Workgroup: Internet Engineering Task Force
Internet-Draft:
draft-many-deepspace-ip-assessment-00
Published: 8 September 2023
Intended Status: Informational
Expires: 11 March 2024
Authors: M. Blanchet
Viagenie
C. Huitema
Private Octopus Inc.
D. Bogdanovic
AlefEdge, Inc

Revisiting the Use of the IP Protocol Stack in Deep Space: Assessment and Possible Solutions

Abstract

Deep space communications involve long delays (e.g., Earth to Mars is 4-20 minutes) and intermittent communications, because of orbital dynamics. Up to now, communications have been done on a layer-2 point to point basis, with sometimes the use of relays, therefore no layer-3 networking was possible. RFC4838 reports an assessment done around 25 years ago concluding that the IP protocol stack was not suitable for deep space networking. This result lead to the definition of a new protocol stack based on a store-and-forward paradigm implemented in the Bundle Protocol(BP). More recently, space agencies are planning to deploy IP networks on celestial bodies, such as Moon or Mars, ground, and vicinity. This document revisits the initial assessment of rejecting IP and provides solution paths to use the IP protocol stack, from IP forwarding to transport to applications to network management, in deep space communications.

Table of Contents

1. Introduction

Deep space communications involve long delays (e.g., Earth to Mars is 4-20 minutes) and intermittent communications, because of orbital dynamics. Up to now, communications have been done on a layer-2 point to point basis, with sometimes the use of relays, therefore no layer-3 networking was possible. [RFC4838] reports an assessment done around 25 years ago concluding that the IP protocol stack was not suitable for deep space networking. This result lead to the

definition of a new protocol stack based on a store-and-forward paradigm implemented in the Bundle Protocol(BP) [RFC9171] and its various components, such as convergence-layer adapters([RFC9174], [RFC7122]) and BP Security(BPSEC)[RFC9172].

More recently, space agencies are planning to deploy IP networks on celestial bodies, such as Moon[ioag] or Mars[ioag-mars], ground, and vicinity, using layer2 technologies such as WIFI or 5G.

This document revisits the initial assessment of rejecting IP and provide solution paths to use IP in deep space communications. IP in deep space means running IP over deep space layer-2 links, a reliable transport over IP, applications protocols over that transport and applying proper routing, security and network management on that IP network. Reusing the whole IP stack in deep space enables the reuse of all protocols, tools and software currently used on Internet. However, as one might already argue, most of the IP stack can not be used as is and therefore requires careful configuration and possibly some protocol changes that are discussed in this document.

The examplary network for this document is where deep space links are using IP over CCSDS space links[IPoverCCSDSSpaceLinks] and that on and around a celestial body, a connected network is established with local network infrastructure and services.

The keyword Delay-Tolerant Networking (DTN), also expanded to Delay and Disruption-Tolerant Networking, has been used to identify the problem space and given that up to now, the solution was based on the Bundle protocol, DTN was also associated with Bundle protocol. This document tries to solve the DTN problem using the Internet Protocol stack. Therefore, in this document, the DTN keyword is used to name the problem space, not the Bundle protocol solution.

This document covers more topics than what may need to be discussed or standardized in IETF, but the intent is to help answer many questions raised while looking at the whole problem space, and, in this context, provides an non-exhaustive list of topics that needs to be addressed.

Since Moon is a few light seconds away from Earth, it is possible to somewhat configure and run various IP based protocols and applications to make it "work". Mars with a much longer delay is more difficult. Therefore, this document uses Mars as the base example, knowing that if it works for Mars, a much harder problem, it could be replicated easily for Moon, or for other networks made with relays around a celestial body. This framework shall also work for longer delays, such as reaching Jupiter or the whole Solar System Internet(SSI), but it is not specifically discussed. This document uses "deep space" extensively in order to differentiate with "space"

which often includes Earth orbiting communications, which is not discussed in this document.

It should also be noted that DTN and BP were also designed for non-space use cases. While this document focuses on the deep space use case, it shall work for the other use cases of BP, but no work or discussion on these other use cases is provided in this document.

Space missions are typically planned many years in advance and are long-lived, spanning over many years even decades. Spacecrafts are controlled from Earth and therefore should always be manageable from Earth. Given the remoteness and the difficulty to physically access the spacecraft, software upgrades and configuration changes are avoided whenever possible.

