party that is in a position to inspect DNS traffic. These parties include not only potential eavesdroppers in the network, but also the operating system platform that hosts the application making the DNS query, the recursive resolver that receives the query, and any forwarding agent that the recursive resolver uses. Depending on the state of the local cache in the recursive resolver, the recursive resolver may need to perform some level of top-down navigation through the nameserver hierarchy, asking an authoritative server at each level the full original query name. The recursive resolver normally lists itself as the source of these queries, so the identity of the original user is occluded, but the query name is still visible.

RFC 7858 provides a specification for DNS over a *Transport Layer Security* (TLS) session (DoT)[4]. This specification allows the client and server to securely set up a shared session key that is then used to encrypt all subsequent transactions between the two parties. TLS can also be used to authenticate the server name in order to assure the client that it is connecting to an instance of the named server.

There is some overhead to setting up a TLS session, and the most efficient use of this approach is in the stub-to-recursive DNS environment where a single TLS session can be kept open and reused for subsequent queries, amortizing the initial setup overheads across these queries. The standard specification of DoT defines the use of TCP port 853, which allows an onlooker to identify that DoT is being used and identify the two end parties by their IP addresses, but not the DNS queries or responses.

Subsequent standards work has defined *DNS over QUIC* (DoQ), RFC 9250[5]. The encryption that QUIC provides has properties similar to those that TLS provides, while QUIC transport eliminates the head-of-line blocking issues inherent with TCP and provides more efficient packet-loss recovery than *User Datagram Protocol* (UDP).

In addition, it is possible to add a *Hypertext Transfer Protocol* (HTTP) wrapper to the DNS data object, defining *DNS over HTTPS* (DoH), RFC 8484[6]. DoH uses port 443, using either TCP in the case of HTTP/2 or UDP with the QUIC-based HTTP/3, so the DNS transactions would be largely indistinguishable from Web traffic. HTTP adds its own ability to perform object caching, redirection, proxying, authentication, and compression beyond that provided in the conventional DNS model, although the use of such HTTP capabilities in the DNS context is not well understood. HTTP also allows a server to push content to a client. In the DoH scenario this possibility could permit the use of queryless DNS, where the server pushes DNS responses to a client without any initial triggering DNS query.

In these approaches to encrypted transport for the DNS, the remote server is aware of the client's IP address and the queries that the client is making. In the stub-to-recursive scenario, this awareness allows the recursive resolver to be privy to the user's DNS actions, even when the network path between the two parties is secure. A stronger level of privacy is obtained by using *Oblivious DNS over HTTPS*, RFC 9230[7], where no single DNS server is simultaneously aware of the client's IP address and the content of the DNS queries. Here a double level of encryption is used in conjunction with two independent agents within the network. The client sends an encrypted DNS query to the first proxy using DoH. This proxy is aware of the client's IP identity, but is not able to decrypt the DNS query. The proxy makes its own query using the encrypted query to a separate target, again using DoH, but this time there is no record of the original client. The target can decrypt the query and function as a conventional recursive resolver.

These four specifications show that it is possible to cloak DNS transactions within a secure veil of secrecy, but it remains a topic of speculation as to the extent of uptake of these technologies. Encrypted transport sessions impose higher costs on the operation of DNS infrastructure (recursive resolvers and authoritative servers), and it is unclear how the current DNS economic models, where individual DNS queries are essentially unfunded by the client, can absorb these higher costs.

An entirely different approach to improving DNS privacy is described in DNS *Query Name Minimisation*, RFC 7816[8]. The observation is that as a recursive resolver navigates its path through the DNS hierarchy, it uses the original query name to query authoritative name servers, essentially sharing the knowledge of the name being queried with a set of servers. The rationale for this approach is that the client does not necessarily know where a zone cut may exist in advance. Query Name Minimisation proposes to minimise the amount of information being disclosed to authoritative name servers by sending a request to the nameserver authoritative for the closest known ancestor of the original query name, and asking for a *Name Server* (NS) delegation record rather than the original query type. This approach does not impose additional overheads on DNS server infrastructure. It does not offer channel security, but it does limit the amount of information "leakage" that is a feature of the DNS name-resolution process.

On a more general level, none of these DNS privacy measures can assure users of the authenticity of the DNS response that they receive. These measures limit the ability of other parties to eavesdrop on DNS queries and responses, but detecting (and presumably rejecting) DNS responses that are inauthentic is a separate issue for the DNS.

DNS Authenticity – DNSSEC

Domain Name System Security Extensions (DNSSEC) is an extension to the DNS that associates a cryptographically-generated digital signature with each record in a DNSSEC-signed zone, specified in RFC 4033[9]. DNSSEC does not change the DNS namespace, nor the DNS name-resolution protocol. Clients who are aware of DNSSEC can request that a DNS response should include a DNSSEC signature, if one is available for the zone, and may then validate the response using that signature.

You might think that a tool that allows the client to verify a DNS response would be immediately popular. If the relationship between the names that applications use and services and IP addresses that are used at the protocol level is disrupted, then users can be readily deceived. Yet, after close to three decades from its initial specification, DNSSEC is still struggling to achieve mainstream adoption. Part of the issue is that the strong binding of the DNS protocol to a UDP transport causes a set of problems when responses bloat in size because of attached signatures and keys. Another part of the issue lies in the care and attention required to manage cryptographic keys and the unforgiving nature of cryptographic validation. And a large part of the problem is that when the Web began using TLS as a means of verifying the identity of a remote server, many didn't consider any marginal incremental benefit of DNSSEC in the DNS part of session creation to be worth the incremental effort and cost of using DNSSEC.

