

Internet Engineering Task Force
Internet-Draft
Intended status: Informational
Expires: 11 October 2026

M. Blanchet
Viagenie
W. Eddy
Aalyria Technologies
T. Li
Hewlett Packard Enterprise
9 April 2026

An Architecture for IP in Deep Space
draft-ietf-tiptop-ip-architecture-00

Abstract

The IP protocol stacks used on Earth's Internet are typically configured based on assumptions of short delays and mostly uninterrupted communications. This document describes an architecture of the IP protocol stack tailored for its use in deep space. It involves buffering IP packets in IP forwarders facing intermittent links and adjusting transport protocol configurations and application protocol timers. This architecture applies to the Moon, Mars, and general interplanetary networking.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <https://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on 11 October 2026.

Copyright Notice

Copyright (c) 2026 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document.

Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Revised BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Revised BSD License.

Table of Contents

1. Introduction	2
1.1. Document and Discussion Location	4
2. Architecture Overview	4
3. Layer 2 in Deep Space	7
3.1. Celestial Body Surface	7
3.2. Deep Space Links	7
3.3. Celestial Body Orbits	7
4. Internet Protocol	7
4.1. IP Forwarding and Store-and-Forward	8
4.2. Header Compression	9
5. IP Addressing and Routing	9
5.1. Addressing	9
5.2. Routing	10
6. Transport	11
6.1. General Transport Issues	11
6.2. UDP	14
6.3. QUIC	15
6.4. Other Transports	16
7. Applications	16
7.1. HTTP	16
7.1.1. CoAP	18
8. Network services	18
8.1. Naming	18
8.2. Network Management	18
9. IANA Considerations	19
10. Security Considerations	19
11. References	20
11.1. Informative References	20
Acknowledgements	27
Authors' Addresses	28

1. Introduction

Deep-space communications involve long delays and intermittent connectivity due to the distances and orbital motion involved. For instance, Earth to Mars transmissions vary 4-24 minutes, and are unavailable periodically due to rotation, conjunctions, and other conditions.

Deep space communications have traditionally been done on a layer-2 point-to-point basis, sometimes using relays, but no layer-3 networking has been in routine use. [RFC4838] reports an assessment done around 25 years ago concluding that the IP protocol stack was not suitable for deep space networking. However, some things have changed since that time, as there has been evolution in multiple areas including Internet transport and security protocols (e.g. QUIC), routing (e.g. SDN), and forwarding technology (e.g. MPLS and Segment Routing), compression (e.g. SCHC), network management, and applications (e.g. HTTP and CoAP).

Currently, space agencies have published plans that include deploying IP networks on celestial bodies, such as the Moon [ioag] and Mars [ioag-mars], both on the surfaces and orbiting constellations, including link layer technologies such as Wi-Fi or 5G. On the surface, the plans involve dense networking around facilities and habitats. New mission concepts are also including clusters of multiple networked nodes co-located at Lagrange points.

Given the evolution of modern IP application protocol stacks and the new needs of deep space missions, this document describes an architecture for the use of IP in deep space, meeting needs described in [I-D.ietf-tiptop-usecase]. This includes:

- * running IP over end-to-end paths that include high-latency, deep space layer-2 links,
- * buffering of IP packets, when needed,
- * IP addressing and routing,
- * transport protocol configuration,
- * application protocols,
- * and other network services, including network management of deep space IP networks.

An aspect of this framework is the potential for queueing outbound packets for upcoming contact opportunities. In this case, the IP layer has to deal with the fact that a next-hop may not be currently reachable and that IP packets could be buffered for an unusual amount of time, such as minutes, hours, or days in the forwarding device waiting for next-hop reachability or a link-up opportunity. The transport layer should be able to work with long and variable delays, including intermittent communications. The application protocols and the applications themselves should be properly set to wait a longer time than on the current Internet to receive a response to a query.

Finally, all network services such as routing, security, naming, and network management should also be adapted to this new context. This document is structured around these layers. General architecture concerns are described along with suitable ways to address them. Separate documents cover the details of configuring specific protocols, e.g. CoAP, DNS, QUIC, in accordance with this framework.