As with Bundle protocol, this framework proposes to use IP in deep space with the same store-and-forward paradigm. Therefore, the IP layer has to deal with the fact that a destination may not be currently reachable and that IP packets should be stored for an unusual amount of time, such as minutes or hours or days, in the forwarding device waiting for a new link up opportunity. The transport layer should be able to work with long and variable delays, including intermittent communications. The application protocols and application themselves should be properly set to wait a longer time than on Internet to receive a response to a query. Finally, all network services such as routing, security, naming and network management should also be adapted in this new context. This document is structured around these layers.

In a nutshell, this framework is based on the following main pillars: the storage of IP packets in intermediary nodes while the destination is unreachable, the use of the QUIC transport[RFC9000] with proper configuration, the predominance of using HTTP protocol (over QUIC) for applications, and considerations related to time in all levels of the stack.

1.1. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

1.2. Document and Discussion location

The source of this document is located at https://github.com/marcblanchet/draft-deepspace-ip-assessment. Comments or changes are welcomed as a PR or an issue.

This subject should be discussed on the deepspace@ietf.org mailing list.

2. IP forwarding

In the context of deep space, an IP packet would need to be stored temporarily over some possible longer period than typical Internet when the next hop is currently unreachable or undefined, for example due to orbital dynamics. Therefore, a new queueing discipline might be needed to store packets in this context, that might be implemented as a deep queue with active queue management(AQM)[RFC7567]. Bidirectional Forwarding Detection(BFD)[RFC5880] with large timers should be considered. When the link to the next hop is up, maybe minutes or hours later, forwarding tables would be updated and stored packets would be forwarded on the link for appropriate destinations. This is being discussed in the Time-Variant Routing(TVR) working group [TVRWG] and one part is implemented as a YANG model [I-D.qu-tvr-schedule-yang].

The store-and-forward paradigm, either implemented in BP or IP or at higher layers, requires proper sizing of memory and storage at each forwarding node for the target deployment and usage.

Store and Forward policies shall be defined and implemented to cover cases such as when storage is full and new packets are received or which priorities should be given to packets when link becomes up. An example is described in [I-D.blanchet-tvr-forwarding].

Various IP stack kernel buffers used for example for reassembly queues need to be properly sized for the target usage.

Storing IP packets, or BP bundles, may in fact make the buffer bloat issue[buffer-bloat] a much bigger problem if the management of stored packets is not properly implemented with the right queueing discipline.

3. IP Routing

Given the relative static nature of space networks at least for the forseeable future, e.g., new nodes or routers are not often added or deleted in the network, use of static routes configured based on contact plan schedules ([I-D.blanchet-tvr-contactplan], [I-D.qu-tvr-schedule-yang] may be sufficient in the short term.

If run over space links, IGPs such as OSPFv3 [RFC5340] have time-related configurable parameters such as RxmtInterval, InfTransDelay, HelloInterval, RouterDeadInterval. There are 16 bits values in seconds, which means a maximum of ~18 hours. This maximum may be too small depending on the contact plan schedule. Time-Variant Routing(TVR) working group [TVRWG] is looking at this problem space.

Given that it is likely that multiple network operators will be present on celestial bodies, it is expected that BGP [RFC4271] would be used.

4. QUIC Transport

[RFC4838]describes various issues that make the IP protocol suite not suitable for space. One of them was TCP handshake and timers that would not work over a 20 minute delay link. If TLS handshake is added on top of TCP, then it is even worse. In fact, this is similar, but not identical because of disruptions, to large bandwidth-delay product use cases[RFC1072], because the delay is large, even if the bandwidth is somewhat much smaller than on Internet. In last 25 years, transport protocols have evolved a lot. QUIC ([RFC9000], [RFC9001], ...) is a potential transport solution to the space communications characteristics, given all its novel features.

Current implementations of QUIC typically set the initial RTT estimates in hundreds of milliseconds as it is expected to be a good start for connecting establishment on Internet. However, that value is way too low for long delay communications in deep space. By adjusting initial RTT to a proper value in the QUIC stack for the deep space connection, and given that many timers in QUIC are related to the RTT, a successful connection can be established. An initial proof of concept[picoquic-poc] with a QUIC implementation[picoquic] showed potential use of QUIC in deep space. Additional timers and parameters may need to be adjusted. Further investigation of QUIC use in deep space is needed, especially for congestion control, detection and recovering from loss of data. A document describing how to profile a QUIC stack for this use case is being written[draft-huitema-quic-in-space]. The possible use of careful resume [I-D.ietf-tsvwg-careful-resume] should be considered, as a way to dynamically update QUIC client stacks based on a better known RTT estimate from the server.