For these reasons DNSSEC continues in the DNS environment as a "work in progress."

Evolution of Query Mechanisms

The base DNS specification uses a limited repertoire, where queries contain a query name and a query type, and, if carried over the UDP, DNS responses are limited to 512 bytes in length. The restrictions in the size of several flag fields, return codes, and label types available in the basic DNS protocol were hindering the development of DNSSEC. The chosen path to resolve this dilemma was to use a so-called *Pseudo Resource Record*, the OPT (for "options") record that is included in the additional data section of a DNS message. To ensure backward compatibility, a responder does not use the OPT record unless it was present in the query. This is the general *Extension Mechanism for DNS*, or EDNS[10].

EDNS options have been used so far to support DNSSEC functions, padding, TCP keepalive settings, and Client Subnet fields. It has also been used to extend the maximum size of UDP messages in the DNS by using a EDNS Buffer Size.

It is often desirable to separate the name of a service and the location of the service platform that delivers the service, and service record type that was intended to achieve that outcome. *Service Records*, or SRV records, can provide that form of flexibility, where the service is defined by a host name, a port identifier, and a protocol identifier, and the associated resource record provides the TCP or UDP port number and the canonical service name of the target service platform. Multiple service targets can be specified with an associated preference for use. The functional shift in the use of the SRV record was loading the DNS query with a service profile rather than a plain domain name, and in return receiving enough information to enable the user to then connect to the desired service without making further DNS queries.

This functional shift was further extended in the *Service Binding and Parameter Specification via the DNS* (SVCB and HTTPS Resource Records) specification, RFC 9460[11]. By providing more information to the client before it attempts to establish a connection, these records offer potential benefits to both performance and privacy. These enhancements represent a shift in the design approach of the DNS, where the prior use of DNS resource record types was to segment the information associated with a DNS name, so that a complete collection of information about a service name was obtained by making a set of queries. The SVCB record effectively provides an "omnibus" response to a service query, so that the client can gather sufficient information to connect to a service with a single DNS transaction.

Delegation Records

One of the fundamental parts of the DNS data structure is the *delegation record*, which passes the control of an entire subtree in the DNS hierarchy from one node to another.

While this NS record has served the DNS since its inception, it has a few limitations. The target of the delegation record is one or more DNS server names, not their IP addresses.

Conventionally the IP addresses are provided as "glue records" contained in the *Additional Section* of a DNS referral response. However, the veracity of such glue records cannot be established, and this weakness has been the focal point of numerous DNS attacks over the years. The target of a NS record cannot be a CNAME alias. The NS record is shared across both the parent and child zones, and the child zone is deemed to be authoritative for this record. The implication is that while the parent-zone name servers can (and must) respond with referral responses with this NS record, it cannot provide a DNSSEC-signed response. It is not possible to provide a DNS service profile in a referral response. If the zone authoritative servers can be accessed using an encrypted transport protocol, this capability cannot be signalled by the NS record.

Work is underway in the *Internet Engineering Task Force* (IETF) in the *DNS Delegation* (deleg) Working Group to take the existing specification of service binding mapping for DNS servers, RFC 9461[12], and see how it could be used as a more flexible delegation record that addresses some or all of these identified shortcomings in the existing NS form of delegation.

Alternate Name Systems

The Internet protocol suite can be regarded as a collection of elements, including addressing, routing, forwarding, and naming, and it's possible to substitute a different technology for one element without necessarily impacting the others. For example, the transition from IP version 4 to IP version 6 in the addressing realm does not necessitate any fundamental changes to routing, forwarding, or naming. The same can be said of the DNS name system. Alternate name systems can be used and to some extent they can coexist with the DNS.

In the traditional model of DNS resolution, users have little control over their DNS settings. Some technically literate users may choose settings that differ from the defaults, but there has been little incentive to do so, and the vast majority of users have their DNS settings configured for them by administrators via a protocol such as the *Dynamic Host Configuration Protocol* (DHCP).

Many alternative naming systems in use today come bundled with the specific applications that use them: a particular alternative naming system is often tied to a corresponding application, and this application often bypasses administrator-controlled settings and any preconfigured DNS settings. For example, the *Tor Project* uses its own naming system that bypasses traditional DNS resolution. Users can install the *Tor Browser*, and it will use the Tor naming system for names ending in **`.ONION`**, while forwarding any other names to the local DNS library. The application developer makes the choice of which naming system to use without users even knowing that they are using an alternative naming system, nor do they understand potential implications.

Various forms of experimentation have used decentralised models that eschew a single name hierarchy and allow individual names to exist in an unstructured flat namespace. The underlying registry framework that associates a name with an "owner" has often relied on some blockchain-like approach, where the association of a name and a public-key value is placed into the blockchain. Numerous such alternate name systems exist today, including the *Ethereum Name Service* (ENS), which uses so-called "smart contracts" in its blockchain, and *Unstoppable Domains*, which uses a blockchain platform but operates the namespace as a centrally operated space. The *GNU Name System* (GNS) is also a decentralised platform that offers name persistence, but it has no concept of a root zone. Instead GNS uses the concept of a "start zone" that is configured locally and determines where to begin resolution. Since local users have complete control over their own start zone, every GNS user can potentially use a different namespace. Thus, there is no guarantee that names will be globally unique, or that a given name will resolve the same for different users. The only guarantee is that users with the same start zone will have the same view of the namespace. Every unique start zone defines its own namespace. This scenario is similar in practice to DNS resolution using different root zones. The key innovation in GNS is to replace a search hierarchy with a distributed hash table that can include links to other hash tables.