Applying the IP stack and its mature family of standards in deep space enables the reuse of many protocols, tools, and software currently used on the Internet. However, many components of the IP stack cannot be used as-is and require careful configuration and deployment considerations that are discussed in this document.

Since the Moon is within a few light seconds away from Earth, it is possible to configure and run typical Internet protocols and applications to basically "work", however, Mars with a much longer delay is more difficult. The framework in this document covers both, and would also work for longer delays, such as reaching Jupiter or the whole Solar System. The focus is on the Moon and Mars because those are where significant numbers of relevant missions are being developed presently.

This document uses "deep space" extensively as opposed to "space", as that more generally includes other topics such as terrestrial satellite access, earth-orbiting, and near-Earth missions, which are not covered in this document, and subject of other RFCs (e.g. [RFC2488], [RFC2760], [RFC3135], [RFC9717]). The scope of "deep space" in this document includes the Moon, although this is not strictly within the ITU's definition of the term.

The key characteristics of space communications and networking, the relevant use case aspects and requirements are discussed in [I-D.ietf-tiptop-usecase].

1.1. Document and Discussion Location

The source of this document is located at <https://github.com/marcblanchet/draft-tiptop-ip-architecture>. Comments or changes are welcomed using a PR or an issue. However, this document should be discussed on the deepspace@ietf.org mailing list of the Taking IP To Other Planets (TIPTOP) working group.

2. Architecture Overview

This section provides a short overview of this architecture for IP in deep space.

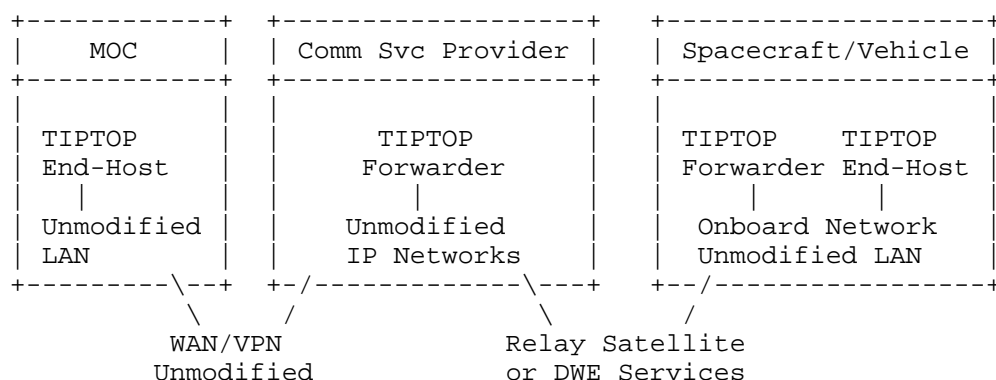
One of the key reasons to use IP is to maintain compatibility and potential smooth interoperability with other IP networks that exist terrestrially and within space vehicle onboard networks. While some nodes will require specific deep space protocol support, many nodes within the network can have traditional unmodified IP stacks that remain configured as if for typical Internet uses.

Within an end-to-end deep space application data flow, several different types of nodes can be involved:

- * Unmodified / Internet nodes - These consists of existing "untuned" IP stacks (not necessarily modified in any way for deep space use versus typical Internet use). These can exist within planetary surface networks and spacecraft onboard networks, for example. For instance, these may be the majority of nodes within terrestrial ground systems, mission operations centers, ground stations, and terrestrial backbone networks. Given use of 3GPP architecture in some agency plans, these nodes may be present within onboard or surface core networks.
- * TIPTOP end-hosts - These nodes include specially tuned transport/application protocol parameters in order to function in deep space scenarios. Within the network, these specially tuned nodes need only exist at application endpoints (or proxies).
- * TIPTOP forwarding nodes - These are like typical IP forwarding nodes, except include queuing capabilities for IP packets that go beyond normal Internet use, as described more in Section 4. Within the total deep space network, these need only exist at transition points between well-connected regimes and the hops through deep space communication system links. For instance, these may be at the edge of onboard networks or ground station networks.
- * TIPTOP-full nodes - These nodes combine aspects of TIPTOP end hosts and TIPTOP forwarding nodes. This combination may be useful in cases such as small spacecraft or other vehicles that consist of only a single node that must roam autonomously rather than functioning within a well-connected onboard or surface network regime.