Establishing a QUIC connections includes discovering the network conditions, which typically requires several RTT. For space communications, we need to minimize the impact of the initial delays either by keeping connections up for a long time, or if that is not possible by using mechanisms like 0RTT or careful resume to accelerate the re-discovery of network conditions by the new connection.

The ability to have multiple streams and applications within a single QUIC connection is also very valuable and useful for this use case. A ground station may setup the initial QUIC connection with a spacecraft and then carry all needed applications and streams over that same connection.

Given that spacecrafts and ground systems are aware of each other and are typically managed by the same organization, the trust anchors may be preloaded so that each peer already have a trust relationship with the other. This, together with the resume token acquired during a previous connection, would enable the use of the 0-RTT QUIC feature enabling sending application data on the first packet. Even more, the initial QUIC connection could be established while the spacecraft is on the ground, so that while moving in space, the connection only needs to be updated with new RTT estimates, either by configuration or automatically. It should be noted that the 0-RTT feature is encrypted but vulnerable to replay attacks, which should be considered in the missions risk assessment.

The mandatory ability to have TLS negotiated at the beginning of the QUIC connection makes the use of IP layer security (aka IPSec) less needed to be deployed, while knowing that IP and transport security are different. Operational complexity in space is very costly and brings brittleness of the reliability and therefore, a single layer of security, managed at the application transport (aka HTTP-QUIC), is very valuable.

Session key and certificate lifetime together with certificate validation and trust chain anchors need to be carefully configured and handled. This is further discussed in section Section 7.4.

The store-and-forward paradigm, needed for deep space, may also be implemented at the QUIC transport layer instead of the IP layer, by architecting a string of QUIC proxies. Those proxies would need to react to the socket error of no connection by storing the data payload, until the next opportunity arise. However, that would require that the actual topology of the network be known at each proxy and that contact plans also be known at the QUIC proxy layer so that the proxy does not try to contact when in fact, no connectivity is happening. There would be no routing updates, so any link change from the contact plan would not be used. However, some messaging layer may be put on a network of message relays, connected by QUIC connections.

QUIC in deep space has very good promise but requires extensive investigation and testing to ensure proper usage and deployment.

5. HTTP

HTTP by itself has no notion of time. An HTTP request and response may take minutes or hours to be completed, theoretically. However, current infrastructure and software on Internet have various time-related configurations that will not work as is in the deep space context.

HTTP headers containing time, such as Cache-Control and Expires [RFC9111] need to be set large enough to cover the longest delay so that expiration does not happen before the actual data arrives at the destination. As with any HTTP application and content on Internet, these headers should be set properly based on the deployment use case, which is ever more important for deep space. Similarly, when continuous content transfer is used, as with 100-Continue [RFC9110], proper values for headers should be set.

HTTP clients and servers typically have default timeouts that shall be modified. For example, curl [curl] has the "-m" option for this use case. Similarly, HTTP server implementations have various timeouts configuration variables to be set properly. Testing with HTTP client Curl and HTTP server nginx and an introduced network delay of 20 minutes showed that HTTP communications work just fine with very basic configuration changes.

HTTP applications themselves must be developed using an asynchronous pattern and if they have timeouts, they should be adjusted / appropriately.

Internet Web sites are designed with the assumption of hundred of milliseconds delay and relatively always connected, where pages contain multiple queries to further get resources, media, queries to web services and downloading additional code and frameworks. This could work in theory in this context of space, but it will not be optimal, as multiple queries will be generated and therefore taking multiple RTT before the whole page is received complete. This issue can be mitigated by using various techniques such as Web Assembly [wasm] or pre-caching. Moreover, tt could be possible to have very basic HTML pages with zero or very few href and no media content unless locally cached to be used. An example would be a rover on Mars presenting an HTTP server with a base and bare HTML page to offer basic info on its status (maybe all in text) and some additional detailed pages, most likely also in base html text. However, it is foreseen that most applications based on QUIC-HTTP transport in deep space would be using REST or similar asynchronous patterns and not typical web browsing.

Caching should be used extensively on celestial bodies networks to maximize local fetching. Preemptive caching by pre-populating caches with data that shall be used locally on the celestial body network shall be done as much as possible to provide better response time on the local celestial body network.