Such alternate name systems interact with the existing DNS-defined namespace in a variety of ways. Some attempt to coexist with the DNS with the alternate names being some form of extension to the DNS namespace, potentially using a different name-resolution protocol. Other systems are completely self-contained and make no effort to coexist with the DNS. This situation is more commonly seen in an application-specific context where the application environment is exclusively associated with an alternate namespace.

Conclusions

Only a completely moribund technology is impervious to change! As digital technologies and services evolve, the demands placed on the associated namespaces also evolve in novel and unpredictable ways.

The DNS is an interesting case in that so far it has been able to respond to the evolving Internet without requiring fundamental changes to the structure of its namespace, the distributed information model, or the name-resolution protocol. Most of the evolutionary changes that have been folded into the DNS to date have been undertaken in a way that preserves backward compatibility, and the cohesion of the underlying namespace has been largely preserved.

However, maintaining this cohesion across the Internet is not an assured outcome for the future. The pressures to impose barriers to the access to content at national and regional levels are often expressed by imposing selective barriers to the resolution of content service names, and the DNS is left carrying the burden of supporting such selective fragmentation in the Internet.

The camel has undeniably poked its nose into the tent of name coherence in the form of EDNS *Client Subnet*[13], where the response given to a query may be dependent on who is querying, as much as the name that is being used in the query, and it's likely that this more qualified and fragmented model of a namespace will persist and support an increasingly fragmented Internet.

References and Further Reading

[1] Jon Postel, "Internet Name Server," IEN 61, October 1978.

[2] Paul Mockapetris, "Domain names: Concepts and facilities," RFC 882, November 1983.

[3] Paul Mockapetris, "Domain names: Implementation specification," RFC 883, November 1983.

[4] Zi Hu, Liang Zhu, John Heidemann, Allison Mankin, Duane Wessels, and Paul Hoffman, "Specification for DNS over Transport Layer Security (TLS)," RFC 7858, May 2016.

[5] Christian Huitema, Sara Dickinson, and Allison Mankin, "DNS over Dedicated QUIC Connections," RFC 9250, May 2022

[6] Paul Hoffman, and Patrick McManus, "DNS Queries over HTTPS (DoH)," RFC 8484, October 2018.

[7] Eric Kinnear, Patrick McManus, Tommy Pauly, Tanya Verma, and Christopher A. Wood, "Oblivious DNS over HTTPS," RFC 9230, June 2022.

[8] Stephane Bortzmeyer, "DNS Query Name Minimisation to Improve Privacy," RFC 7816, March 2016.

[9] Roy Arends, Rob Austein, Matt Larson, Dan Massey, and Scott Rose, "DNS Security Introduction and Requirements," RFC 4033, March 2005.

[10] Joao Damas, Michael Graff, and Paul Vixie, "Extension Mechanisms for DNS (EDNS(0))," RFC 6891, April 2013.

[11] Ben Schwartz, Mike Bishop, and Erik Nygren, "Service Binding and Parameter Specification via the DNS (SVCB and HTTPS Resource Records)," RFC 9460, November 2023.

[12] Ben Schwartz, "Service Binding Mapping for DNS Servers," RFC 9461, November 2023.

[13] Carlo Contavalli, Wilmer van der Gaast, David C. Lawrence, and Warren Kumari, "Client Subnet in DNS Queries," RFC 7871, May 2016.

[14] Miek Gieben, "DNSSEC: The Protocol, Deployment, and a Bit of Development," *The Internet Protocol Journal*, Volume 7, No. 2, June 2004.

[15] Richard Barnes, "Let the Names Speak for Themselves: Improving Domain Name Authentication with DNSSEC and DANE," *The Internet Protocol Journal*, Volume 15, No.1, March 2012.

[16] Geoff Huston, "A Question of DNS Protocols," *The Internet Protocol Journal*, Volume 17, No. 1, September 2014.

[17] Geoff Huston, "What's in a DNS Name?" *The Internet Protocol Journal*, Volume 19, No. 1, March 2016.

[18] Geoff Huston and Joao Luis Silva Dama, "DNS Privacy," *The Internet Protocol Journal*, Volume 20, No. 1, March 2017.

[19] Geoff Huston, "The Root of the DNS," *The Internet Protocol Journal*, Volume 20, No. 2, June 2017.

[20] Geoff Huston, "DNS Privacy and the IETF," *The Internet Protocol Journal*, Volume 22, No. 2, July 2019.

[21] Geoff Huston, "DNS Trends," *The Internet Protocol Journal*, Volume 24, No. 1, March 2021.

[22] Burton Kaliski Jr., "Minimized DNS Resolution: Into the Penumbra," *The Internet Protocol Journal*, Volume 25, No. 3, December 2022.

[23] Wikipedia article on the DNS:
https://en.wikipedia.org/wiki/Domain_Name_System

GEOFF HUSTON AM, 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 1990s. He is author of numerous Internet-related books, and was a member of the Internet Architecture Board from 1999 until 2005. He served on the Board of Trustees of the Internet Society from 1992 until 2001. At various times Geoff has worked as an Internet researcher, an ISP systems architect, and a network operator.
E-mail: **gih@apnic.net**

Check your Subscription Details!