- * Scheduling/orchestration systems - These systems may typically operate out-of-band from application data flows, but are responsible for creating and distributing coordinated network plans based on projected application needs (e.g. to support planned mission operations schedules). These may only exist within terrestrial or other planetary core networks, but are essential, especially in early growth of deep space networks, in order to provide knowledge of expected connectivity schedules and associated stack parameters.

The figure below illustrates a simplified example set of end-to-end of mission network elements based on this taxonomy, with the data flow described following.



An example end-to-end command application flow might pass through many nodes, starting within a mission operations center (MOC) at (1) a TIPTOP end-host running command workstation software, and then traversing (2) an unmodified LAN within the MOC, which routes the command packet via (3) an unmodified WAN or VPN connection to a communication service provider (CSP) network. Within the CSP network, there may be (4) unmodified IP forwarding in other WAN, VPN, and LAN nodes, and (5) a TIPTOP forwarder that queues packets and routes according to schedule via (6) unmodified relay or direct communications services towards the destination spacecraft node. At the destination spacecraft/vehicle, there may be (7) a TIPTOP forwarder, (8) an unmodified onboard network, and (9) a TIPTOP end-host.

Within this example command path, there are 2 TIPTOP end hosts, 2 TIPTOP forwarders, and 5+ unmodified typical Internet stack nodes (since multiple hops can be involved in the several LAN and WAN traversals). This deep space IP architecture is incrementally deployable.

Making the end-to-end example command data flow possible, an orchestration system on Earth can use knowledge of the spacecraft mission and its commanding needs in order to ensure that at proper times there are scheduled communication system resources (antennas, modems, etc.) that are physically pointed, tracking, and otherwise configured to close a forward link to the spacecraft at the required data rate. The orchestration system can also set scheduled routing table entries for the relevant nodes within the path.

3. Layer 2 in Deep Space

3.1. Celestial Body Surface

The Interagency Operations Advisory Group (IOAG) [ioag] has defined the communications architecture for the Moon and Mars. On the celestial body surface, it is planned to use 3GPP and Wi-Fi link layer protocols. IP will be used over these link layers.

3.2. Deep Space Links

Deep space links typically use the Consultative Committee for Space Data Standards (CCSDS) [CCSDSWEB] standards for link layers, such as Telecommand (TC) [CCSDS_TC], Telemetry (TM) [CCSDS_TM], Advanced Orbiting Systems (AOS) [CCSDS_AOS], Proximity1 (Prox1) [CCSDS_PROX1] or the Unified Space Data Link Protocol (USLP) [CCSDS_USLP]. CCSDS has defined a generic encapsulation mechanism for the payloads for all these link layer protocols with IP as an encapsulated protocol [IPoverCCSDSSpaceLinks] [SANAIPEHeaderRegistry]. Therefore, IP packets can be transported over any CCSDS link layers.

3.3. Celestial Body Orbits

For celestial body orbits, IOAG has planned the use of CCSDS link layer protocols. However, as on Earth, it may be possible to use 6G-NTN technology[ntn6g3gpp] around celestial bodies, such as in lunar or Martian orbits. 6G-NTN technologies use IP as its layer 3 technology.

4. Internet Protocol

IPv4 or IPv6 packets can be carried as is over long delays and disruptions, as IP has no notion of time. Originally, the Time To Live (TTL) field of IPv4 was defined based on time [STD5], but it has been effectively implemented as a hop count, which was renamed as "Hop Count" in IPv6 [STD86]. Nothing needs to be changed to the IP protocol or its packet format.

This architecture can support IPv4 and/or IPv6. While some may prefer one version or another, there are important considerations that complicate matters. Deep space links are very low bandwidth. The additional overhead of an IPv6 packet will have a bandwidth and performance impact that needs to be considered early in mission planning. From a future-proofing perspective, IPv6 is to be preferred. Eventually, as technology evolves and the bandwidth constraints are relieved, it would be preferable to not have to retrofit existing, deployed spacecraft with IPv6. Ultimately, however, the IETF does not have authority over this decision. The various space agencies, working in unison, should make this joint decision and select one version for interoperability.