QPACK [RFC9204] should be considered for higher bandwidth efficiency.

It should be noted that COAP [RFC7252] is worth considering for application transport in deep space.

6. Applications

There are a large number of IETF-defined IP-based application protocols, as well as non-standard ones. Some may work as is in a deep space environment, some others may require changes in timers or else in protocol or in the implementation, and some may not work at all. It would be appropriate to pick the most likely used application protocols and assess their usability for this use case. It may also be useful to test implementations. Obviously, the needed characteristic of these application protocols is the asynchronous paradigm, given long delays and intermittent communications. One outcome of this assessment could be a best practice document on how to write applications in such use case.

It should be noted that if the application is using HTTP as a transport, and that guidance on using HTTP as described in Section 5 is followed, then the HTTP application should work.

7. Network services

7.1. Domain Name System(DNS)

Domain name requests and response over long delays generate timeouts and when there is no reachability to the DNS server, requests will not be answered. Therefore, on celestial bodies IP networks, a local DNS infrastructure with all the names and values stored locally is needed. Moreover, to keep the same DNS root and the current DNSSEC trust chain, all keys necessary for validation should also be stored locally. The DNSSEC RR TTL values would need to be longer than the mission lifetime. [dns-isolated-networks] describes the various ways to achieve naming in isolated networks use case, which applies to deep space.

7.2. Network Management

NETCONF[RFC6241] and RESTCONF[RFC8040] can be used with proper configuration values in implementations to avoid timeouts. RESTCONF with appropriate HTTP config would enable long delayed queries to be working. NETCONF uses TCP which won't work on delayed and intermittent communications. If NETCONF is defined over QUIC, then it could be used with proper QUIC profile as discussed in Section 4. On the other hand, RESTCONF with proper HTTP profile would just work fine.

RESTCONF uses various timestamps and HTTP time related headers to compare transactions time, so for example to avoid any race in configuration changes. However, there is no notion of timeout in the protocol itself, so it should work as is. Implementations, at the RESTCONF level or underneath (HTTP) may have implementation-specific timeouts that should be configured properly to handle long delays.

A network manager should obviously be aware that a RESTCONF notification sent by a server that travels over some delayed links or networks will arrive later than typical, and such notification may be less useful at the time it arrives. However, this is the reality of a delayed and intermittent communications network.

SSH in its normal interactive mode sends each character in a separate packet, which over long delay networks, will not be optimal. SSH can be configured in line mode where packets are sent only when a full line is entered. That mode shall be preferred. However, SSH like NETCONF runs over TCP which will not work. However, SSH over QUIC was proposed[I-D.bider-ssh-quic] which proper QUIC configuration might work.

As a summary, it seems possible that all IP nodes (and even dual-stack (BP and IP) nodes) in space can be remotely managed using RESTCONF, as well as locally managed.

7.3. Network Operations and Security

On Earth, as it is planned today, the space network shall be isolated from the current Internet by "air gap", to disable any direct communications from Internet to deep space. Moreover, destination IP prefixes filtering shall be used to restrict the traffic to only the relevant one for each link. Note that this shall also be implemented in the routing control plane, but additional security might be appropriate to further protect the deep space links.

Each celestial network edge device shall have firewall rules to disable non-useful trafic to go through deep space links. If communications from Mars may only occur to Earth, but not Moon, then appropriate filtering based on destination IPv6 prefixes shall be used.

Given the air gap on Earth for Internet, there shall be no default route advertised in space that could for example point to Earth Internet.

Caching should be used aggressively in all levels of the IP stack in this architecture. For example, DNS servers on remote celestial bodies should have all useful names already configured. HTTP Caches should be deployed and preemptively filled with all necessary objects.

IPsec may be used to provide secure tunnelling or VPN set of services. However, QUIC-TLS may be instead investigated to provide the needed security, given that at the transport level, the security parameters may be dictated by the applications, therefore ensuring security policies from end to end applications, instead of at the network level.

There will likely be multiple "operators" on celestial IP networks. Therefore it is likely needed to provide some kind of exchange point similar to the Internet Exchange Point(IXP) as we know today on Internet. This would enable local exchange of trafic on the celestial body without going through deep space links. BGP should then be considered.