Make sure that both your postal and e-mail addresses are up-to-date since these are the only methods by which we can contact you. If you see the words "Invalid E-mail" on your printed copy this means that we have been unable to contact you through the e-mail address on file. If this is the case, please contact us at **ipj@protocoljournal.org** with your new information. The subscription portal is located here:
https://www.ipjsubscription.org/

Fragments

An Open Letter to the United Nations
1 July 2024

Secretary-General António Guterres
and Envoy on Technology Amandeep Singh Gill,

Since its inception more than fifty years ago, the Internet's technical architecture has evolved and been collaboratively maintained through multistakeholder processes. While it was born in government laboratories, the Internet became a network of networks that kept expanding and required continuous work. Much of that was coordinated in the *Internet Engineering Task Force* (IETF)[1], an open, consensus-based, bottom-up, voluntary and global standards body.

More than thirty-five years ago, the World Wide Web was born in the laboratories of CERN. It, too, quickly evolved into a global public tool, maintained and developed by a collaboration of like-minded engineers and other stakeholders at the *World Wide Web Consortium* (W3C)[2]. It, too, is an open, bottom-up, consensus-driven, voluntary and global standards body.

The success of both IETF's and W3C's work can be measured by where the Internet is today and what it has achieved: global communication has flourished, bringing education, entertainment, information, connectivity and commerce to most of the world's population. The Internet has been a catalyst for advancing development. These communities and the way they have structured themselves have paid off.

We recognize that governments take seriously their responsibility to protect their citizens. So, as harms associated with the Internet and the Web become more apparent, there is a desire on the part of governments to act through regulation and legislation. Technical architecture can enable and influence how the Internet is used, but on its own it cannot address abuse, misinformation, inequality, or many other issues. There is nevertheless a potential danger in regulation and legislation, if it undermines the fundamentally empowering nature of the Internet.

The Internet is an unusual technology because it is fundamentally distributed. It is built up from all of the participating networks. Each network participates for its own reasons according to its own needs and priorities. And this means, necessarily, that there is no center of control on the Internet. This feature is an essential property of the Internet, and not an accident. Yet over the past few years we have noticed a willingness to address issues on the Internet and Web by attempting to insert a hierarchical model of governance over technical matters. Such proposals concern us because they represent an erosion of the basic architecture.

In particular, some proposals for the *Global Digital Compact* (GDC)[3] can be read to mandate more centralized governance. If the final document contains such language, we believe it will be detrimental to not only the Internet and the Web, but also to the world's economies and societies.

Furthermore, we note that the GDC is being developed in a multilateral process between states, with very limited application of the open, inclusive and consensus-driven methods by which the Internet and Web have been developed to date. Beyond some high-level consultations, non-government stakeholders (including Internet technical standards bodies and the broader technical community) have had only weak ways to participate in the GDC process. We are concerned that the document will be largely a creation only of governments, disconnected from the Internet and the Web as people all over the world currently experience them.

Therefore, we ask that member states, the Secretary-General and the Tech Envoy seek to ensure that proposals for digital governance remain consistent with the enormously successful multistakeholder Internet governance practice that has brought us the Internet of today. Government engagement in digital and Internet governance is needed to deal with many abuses of this global system but it is our common responsibility to uphold the bottom-up, collaborative and inclusive model of Internet governance that has served the world for the past half century.

Signed,

All signatures are in a personal capacity; affiliations are informational only.

Daniel Appelquist, W3C TAG co-chair
David Baron, former W3C TAG
Hadley Beeman, W3C TAG
Robin Berjon, former W3C TAG; former W3C HTML Activity Lead
Andrew Betts, former W3C TAG
Sir Tim Berners-Lee, inventor of the World Wide Web; founder & emeritus director, W3C
Tim Bray, former W3C TAG; Editor of XML (W3C), JSON (IETF)
Randy Bush, former IESG, former ISO/WG13
Dr. Brian E. Carpenter, former Group Leader, Communication Systems, CERN; former IAB chair; former ISOC BoT chair; former IETF chair
Vint Cerf, Internet Pioneer
David Conrad, former IANA general manager; former ICANN CTO
Martin Duke, former IESG
Dr. Lars Eggert, former IETF chair; former IRTF chair
David Jack Farber, former IAB; former ISOC BoT; former Chief Technologist USA FCC
Dr. Stephen Farrell, Trinity College Dublin; former IESG; former IAB
Demi Getschko, .br
Christian Huitema, former IAB chair
Geoff Huston, former ISOC BoT chair; former IAB
Erik Kline, IESG
Mallory Knodel, former IAB
Olaf Kolkman, former IAB chair
Konstantinos Komaitis, senior resident fellow, Internet Governance lead, Democracy and Tech Initiative, Atlantic Council
Chris Lilley, W3C Technical Director; former W3C TAG
Peter Linss, W3C TAG co-chair
Sangwhan Moon, former W3C TAG
Jun Murai, former IAB; WIDE Project founder; former W3C steering committee; former ISOC BoT
Mark Nottingham, former IAB; former W3C TAG
Lukasz Olejnik, former W3C TAG
Colin Perkins, IRTF chair
Pete Resnick, former IAB; former IESG
Alex Russell, former W3C TAG
Peter Saint-Andre, former IESG
David Schinazi, IAB
Melinda Shore, IRSG; former IAB
Robert Sparks, former IAB; former IESG
Lynn St. Amour, former Internet Society President and CEO; former UN IGF Multistakeholder Advisory Group chair
Andrew Sullivan, former IAB chair
Martin Thomson, W3C TAG; former IAB
Brian Trammell, IRSG; former IAB
Léonie Watson, W3C Web Applications Working Group Chair
Paul Wouters, IESG