4.1. IP Forwarding and Store-and-Forward

For deep space applications, an IP packet may need to be queued for much longer periods than typical for the Internet when the next hop is currently unreachable or undefined, which can happen due to planning of periodic contacts around orbital dynamics, or other antenna scheduling limitations. Terms have been used including "store-and-forward" (which is already used differently elsewhere in the context of switch architectures), or "store, carry, and forward", to describe the behavior of buffering IP packets rather than immediately discarding them when the next-hop is not immediately available on-link.

This queuing may be implemented at layer 2 as currently done by the Mars orbiters, where frames are stored, regardless of payload type. In this case, IP packets are unaware of the store-and-forward and no changes are needed in the IP forwarding function. The layer 2 network is just behaving as a point-to-point link with a large and variable latency.

If an IP forwarder has an interface on an intermittent link, and that link is down, some destinations may become unreachable when there is no alternate route. In this case, the forwarder buffers the IP packets locally instead of dropping them. This might be implemented as a deep queue with active queue management (AQM) [RFC7567]. When the route to the destination is back, on the same link or a different link, maybe minutes or hours later, the stored packets can be transmitted.

This requires proper provisioning of buffer storage memory for the target deployment and usage. If the buffer is full, then packets must be dropped. The choice of which packets to drop depends on the policies configured on the node, which may be a to drop depends on the policies configured on the node, which may be a function of traffic class, source or destination IP addresses, flow labels, or other parameters. An example is described in [I-D.blanchet-tvr-forwarding].

Packets might also be lost if a buffer is cleared for any other reason (e.g. due to reboot), or corrupted somehow (e.g. due to cosmic rays or other uncorrectable memory upsets). Given the use of end to end reliable transport, this architecture does not require IP forwarding nodes to necessarily implement anything beyond a typical "best effort" service and reliability, though more complex means to support reliable packet delivery are compatible and can interoperate within this architecture if desirable.

4.2. Header Compression

Deep space links are point-to-point links and bandwidth in space is very valuable, so header compression is very effective. Static Context Header Compression (SCHC) [I-D.ietf-schc-architecture] is a header compression technique that relies on rules in a static context and is, therefore, more efficient for deep space. SCHC should be considered on a deep space point-to-point link or a string of L2 links.

5. IP Addressing and Routing

5.1. Addressing

The IP address space is a hierarchical namespace where ranges of addresses are encoded as "prefixes". Individual domains advertise prefixes to the broader Internet and assign these addresses internally. Prefixes may be aggregated into less-specific prefixes, which makes the routing subsystem more efficient by decreasing overhead.

Space networks provide a unique opportunity to provide extremely efficient routing by assigning a unique prefix or block of addresses per celestial body and its proximal orbits. Management of the IP address space is currently documented in [RFC7020], but this only covers continental regions and does not provide for addressing for space. The depletion of the IPv4 address space means that there is little than can be done if agencies decide to use IPv4.

However, if agencies decide to utilize IPv6, it would be deeply beneficial if addressing was designed to allow for the maximum aggregation of routing prefixes. Aggregation of prefixes assigned to celestial bodies would minimize the overhead incurred by relaying spacecraft, minimizing expensive hardware requirements and would help minimize route flap. Thus, to help achieve maximum aggregation, the address space for outer space should be managed by a Regional Internet Registry (RIR) and blocks of address space should be allocated for each celestial body of interest. Space service providers should use prefixes assigned by this RIR. This is discussed in more detail in [I-D.li-tiptop-address-space].

5.2. Routing

Existing routing protocols require proof of liveness between protocol partners, implemented through the periodic exchange of packets between partners. This is impractical on long-delay or intermittent links, so a Path Computation Engine (PCE) [RFC4655] based approach seems appropriate for those domains possibly supplemented by contact plan schedules[I-D.ietf-tvr-schedule-yang]. Interconnection between domains can still be done with BGP [RFC4271], but long-delay or intermittent links should be avoided. Domains straddling such links must provide proxy advertisements for prefixes reachable across such links.