7.4. Key Management and Distribution

Protocols or infrastructure using crypto keys and certificates should be carefully managed. Certificates and session keys should have a lifetime large enough so that they will still be valid even when longer than expected disruptions happen. For missions, certificate lifetimes should be considered based on a mission extended lifetime. The whole trust chain for certificate validation, including updates, should also be transferred in advance to the appropriate location on the celestial body network to avoid invalid verifications because of the lack of an intermediate certificate, as one do not want to query any parent certificate over space links. Similarly, Certificate Revocation Lists(CRL) real-time fetching [RFC5280] or Online Certificate Status Protocol[RFC2560] should be investigated before considering their use in this environment.

7.5. Time

Since this framework reuses the IP protocol stack, it inherently assumes time coherence between the celestial bodies networks and no changes in how protocols are specifying time. Therefore, it practically assumes UTC-based time being available on all celestial bodies, so that time related comparisons can be achieved. The way to accomplish this on the celestial bodies networks and in space is out of scope for this document.

8. Summary

With proper profiling of protocols, software and operations, and with possibly little to no changes in protocols, the use of the IP protocol stack in deep space with long delays and intermittent communications seems possible and provides an alternative to a Bundle protocol based network. This is now possible to envision because of the various advances in the IP stack, specially the QUIC transport, compared to the initial assessment done 25 years ago.

9. IANA Considerations

This memo includes no request to IANA.

10. Security Considerations

TBD

11. References

11.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

11.2. Informative References

[RFC1072] Jacobson, V. and R. Braden, "TCP extensions for long-delay paths", RFC 1072, DOI 10.17487/RFC1072, October 1988, <https://www.rfc-editor.org/info/rfc1072>.

[RFC2560] Myers, M., Ankney, R., Malpani, A., Galperin, S., and C. Adams, "X.509 Internet Public Key Infrastructure Online Certificate Status Protocol - OCSP", RFC 2560, DOI 10.17487/RFC2560, June 1999, <https://www.rfc-editor.org/info/rfc2560>.

[RFC3633] Troan, O. and R. Droms, "IPv6 Prefix Options for Dynamic Host Configuration Protocol (DHCP) version 6", RFC 3633, DOI 10.17487/RFC3633, December 2003, <https://www.rfc-editor.org/info/rfc3633>.

[RFC4193] Hinden, R. and B. Haberman, "Unique Local IPv6 Unicast Addresses", RFC 4193, DOI 10.17487/RFC4193, October 2005, <https://www.rfc-editor.org/info/rfc4193>.

[RFC4271] Rekhter, Y., Ed., Li, T., Ed., and S. Hares, Ed., "A Border Gateway Protocol 4 (BGP-4)", RFC 4271, DOI 10.17487/RFC4271, January 2006, <https://www.rfc-editor.org/info/rfc4271>.

[RFC4291] Hinden, R. and S. Deering, "IP Version 6 Addressing Architecture", RFC 4291, DOI 10.17487/RFC4291, February 2006, <https://www.rfc-editor.org/info/rfc4291>.

[RFC4838] Cerf, V., Burleigh, S., Hooke, A., Torgerson, L., Durst, R., Scott, K., Fall, K., and H. Weiss, "Delay-Tolerant Networking Architecture", RFC 4838, DOI 10.17487/RFC4838, April 2007, <https://www.rfc-editor.org/info/rfc4838>.

[RFC4861] Narten, T., Nordmark, E., Simpson, W., and H. Soliman, "Neighbor Discovery for IP version 6 (IPv6)", RFC 4861,

DOI 10.17487/RFC4861, September 2007, <https://www.rfc-editor.org/info/rfc4861>.

[RFC5280] Cooper, D., Santesson, S., Farrell, S., Boeyen, S., Housley, R., and W. Polk, "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile", RFC 5280, DOI 10.17487/RFC5280, May 2008, <https://www.rfc-editor.org/info/rfc5280>.

[RFC5340] Coltun, R., Ferguson, D., Moy, J., and A. Lindem, "OSPF for IPv6", RFC 5340, DOI 10.17487/RFC5340, July 2008, <https://www.rfc-editor.org/info/rfc5340>.

[RFC5880] Katz, D. and D. Ward, "Bidirectional Forwarding Detection (BFD)", RFC 5880, DOI 10.17487/RFC5880, June 2010, <https://www.rfc-editor.org/info/rfc5880>.

[RFC6241] Enns, R., Ed., Bjorklund, M., Ed., Schoenwaelder, J., Ed., and A. Bierman, Ed., "Network Configuration Protocol (NETCONF)", RFC 6241, DOI 10.17487/RFC6241, June 2011, <https://www.rfc-editor.org/info/rfc6241>.