References and Acronym Expansions

[1] Internet Engineering Task Force (IETF): **`https://ietf.org/`**

[2] World Wide Web Consortium (W3C): **`https://w3.org/`**

[3] Global Digital Compact (GDC): **`https://www.un.org/techenvoy/global-digital-compact`**

[4] Internet Architecture Board (IAB): **`https://iab.org/`**

[5] Internet Engineering Steering Group (IESG): **`https://iesg.org/`**

[6] Internet Research Steering Group (IRSG): **`https://www.irtf.org/irsg.html`**

[7] Internet Research Task Force (IRTF): **`https://www.irtf.org/`**

[8] ISOC BoT: Internet Society Board of Trustees: **`https://www.internetsociety.org/board-of-trustees/`**

[9] World Wide Web Consortium Technical Architecture Group (W3C TAG): **`https://w3ctag.org/`**

[10] Originally posted here: **`https://open-internet-governance.org/letter`**

Call for Papers: IAB Workshop on AI-Control

The *Internet Architecture Board* (IAB) is planning a workshop to explore practical opt-out mechanisms for *Artificial Intelligence* (AI), and build an understanding of use cases, requirements, and other considerations in this space. The workshop will be held in September 2024 in the Washington, DC area. Exact dates and location to be confirmed soon. The deadline for submissions is August 2nd, 2024 and invitations will be issued by August 15th, 2024.

Large Language Models (LLM) and other machine learning techniques require voluminous input data, and one common source of such data is the Internet—usually, "crawling" Web sites for publicly available content, much in the same way that search engines crawl the Web. This similarity has led to an emerging practice of allowing the *Robots Exclusion Protocol*, defined in RFC 9309, to control the behavior of AI-oriented crawlers.

This emerging practice raises many design and operational questions. It is not yet clear whether **`robots.txt`** (the mechanism specified by RFC 9309) is well-suited to controlling AI crawlers. A content creator or host may not be able to distinguish a crawler used for search indexing from a crawler used for LLM ingest—and indeed some crawlers may be used for both purposes. Potential use cases may extend across many different units of content, policies to be signaled, and types of content creators. Before **`robots.txt`** becomes a de facto solution to AI crawling opt-out, it is necessary to examine whether it is an appropriate mechanism: in particular, whether the creator of a particular unit of content can realistically and fully exercise their right to opt-out, and the scope of data ingest to which that opt-out applies.

This workshop aims to explore practical opt-out mechanisms for AI, and build an understanding of use cases, requirements, and other considerations in this space. It will focus on mechanisms to communicate the opt-out choice and their associated data models. Technical enforcement of opt-out signals is not in scope. The IAB is looking for short position papers on the following topics; however, this list is non-exhaustive and should be interpreted broadly:

- User stories, use cases, and requirements for opting content out of inclusion in large language models, from a variety of sources including but not limited to the Web
- Interactions between opt-out mechanisms and different use cases for AI
- Advantages and/or deficiencies of reusing robots.txt for controlling AI crawlers on the Web
- Comparisons of use cases for crawling opt-out
- Desired properties of an AI opt-out mechanism
- Potential developments in AI that may require adjustments in opt-out mechanisms
- Implications of legal/policy frameworks (for example, copyright, privacy, research ethics) and requirements on the design of opt-out mechanisms
- Evolution of opt-out signals

Because `robots.txt` is emerging as a solution in this space, the discussion will be anchored on it as a starting point, but not limited to that mechanism. Proposals for alternative solutions may be made, but time will not be available for a detailed presentation or discussion.

Interested participants are invited to submit position papers on the workshop topics. Participants can choose their preferred format, including Internet-Drafts, text- or Word-based documents, or papers formatted similar as used by academic publication venues. Submission as PDF is preferred. Paper size is not limited, but brevity is encouraged. By default, submissions that are considered relevant will be published on the workshop website. If you wish for your submission to be anonymous or withheld from such publication, please indicate that clearly in the submission. The organizers will issue invitations based on the submissions received. Sessions will be organized according to the submissions received, and not every accepted submission or invited attendee will have an opportunity to present; the intent is to foster an active discussion and not simply to have a sequence of presentations.

Discussion at the workshop will be held under *Chatham House Rule*, and therefore will not be recorded or minuted. However, a workshop report will be published afterwards.

It is anticipated that the workshop report will include:

- A list of participants (unless they request to be withheld)
- Documentation of use cases and requirements discussed
- Recommendations for IETF standards work to be considered (if any)
- Recommendations for non-IETF standards work to be considered (if any)

The workshop will be by invitation only. Those wishing to attend should submit a position paper to **`ai-control-workshop-pc@iab.org`**. Position papers from those not planning to attend the workshop themselves are also encouraged. Feel free to contact the Program Committee with any further questions: **`ai-control-workshop-pc@iab.org`**.