Optimal routing for domains with intermittent links is out of scope for this document.

On the surface of celestial bodies and in proximal orbit, traditional protocols are applicable and should be used (e.g., [RFC9717]).

Because of bandwidth limitations, the PCE for a domain may require the ability to specify an explicit path, but using an explicit path across an intermittent link may be problematic. If a packet is bound to an explicit path and stored in an intermediate node, then the path may be invalid at the end of the storage period. Conveying a backup path within the packet itself would incur a large performance penalty and is not recommended. Allowing the intermediate node itself to compute the remainder of the path would seem to be the most robust solution. Conventional traffic engineering techniques for getting to storage nodes seem to be appropriate.

6. Transport

Modern transport protocols and stacks share similar algorithms for common transport needs. This section first addresses general transport issues, and then addresses use of specific individual transport protocols. While there have been academic research experiments in developing new protocols specifically for interplanetary transport, they are not directly considered here because the goal is to leverage IETF protocols and commonly available software.

6.1. General Transport Issues

Internet transport protocols and transport stacks have developed to meet the needs for different classes of applications on the terrestrial Internet, using similar algorithms with parameters that are tuned for typical conditions. Some of these algorithms might simply be parameterized differently for interplanetary network paths, while other algorithms may have fundamental challenges. In some cases, minor adjustments (or no adjustment) will suffice for Earth-lunar networking.

- * Protocol Negotiation / "Happy Eyeballs" - Due to the presence of both IPv4 and IPv6 in the terrestrial Internet, applications or transport stacks may include "happy eyeballs" capabilities [RFC8305] that make use of multiple DNS queries and connection attempts for different addresses and address families returned. Downsides to this in interplanetary applications might include (1) the additional load on interplanetary DNS (although solutions to this are available [I-D.many-tiptop-dns]), (2) unnecessary use of network capacity for parallel connection attempts, and (3) unnecessary additional transport state maintained for an extended period of time. If applications for interplanetary use initially rely on either fixed IP addresses (instead of DNS names) or a single address family (e.g. IPv6), then there should be no downsides to presence of happy eyeballs algorithms within the transport stack. If happy eyeballs is present and triggered, then the values in section 8 of [RFC8305] should be carefully considered and tuned based on the DNS and path expectations (e.g. resolution delay, first address family count, connection attempt delay, minimum connection attempt delay, and maximum connection attempt delay) as most of these values are not likely to be appropriate even for Earth-lunar paths.
- * Connection Initiation / Handshaking - Very long-lived connection state may be used to avoid connection initiation in some cases, e.g. by establishing state prior to launch and using those pre-established connections long-term. However, this will not be

possible in all cases for all applications, if flows can't be planned pre-launch. Due to the need to be robust to stale packets, errors, and denial of service attacks, historically, Internet transports and security protocols have included handshaking state machines, such as 3-way and 4-way handshakes for connection establishment (needing 1.5 or 2 RTTs, if client authentication is needed and pre-shared keys are not available). Since the interplanetary round trip times may be larger than the duration of contact periods, these handshaking mechanisms are very inefficient. Though they are tolerable in cases such as between Earth and lunar networks, they are stifling for Earth-Mars and other interplanetary network paths. Transports, such as QUIC, that have the ability to resume based on shared state from prior application connections or to rapidly start transferring data ("0-RTT") can be suitably efficient, once initial state is obtained; however, they still require a complete initial handshake with the full set of round trip times imposed. The use of 0-RTT is subject to replay attacks[RFC9001] and therefore should be considered to be disabled depending on the security policy of the mission. For interplanetary use, it may be beneficial to find ways to securely pre-set information to allow this more efficient startup, without requiring the full initial handshake even once.