[RFC6724] Thaler, D., Ed., Draves, R., Matsumoto, A., and T. Chown, "Default Address Selection for Internet Protocol Version 6 (IPv6)", RFC 6724, DOI 10.17487/RFC6724, September 2012, <https://www.rfc-editor.org/info/rfc6724>.

[RFC7122] Kruse, H., Jero, S., and S. Ostermann, "Datagram Convergence Layers for the Delay- and Disruption-Tolerant Networking (DTN) Bundle Protocol and Licklider Transmission Protocol (LTP)", RFC 7122, DOI 10.17487/RFC7122, March 2014, <https://www.rfc-editor.org/info/rfc7122>.

[RFC7252] Shelby, Z., Hartke, K., and C. Bormann, "The Constrained Application Protocol (CoAP)", RFC 7252, DOI 10.17487/RFC7252, June 2014, <https://www.rfc-editor.org/info/rfc7252>.

[RFC7567] Baker, F., Ed. and G. Fairhurst, Ed., "IETF Recommendations Regarding Active Queue Management", BCP 197, RFC 7567, DOI 10.17487/RFC7567, July 2015, <https://www.rfc-editor.org/info/rfc7567>.

[RFC8040] Bierman, A., Bjorklund, M., and K. Watsen, "RESTCONF Protocol", RFC 8040, DOI 10.17487/RFC8040, January 2017, <https://www.rfc-editor.org/info/rfc8040>.

[RFC9000] Iyengar, J., Ed. and M. Thomson, Ed., "QUIC: A UDP-Based Multiplexed and Secure Transport", RFC 9000, DOI 10.17487/

RFC9000, May 2021, <https://www.rfc-editor.org/info/rfc9000>.

[RFC9001] Thomson, M., Ed. and S. Turner, Ed., "Using TLS to Secure QUIC", RFC 9001, DOI 10.17487/RFC9001, May 2021, <https://www.rfc-editor.org/info/rfc9001>.

[RFC9110] Fielding, R., Ed., Nottingham, M., Ed., and J. Reschke, Ed., "HTTP Semantics", STD 97, RFC 9110, DOI 10.17487/RFC9110, June 2022, <https://www.rfc-editor.org/info/rfc9110>.

[RFC9111] Fielding, R., Ed., Nottingham, M., Ed., and J. Reschke, Ed., "HTTP Caching", STD 98, RFC 9111, DOI 10.17487/RFC9111, June 2022, <https://www.rfc-editor.org/info/rfc9111>.

[RFC9171] Burleigh, S., Fall, K., and E. Birrane, III, "Bundle Protocol Version 7", RFC 9171, DOI 10.17487/RFC9171, January 2022, <https://www.rfc-editor.org/info/rfc9171>.

[RFC9172] Birrane, III, E. and K. McKeever, "Bundle Protocol Security (BPSec)", RFC 9172, DOI 10.17487/RFC9172, January 2022, <https://www.rfc-editor.org/info/rfc9172>.

[RFC9174] Sipos, B., Demmer, M., Ott, J., and S. Perreault, "Delay-Tolerant Networking TCP Convergence-Layer Protocol Version 4", RFC 9174, DOI 10.17487/RFC9174, January 2022, <https://www.rfc-editor.org/info/rfc9174>.

[RFC9204] Krasic, C., Bishop, M., and A. Frindell, Ed., "QPACK: Field Compression for HTTP/3", RFC 9204, DOI 10.17487/RFC9204, June 2022, <https://www.rfc-editor.org/info/rfc9204>.

[I-D.qu-tvr-schedule-yang] Qu, Y., Lindem, A., and M. Blanchet, "YANG Model for Scheduled Attributes", Work in Progress, Internet-Draft, draft-qu-tvr-schedule-yang-00, 7 July 2023, <https://datatracker.ietf.org/doc/html/draft-qu-tvr-schedule-yang-00>.

[I-D.ietf-tsvwg-careful-resume] Kuhn, N., Emile, S., Fairhurst, G., and C. Huitema, "Careful Convergence of Congestion Control from Retained State", Work in Progress, Internet-Draft, draft-ietf-tsvwg-careful-resume-01, 5 July 2023, <https://datatracker.ietf.org/doc/html/draft-ietf-tsvwg-careful-resume-01>.