For more information, visit:
`https://datatracker.ietf.org/group/aicontrolws/about/`

Our Privacy Policy

The *General Data Protection Regulation* (GDPR) is a regulation for data protection and privacy for all individual citizens of the *European Union* (EU) and the *European Economic Area* (EEA). Its implementation in May 2018 led many organizations worldwide to post or update privacy statements regarding how they handle information collected in the course of business. Such statements tend to be long and include carefully crafted legal language. We realize that we may need to provide similar language on our website and in the printed edition, but until such a statement has been developed here is an explanation of how we use any information you have supplied relating to your subscription:

- The mailing list for *The Internet Protocol Journal* (IPJ) is entirely "opt in." We never have and never will use mailing lists from other organizations for any purpose.
- You may unsubscribe at any time using our online subscription system or by contacting us via e-mail. We will honor any request to remove your name and contact information from our database.
- We will use your contact information only to communicate with you about your subscription; for example, to inform you that a new issue is available, that your subscription needs to be renewed, or that your printed copy has been returned to us as undeliverable by the postal authorities.
- We will never use your contact information for any other purpose or provide the subscription list to any third party other than for the purpose of distributing IPJ by post or by electronic means.
- If you make a donation in support of the journal, your name will be listed on our website and in print unless you tell us otherwise.

Thank You!

Publication of IPJ is made possible by organizations and individuals around the world dedicated to the design, growth, evolution, and operation of the global Internet and private networks built on the Internet Protocol. The following individuals have provided support to IPJ. You can join them by visiting **`http://tinyurl.com/IPJ-donate`**

Kjetil Aas
Fabrizio Accatino
Michael Achola
Martin Adkins
Melchior Aelmans
Christopher Affleck
Scott Aitken
Jacobus Akkerhuis
Antonio Cuñat Alario
William Allaire
Nicola Altan
Shane Amante
Marcelo do Amaral
Matteo D'Ambrosio
Selva Anandavel
Jens Andersson
Danish Ansari
Finn Arildsen
Tim Armstrong
Richard Artes
Michael Aschwanden
David Atkins
Jac Backus
Jaime Badua
Bent Bagger
Eric Baker
Fred Baker
Santosh Balagopalan
William Baltas
David Bandinelli
A C Barber
Benjamin Barkin-Wilkins
Feras Batainah
Michael Bazarewsky
David Belson
Richard Bennett
Matthew Best
Hidde Beumer
Pier Paolo Biagi
Arturo Bianchi
John Bigrow
Orvar Ari Bjarnason
Tyson Blanchard
Axel Boeger
Keith Bogart
Mirko Bonadei
Roberto Bonalumi
Lolke Boonstra
Cente Cornelis Boot
Julie Bottorff Photography
Gerry Boudreaux
Leen de Braal
Kevin Breit
Thomas Bridge
Ilia Bromberg
Lukasz Bromirski
Václav Brožík
Christophe Brun
Gareth Bryan
Ron Buchalski
Paul Buchanan
Stefan Buckmann
Caner Budakoglu
Darrell Budic
BugWorks
Scott Burleigh
Chad Burnham
Randy Bush
Colin Butcher
Jon Harald Bøvre
Olivier Cahagne
Antoine Camerlo
Tracy Camp
Brian Candler
Fabio Caneparo
Roberto Canonico
David Cardwell
Richard Carrara
John Cavanaugh
Lj Cemeras
Dave Chapman
Stefanos Charchalakis
Molly Cheam
Pierluigi Checchi
Greg Chisholm
David Chosrova
Marcin Cieslak
Lauris Cikovskis
Brad Clark
Narelle Clark
Horst Clausen
James Cliver
Guido Coenders
Robert Collet
Joseph Connolly
Steve Corbató
Brian Courtney
Beth and Steve Crocker
Dave Crocker
Kevin Croes
John Curran
André Danthine
Morgan Davis
Jeff Day
Fernando Saldana Del Castillo
Rodolfo Delgado-Bueno
Julien Dhallenne
Freek Dijkstra
Geert Van Dijk
David Dillow
Richard Dodsworth
Ernesto Doelling
Michael Dolan
Eugene Doroniuk
Michael Dragone
Joshua Dreier
Lutz Drink
Aaron Dudek
Dmitriy Dudko
Andrew Dul
Joan Marc Riera Duocastella
Pedro Duque
Holger Durer
Karlheinz Dölger
Mark Eanes
Andrew Edwards
Peter Robert Egli
George Ehlers
Peter Eisses
Torbjörn Eklöv
Y Ertur
ERNW GmbH
ESdatCo
Steve Esquivel
Jay Etchings
Mikhail Evstiounin
Bill Fenner
Paul Ferguson
Ricardo Ferreira
Kent Fichtner
Ulrich N Fierz
Armin Fisslthaler
Michael Fiumano
The Flirble Organisation
Jean-Pierre Forcioli
Gary Ford
Susan Forney
Christopher Forsyth
Andrew Fox
Craig Fox
Fausto Franceschini
Erik Fredriksson
Valerie Fronczak
Tomislav Futivic
Laurence Gagliani
Edward Gallagher
Andrew Gallo
Chris Gamboni
Xosé Bravo Garcia
Osvaldo Gazzaniga
Kevin Gee
Rodney Gehrke
Radu Cristian Gheorghiu
Greg Giessow
John Gilbert
Serge Van Ginderachter
Greg Goddard
Tiago Goncalves
Ron Goodheart
Octavio Alfageme Gorostiaga
Barry Greene
Jeffrey Greene
Richard Gregor
Martijn Groenleer
Geert Jan de Groot
Ólafur Guðmundsson
Christopher Guemez
Gulf Coast Shots
Sheryll de Guzman
Rex Hale
Jason Hall
James Hamilton
Darow Han
Handy Networks LLC
Stephen Hanna
Martin Hannigan
John Hardin
David Harper
Edward Hauser
David Hauweele
Marilyn Hay
Headcrafts SRLS
Hidde van der Heide
Johan Helsingius
Robert Hinden
Michael Hippert
Damien Holloway
Alain Van Hoof
Edward Hotard
Bill Huber
Hagen Hultzsch
Kauto Huopio
Asbjørn Højmark
Kevin Iddles
Mika Ilvesmaki
Karsten Iwen
Joseph Jackson
David Jaffe
Ashford Jaggernauth
Thomas Jalkanen
Jozef Janitor
Martijn Jansen
John Jarvis
Dennis Jennings
Edward Jennings
Aart Jochem
Nils Johansson
Brian Johnson
Curtis Johnson
Richard Johnson
Jim Johnston
Jose Enrique Diaz Jolly
Jonatan Jonasson
Daniel Jones
Gary Jones
Jerry Jones
Michael Jones
Amar Joshi
Javier Juan
David Jump
Anders Marius Jørgensen
Merike Kaeo
Andrew Kaiser
Vladislav Kalinovsky
Naoki Kambe
Akbar Kara
Christos Karayiannis
Daniel Karrenberg
David Kekar
Stuart Kendrick
Robert Kent
Thomas Kernen
Jithin Kesavan
Jubal Kessler
Shan Ali Khan
Nabeel Khatri
Dae Young Kim
William W. H. Kimandu
John King
Russell Kirk
Gary Klesk
Anthony Klopp
Henry Kluge
Michael Kluk
Andrew Koch
Ia Kochiashvili
Carsten Koempe
Richard Koene
Alexader Kogan
Matthijs Koot
Antonin Kral
Robert Krejčí
John Kristoff
Terje Krogdahl
Bobby Krupczak
Murray Kucherawy
Warren Kumari
George Kuo
Dirk Kurfuerst
Mathias Körber
Darrell Lack
Andrew Lamb
Richard Lamb
Yan Landriault
Edwin Lang
Sig Lange
Markus Langenmair