- * Capability / Feature Negotiation - Ideally, transport protocol feature advertisements and agreements can be completed prior to launch as part of connection pre-establishment and cached long-term. Any negotiation of additional features that requires full round trip times is prohibitive for general interplanetary use, especially if data transmission is stalled while negotiation takes place. Commonly, much negotiation can occur in the initial connection handshake. It will be beneficial if transport stacks can cache (or securely pre-configure caches) of pre-negotiated features with typical hosts that they plan to communicate with during the course of a mission.
- * Retransmission - Reliable transport protocols typically keep retransmission timers, computed based on round trip time samples [RFC6298] [RFC8961]. Since it may be impossible to even take RTT measurements within a contact window, and RTTs may vary significantly across contact windows, this type of algorithm is not appropriate for interplanetary use. Additionally, default values may be in ranges such as 1 second, that are inappropriate for even Earth-lunar paths. Based on the known propagation times for interplanetary paths and predicted contacts, either pre-configured values or new algorithms should be used.

- * Handling Failures - Aside from retransmission, there are also other types of timers and counters in transport stacks, for things including DNS resolution, connection establishment retries, connection keep-alives, user timeouts, and delayed acknowledgement. Additionally, transports receive and respond to ICMP messages that indicate other different types of errors from within the network. Similar to the case of retransmission timers, other types of timers that support failure handling should be adjusted based on estimated propagation and forwarding times. Other heuristics for validating ICMP messages and responding may need to be considered.
- * Congestion Control - Internet congestion control algorithms are based on timely receipt of feedback in order to detect losses, delays, or other signals, and typically attempt to react rapidly. Two challenges in the interplanetary case include: (1) The relative delays on interplanetary paths are much longer (preventing "rapid" response) and may vary in orders of magnitude between segments, e.g. across a terrestrial segment, an Earth-Mars segment, and a Mars orbit/surface segment. Congestion may occur in low-latency portions of the network, but fail to be detected quickly because loss/acknowledgement information propagates and feeds back too slowly, leading to sustained loss due to unresponsiveness in a low-latency portion of the network. (2) By the time feedback is received, it is badly out of date, if not completely irrelevant to the future. The advent of IP-based in-network storage for scheduled interplanetary links also changes the way that congestion can be measured (e.g. in delay-based protocols). In the near-term, and in present space mission operations, it can be assumed that data link capacity is explicitly planned and scheduled end-to-end in order to accommodate mission needs. This makes congestion control algorithms largely replaceable with simple nominal rate and flow control. However, for larger and more dynamic future interplanetary networks, and shared trunk links, etc., it will be useful in the future to develop more optimized congestion control methods, including possibly more effective network-based signaling/feedback, rather than simply end-to-end feedback-based mechanisms.
- * Path MTU Discovery - Internet transports typically include probing algorithms and rely on end-to-end acknowledgements (or in legacy cases ICMP errors) in order to infer the maximum packet sizes that can be used at any time without introducing unnecessary IP fragmentation. Because for interplanetary cases, feedback may be slow or unrelated to the current or future paths by the time it arrives, it may be more beneficial to simply rely on known / pre-arranged path MTU limitations that can be communicated between interplanetary providers and other connected providers and users.

Especially for higher bandwidth links, it may be beneficial for this to be significantly larger than the minimum Internet MTU values that are normally assumed. For instance, terrestrially many systems use 9000+ byte MTUs. MTUs in interplanetary networks may vary due to the range of link bandwidths, and simultaneous desires to be efficient on fast links while also wanting to avoid head-of-line blocking for large packets on low-bandwidth links.

- * Multiple Streams - Several transports allow for applications to send/receive multiple independent streams of application data, rather than requiring separate connections (and handshakes, and state) for each stream. Due to the need to leverage pre-established state and make efficient use of connectivity opportunities, these capabilities can be valuable to leverage for interplanetary use, however, specific mechanisms may need to be evaluated/tuned/modified e.g. to avoid additional round trip times.
- * Multipath Transport - There may be a variety of different physical paths between planets (e.g. using relays or direct links), that could be attractive for a mission to use simultaneously as way to increase reliability for mission data, or to increase throughput for an application. However, typical Internet multipath transport algorithms are oriented more towards failover than towards replicating or load-balancing traffic. They may only switch flows between using alternative paths (e.g. for failover), or allow different streams to use different paths, rather than replicating data. For deep space use, Internet multipath transport algorithms could either be tuned or replaced.
- * Forward Error Correction (FEC) - Generally, FEC is expected to be implemented below the link layer. Within the transport layer, FEC and/or erasure coding are not common in typical Internet use today, though FEC and/or erasure coding can be integrated with different protocol stacks. Because it can allow for data recovery in the case of packet losses, without suffering extra round trips, or requiring bidirectional connectivity, this may be valuable to incorporate/enable in future interplanetary transport stacks.