[I-D.blanchet-tvr-contactplan] Blanchet, M., Torgerson, J. L., and Y. Qu, "Contact Plan Yang Model for Time-Variant Routing of

the Bundle Protocol", Work in Progress, Internet-Draft, draft-blanchet-tvr-contactplan-01, 7 July 2023, <https://datatracker.ietf.org/doc/html/draft-blanchet-tvr-contactplan-01>.

[I-D.blanchet-tvr-forwarding]
Blanchet, M., "Forwarding in the context of Time-Variant Routing(TVR)", Work in Progress, Internet-Draft, draft-blanchet-tvr-forwarding-00, 13 March 2023, <https://datatracker.ietf.org/doc/html/draft-blanchet-tvr-forwarding-00>.

[I-D.bider-ssh-quic] bider, D., "QUIC-based UDP Transport for Secure Shell (SSH)", Work in Progress, Internet-Draft, draft-bider-ssh-quic-09, 2 December 2020, <https://datatracker.ietf.org/doc/html/draft-bider-ssh-quic-09>.

[dns-isolated-networks]
Blanchet, M., "Domain Name System in Mostly Isolated Networks", Work in Progress, Internet-Draft, draft-blanchet-dns-isolated-networks-00, September 2023, <https://datatracker.ietf.org/doc/html/draft-blanchet-dns-isolated-networks-00>.

[IPoverCCSDSSpaceLinks] Consultative Committee on Space Data Systems(CCSDS), "IP OVER CCSDS SPACE LINKS, Blue Book 702", September 2012, <https://public.ccsds.org/Pubs/702x1b1c1.pdf>.

[SANAIPEHeaderRegistry] Space Assigned Numbers Authority, "Internet Protocol Extension Header", <https://sanaregistry.org/r/ipe_header/>.

[wasm] World Wide Web Consortium(W3C), "WebAssembly Specifications", <https://github.com/webassembly/spec>.

[ioag] Lunar Communications Architecture Working Group, Interagency Operations Advisory Group, "The Future Lunar Communications Architecture, Report of the Interagency Operations Advisory Group", January 2022, <https://www.ioag.org/Public%20Documents/Lunar%20communications%20architecture%20study%20report%20FINAL%20v1.3.pdf>.

[ioag-mars] Mars and Beyond Communications Architecture Working Group, Interagency Operations Advisory Group, "The Future Mars Communications Architecture, Report of the Interagency Operations Advisory Group", February 2022, <https://www.ioag.org/Public%20Documents/MBC%20architecture%20report%20final%20version%20PDF.pdf>.

[picoquic-poc]
Huitema, C., "QUIC to Mars", February 2023, <https://www.privateoctopus.com/2023/02/07/quic-to-mars.html>.

[picoquic] Huitema, C., "picoquic", <https://github.com/private-octopus/picoquic>.

[draft-huitema-quic-in-space] Huitema, C. and M. Blanchet, "QUIC in Space", <https://github.com/huitema/quic-in-space>.

[TVRWG] IETF, "Time-Variant Routing (tvr)", <https://datatracker.ietf.org/group/tvr/about/>.

[curl] "Curl", <https://curl.se>.

[CCSDSWEB] CCSDS, "Consultative Committee for Space Data Systems", <https://ccsds.org>.

[buffer-bloat] Gettys, J., "The Blind Men and the Elephant", <https://gettys.wordpress.com/2018/02/11/the-blind-men-and-the-elephant/>.

Appendix A. Additional Considerations

This section lists additional considerations that are important for the deployment of IP in deep space, but may not require changes to protocols or implementations, as they are more related to network operations.

A.1. IP Version

As discussed in the previous section, space missions are long-lived and require full reachability to every spacecraft. Since IPv4 address space is consumed, the use of IPv4 address space would rely on using Network Address Translation(NAT) in space. The significant consequence is the one-way reachability that NAT creates: as soon as there is a NAT in the path from the source to the destination, the destination is not be directly reachable. For example, if a NAT is deployed in space, the spacecrafts behind the NAT will not be directly reachable from Earth missions operations and network management consoles. If the NAT is on the other side of the connections, then spacecrafts will not be able to communicate or send notifications to mission operations or network management consoles. This is a significant issue if using IPV4 in deep space.

The Internet has been transitioning from IPv4 to IPv6 to continue its expansion and since the missions are long-lived, IPv6 is the only IP version that has the appropriate lifetime for the deep space network.