Fred Langham
Tracy LaQuey Parker
Christian de Larrinaga
Alex Latzko
Jose Antonio Lazaro Lazaro
Antonio Leding
Rick van Leeuwen
Simon Leinen
Robert Lewis
Christian Liberale
Martin Lillepuu
Roger Lindholm
Link Light Networks
Art de Llanos
Mike Lochocki
Chris and Janet Lonvick
Mario Lopez
Sergio Loreti
Eric Louie
Adam Loveless
Josh Lowe
Guillermo a Loyola
Hannes Lubich
Dan Lynch
David MacDuffie
Sanya Madan
Miroslav Madić
Alexis Madriz
Carl Malamud
Jonathan Maldonado
Michael Malik
Tarmo Mamers
Yogesh Mangar
John Mann
Bill Manning
Diego Mansilla
Harold March
Vincent Marchand
Normando Marcolongo
Gabriel Marroquin
David Martin
Jim Martin
Ruben Tripiana Martin
Timothy Martin
Carles Mateu
Juan Jose Marin Martinez
Ioan Maxim
David Mazel
Miles McCredie
Gavin McCullagh
Brian McCullough
Joe McEachern
Alexander McKenzie
Jay McMaster
Mark Mc Nicholas
Olaf Mehlberg
Carsten Melberg
Kevin Menezes
Bart Jan Menkveld
Sean Mentzer
Eduard Metz
William Mills
David Millsom
Desiree Miloshevic
Joost van der Minnen
Thomas Mino
Rob Minshall
Wijnand Modderman-Lenstra
Mohammad Moghaddas
Charles Monson
Andrea Montefusco
Fernando Montenegro
Roberto Montoya
Joel Moore
Joseph Moran
John More
Maurizio Moroni
Brian Mort
Soenke Mumm
Tariq Mustafa
Stuart Nadin
Michel Nakhla
Mazdak Rajabi Nasab
Krishna Natarajan
Naveen Nathan
Darryl Newman
Mai Nguyen
Thomas Nikolajsen
Paul Nikolich
Travis Northrup
Marijana Novakovic
David Oates
Ovidiu Obersterescu
Jim Oplotnik
Tim O'Brien
Mike O'Connor
Mike O'Dell
John O'Neill
Carl Önne
Packet Consulting Limited
Carlos Astor Araujo Palmeira
Gordon Palmer
Alexis Panagopoulos
Gaurav Panwar
Chris Parker
Alex Parkinson
Craig Partridge
Manuel Uruena Pascual
Ricardo Patara
Dipesh Patel
Dan Paynter
Leif Eric Pedersen
Rui Sao Pedro
Juan Pena
Luis Javier Perez
Chris Perkins
Michael Petry
Alexander Peuchert
David Phelan
Harald Pilz
Derrell Piper
Rob Pirnie
Jorge Ivan Pincay Ponce
Marc Vives Piza
Victoria Poncini
Blahoslav Popela
Andrew Potter
Ian Potts
Eduard Llull Pou
Tim Pozar
David Preston
David Raistrick
Priyan R Rajeevan
Balaji Rajendran
Paul Rathbone
William Rawlings
Mujtiba Raza Rizvi
Bill Reid
Petr Rejhon
Robert Remenyi
Rodrigo Ribeiro
Glenn Ricart
Justin Richards
Rafael Riera
Mark Risinger
Fernando Robayo
Michael Roberts
Gregory Robinson
Ron Rockrohr
Carlos Rodrigues
Magnus Romedahl
Lex Van Roon
Marshall Rose
Alessandra Rosi
David Ross
William Ross
Boudhayan Roychowdhury
Carlos Rubio
Rainer Rudigier
Timo Ruiter
RustedMusic
Babak Saberi
George Sadowsky
Scott Sandefur
Sachin Sapkal
Arturas Satkovskis
PS Saunders
Richard Savoy
John Sayer
Phil Scarr
Gianpaolo Scassellati
Elizabeth Scheid
Jeroen Van Ingen Schenau
Carsten Scherb
Ernest Schirmer
Benson Schliesser
Philip Schneck
James Schneider
Peter Schoo
Dan Schrenk
Richard Schultz
Timothy Schwab
Roger Schwartz
SeenThere
Scott Seifel
Paul Selkirk
Andre Serralheiro
Yury Shefer
Yaron Sheffer
Doron Shikmoni
Tj Shumway
Jeffrey Sicuranza
Thorsten Sideboard
Greipur Sigurdsson
Fillipe Cajaiba da Silva
Andrew Simmons
Pradeep Singh
Henry Sinnreich
Geoff Sisson
John Sisson
Helge Skrivervik
Terry Slattery
Darren Sleeth
Richard Smit
Bob Smith
Courtney Smith
Eric Smith
Mark Smith
Tim Sneddon
Craig Snell
Job Snijders
Ronald Solano
Asit Som
Ignacio Soto Campos
Evandro Sousa
Peter Spekreijse
Thayumanavan Sridhar
Paul Stancik
Ralf Stempfer
Matthew Stenberg
Martin Štěpánek
Adrian Stevens
Clinton Stevens
John Streck
Martin Streule
David Strom
Colin Strutt
Viktor Sudakov
Edward-W. Suor
Vincent Surillo
Terence Charles Sweetser
T2Group
Roman Tarasov
David Theese
Rabbi Rob and Lauren Thomas
Douglas Thompson
Kerry Thompson
Lorin J Thompson
Fabrizio Tivano
Peter Tomsu Fine Art Photography
Joseph Toste
Rey Tucker
Sandro Tumini
Angelo Turetta
Brian William Turnbow
Michael Turzanski
Phil Tweedie
Steve Ulrich
Unitek Engineering AG
John Urbanek
Martin Urwaleck
Betsy Vanderpool
Surendran Vangadasalam
Ramnath Vasudha
Randy Veasley
Philip Venables
Buddy Venne
Alejandro Vennera
Luca Ventura
Scott Vermillion
Tom Vest
Peter Villemoes
Vista Global Coaching & Consulting
Dario Vitali
Rüdiger Volk
Jeffrey Wagner
Don Wahl
Michael L Wahrman
Lakhinder Walia
Laurence Walker
Randy Watts
Andrew Webster
Jd Wegner
Tim Weil
Westmoreland Engineering Inc.
Rick Wesson
Peter Whimp
Russ White
Jurrien Wijlhuizen
Joseph Williams
Derick Winkworth
Pindar Wong
Makarand Yerawadekar
Phillip Yialeloglou
Janko Zavernik
Bernd Zeimetz
Muhammad Ziad Ziayuddin
Tom Zingale
Matteo Zovi
Jose Zumalave
Romeo Zwart
廖明沂.