6.2. UDP

UDP [RFC768] has no notion of time and, therefore can be used as-is in deep space. Hence, protocols using UDP transport can be used in space as-is, if they do not rely on time or can be configured with timeouts appropriate in deep space.

The IETF has published UDP usage guidelines [RFC8085] that help to ensure applications using UDP behave in a "safe" way for the Internet. These guidelines are also appropriate to consider for deep space usage of UDP, though some specific values (e.g. the 1 second initial latency estimate suggested) should be adjusted.

6.3. QUIC

QUIC [RFC9000] like most IP transport protocols implements congestion control mechanisms, which, based on various metrics such as calculated delays or packet loss, pace the rate of sending packets at the source node to decrease the perceived congestion in the network. QUIC supports many new features suitable and useful in deep space such as 1 RTT for connection establishment and security, mobility, 0 RTT for early data after mobility, streams and user space [RFC9000].

Current implementations of QUIC typically set various transport configuration parameters suitable for the Internet environment, expecting an RTT to be in the hundreds of milliseconds and a normally always-connected network. Therefore, QUIC stacks using default configurations will not work in deep space. However, studies and simulations [quic-sim] showed that with proper configuration of parameters, QUIC stacks can support the delays and disruptions in deep space. [I-D.many-tiptop-quic-profile] describes how to properly configure a QUIC stack for deep space applications, where QUIC is unaware of disruptions. If the transport is aware of the disruptions, further optimizations are possible.

Having multiple streams and applications within a single QUIC connection is valuable and useful for deep space. A ground station may set up the initial QUIC connection with a spacecraft and then carry all needed applications and streams over that same connection for the whole duration of the mission.

Session keys and certificate lifetimes together with certificate validation and trust chain anchors need to be carefully configured and handled.

QUIC proxies [I-D.ietf-masque-quic-proxy] can be used at the edge of space to isolate, apply policies, or optimize traffic at the ingress/egress to a celestial body network.

6.4. Other Transports

The Space Communications Protocol Specification (SCPS) Transport Protocol (SCPS-TP) [SCPS-TP] is a set of TCP options that is specifically designed to extend TCP for space use. It has been tested in simulations of Earth-Mars links. SCPS-TP may be a useful transport protocol for some applications to consider in the future. Primarily, to-date, the implementations and usage have been in transport gateway / proxy scenarios that do not match the intended IP architecture of this document, and code is not widely available compared to other transport options.

Other transports such as TCP [RFC9293], SCTP [RFC9260], DCCP [RFC4340] and others were not investigated for their suitability in space.

7. Applications

While many Web applications already use REST, operating them in deep space requires a concrete deployment and design model that differs from terrestrial assumptions. A functional deep-space Web system would consist of RESTful HTTP (or CoAP) services with explicitly configured timeouts, caching behavior, and asynchronous request handling. Application endpoints are designed to accept long request-response delays, return idempotent resources, and avoid chatty multi-RTT interactions. Content is aggressively pre-cached and replicated at planetary or orbital gateways so that most REST interactions are satisfied locally, while end-to-end exchanges tolerate hours or days of delay. In this model, a “Web app” is not an interactive browser session, but a set of asynchronous REST resources whose state can be fetched, updated, or synchronized opportunistically as connectivity becomes available. This section describes the specific protocol configurations and application patterns needed to adapt existing REST-based systems into such a deep-space-capable Web architecture.