IPv6 also brings specific features such as IPv6 DHCP prefix delegation[RFC3633] that could be used for spacecrafts as mobile networks docking into a temporary or more permanent network and getting a prefix from the attaching network.

Finally, running both IPv4 and IPv6 simultaneously, while doable as we do on Internet today, brings additional operational challenges both in security, such as handling multiple versions of access control lists, and network management that one should avoid as much as possible in space.

Therefore, IPv6 is recommended to be the only IP protocol version used in space. Moreover, if there is any required protocol change, IPv6 should be the base IP protocol used underneath.

Most protocols and topics discussed in this document are independent of the IP version, but if there are differences, only the IPv6 version is discussed.

A.2. IPv6 Addressing

Space communications infrastructure must avoid at all costs any address space collision, since it would prevent any reachability between the colliding networks. For example, if one organization creates a network on a celestial body using an address space and another organization, or even worse the same organization, creates another network anywhere in space using the same address space, those two networks will not be able to reach each other, undermining any communications. NATs can be used between the two but this will just further complexify or disable remote network management.

IPv6 addressing in space should only use non-overlapping address space, based on duly allocated IPv6 space assigned to each organization, from the public IPv6 address space[RFC4291] managed by registries, providing IPv6 network services in space. IPv6 unique-local addressing [RFC4193] can be used within a single domain but such traffic should not cross domains using the unique-local addressing, and should instead used the global addressing as managed by the IPv6 preferred address algorithm [RFC6724].

Current Regional Internet Registries(RIR) may not have in place the appropriate policies for the deep space use case, since these policies are aimed towards terrestrial Internet usage, such as broadband usage, ISP peering and other considerations. These policies may need to be revised to include deep space usage or another organization with proper membership to be put in place to allocate and assign IPv6 address space, and possibly Autonomous System Numbers, for that specific community and usage.

A.3. IPv6 over Space Links

The Consultative Committee for Space Data Systems (CCSDS, [CCSDSWEB] is a standard organization for space communications, which membership is space agencies and related commercial organizations. IP packet encapsulation into space links is defined in CCSDS 702 [IPoverCCSDSSpaceLinks] and the codepoint for IPv6 packet encapsulation is 87 for space links, as specified in the Space Assigned Numbers Registry(SANA) [SANAIPEHeaderRegistry]. However, it is unknown if IPv6 has been implemented and tested, given long delays and Neighbor Discovery [RFC4861] timers. An IPv6 over CCSDS Space Links specification may need to be defined.

A.4. Bundle Protocol and IP

As discussed in Section 1, the Bundle Protocol [RFC9171] has been designed to create a store-and-forward networking capability for space and other use cases. This document framework specifies an alternate way to accomplish the networking for the same use case by reusing as much as possible the IP protocol stack, therefore inheriting all the engineering and implementations implemented and running daily on Internet. This re-use also means that implementations have been exercised on real traffic orders of magnitude more than what has been achieved with Bundle protocol stacks.

It is possible to mix BP and IP on the same links or networks. Therefore, this framework and the BP stack can create independent and simultaneous networks or can be mixed.

Acknowledgements

This work started by reassessing the use of the whole IP stack in the context of deep space. Soon, QUIC was identified as the key technology for this endeavour. Christian Huitema was very helpful in not only confirming the ability to use QUIC but also took the time and effort to test and modify its picoquic stack[picoquic] to confirm the initial hypothesis[picoquic-poc]. Its involvement and confirmation are the key for the launch of this work. Then, Martin Thompson has been also kind to take time to answer initial questions on QUIC, further confirming the possibility of using QUIC for deep space. Since then, many individuals have provided significant comments and perspectives on this subject.

This document and its underlying work has been reviewed and discussed by many, who have provided valuable feedback and comments, including disagreements, and made an overall more solid document. These people are, in no specific order: Geoff Huston, Martin Thompson, Peter Ashwood-Smith, Tony Li, George Neville-Neil, Jim Gettys, Nicolas

Kuhn, Eric Vyncke, Russ Housley, Emile Stephan, Dave Taht, Jean-Philippe Dionne.

Authors' Addresses

Marc Blanchet
Viagenie
Canada

Email: marc.blanchet@viagenie.ca

Christian Huitema
Private Octopus Inc.

Email: huitema@huitema.net

Dean Bogdanovic
AlefEdge, Inc

Email: ivandean@gmail.com