Follow us on X and Facebook @protocoljournal https://www.facebook.com/newipj

Call for Papers

The *Internet Protocol Journal* (IPJ) is a quarterly technical publication containing tutorial articles ("What is...?") as well as implementation/operation articles ("How to..."). The journal provides articles about all aspects of Internet technology. IPJ is not intended to promote any specific products or services, but rather is intended to serve as an informational and educational resource for engineering professionals involved in the design, development, and operation of public and private internets and intranets. In addition to feature-length articles, IPJ contains technical updates, book reviews, announcements, opinion columns, and letters to the Editor. Topics include but are not limited to:

- Access and infrastructure technologies such as: Wi-Fi, Gigabit Ethernet, SONET, xDSL, cable, fiber optics, satellite, and mobile wireless.
- Transport and interconnection functions such as: switching, routing, tunneling, protocol transition, multicast, and performance.
- Network management, administration, and security issues, including: authentication, privacy, encryption, monitoring, firewalls, troubleshooting, and mapping.
- Value-added systems and services such as: Virtual Private Networks, resource location, caching, client/server systems, distributed systems, cloud computing, and quality of service.
- Application and end-user issues such as: E-mail, Web authoring, server technologies and systems, electronic commerce, and application management.
- Legal, policy, regulatory and governance topics such as: copyright, content control, content liability, settlement charges, resource allocation, and trademark disputes in the context of internetworking.

IPJ will pay a stipend of US$1000 for published, feature-length articles. For further information regarding article submissions, please contact Ole J. Jacobsen, Editor and Publisher. Ole can be reached at **ole@protocoljournal.org** or **olejacobsen@me.com**

Supporters and Sponsors

Supporters

Diamond Sponsors

Your logo here!

Ruby Sponsors

Sapphire Sponsors

Your logo here!

Emerald Sponsors

Corporate Subscriptions

For more information about sponsorship, please contact **sponsor@protocoljournal.org**

The Internet Protocol Journal
Link Fulfillment
7650 Marathon Dr., Suite E
Livermore, CA 94550

CHANGE SERVICE REQUESTED

The Internet Protocol Journal is published quarterly and supported by the Internet Society and other organizations and individuals around the world dedicated to the design, growth, evolution, and operation of the global Internet and private networks built on the Internet Protocol.

Email: ipj@protocoljournal.org
Web: www.protocoljournal.org