7.1. HTTP

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

HTTP headers containing time, such as Cache-Control and Expires [RFC9111], should not be used or if used, must be set to large enough values 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 the Internet, these headers should be set

properly based on the deployment use case, which is even 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 should be modified. For example, curl [curl] has the "-m" option for this use case. Similarly, HTTP server implementations have various timeout configuration variables that must be set properly. Testing with HTTP client Curl and HTTP server nginx and an introduced network delay of minutes, hours and days showed[quic-sim] that HTTP communications work well with basic configuration changes.

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

Internet websites are designed with the assumption of hundreds of milliseconds delay and relatively always connected, where pages contain multiple queries to get further resources, media, queries to web services, and downloading additional code and frameworks. This could work in theory in space, but it will not be optimal, as multiple queries will be generated and therefore take multiple RTT before the whole page is received complete. This issue can be mitigated by using various techniques such as pre-caching. Moreover, it could be possible to have simple HTML pages with no or very few references and no media content that was not locally cached. 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 HTTP3 transport in deep space would be using REST or similar asynchronous patterns and not typical web browsing.

Caching should be used extensively on space 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.

7.1.1. CoAP

The Constrained Application Protocol (CoAP)[RFC7252] is a specialized web transfer protocol for use with constrained nodes and constrained networks, therefore a candidate for an application protocol for space, similar to HTTP. If considered, proper use of its underlying transport and timers should be looked at as discussed in [I-D.gomez-tiptop-coap].

8. Network services

8.1. Naming

For small-scale deployments, one can use IP addresses directly or a mapping from a name to an IP address such as /etc/hosts. However, this does not provide easy dynamic updates, scaling by hierarchy, service discovery, authentication of records, etc. Therefore, the Domain Name System (DNS) shall be considered early on in the space deployment. However, naming hierarchy and infrastructure must be carefully designed to avoid name resolution over deep space links, given that answers may come after minutes or hours. There are clear advantages of having the space name hierarchy anchored to the current Internet root, as it enables DNSSEC through the same security infrastructure currently used and deployed. Using the same root also does not require new policies. A new TLD or a new root is way more complicated and does not bring any significant value compared to using the current domain tree.

Care must be taken to manage key lifecycles and resource record lifetimes. [I-D.many-tiptop-dns] discusses the various methods and the naming hierarchy that should be used in space.

8.2. Network Management

NETCONF [RFC6241] and RESTCONF [RFC8040] shall be used with proper configuration values to avoid timeouts and appropriate transport. NETCONF over QUIC transport [I-D.ietf-netconf-over-quic] or RESTCONF over HTTP3 can be configured with appropriate QUIC transport parameters as discussed in Section 6.3.

Given the non-typical delays in requests and response in space compared to traditional network management frameworks on Internet and private networks, expectations about when configuration changes will be applied and confirmed or the timeliness of the actual value received from a request need to be taken into account. For example, a configuration change may take hours to be received by the spacecraft, which is not expected in typical network operations. A request for some value may take hours to be received by the

spacecraft and take hours to be received by the requestor, which means the value may not be current or expired. Moreover, typical timeouts of NETCONF clients should be adjusted to the expected RTT.

While being declared historic in IETF, SNMP[RFC1157] runs over UDP and has no notion of time. Therefore, with proper configuration of client timeout, it can be used as is to manage nodes and services in deep space.

9. IANA Considerations

This memo includes no request to IANA.

10. Security Considerations

Using the current IP protocol stack in deep space inherits all the work on privacy, cryptography, key management, firewalls, and scrutiny of protocols that are deployed on the Internet. Since no change is made in the protocols, this architecture does not bring new security issues on the protocols themselves.

However, with longer delays and intermittence, deep space networking brings additional considerations. Security designs must tolerate long validation delays and potentially operate in a disconnected trust model.

Certificates and keys need to be renewed before their expiration, taking into account the delay to send, receive and confirm. Protocols such as OCSP[RFC6960] providing on-line real-time validation and revocation check will likely not work given the too long delays, therefore certificates need to be validated using local trust anchors.

The use of long term keys, such as ones set prior to launch, may create exposure, therefore keys should be renewed at appropriate frequency.

Given possible lower frequency of time synchronization, clock drifts may affect expiration and validation.

Intermediate forwarding nodes may buffer packets for a significant time. While it is presumed that most IP packet payloads will be encrypted by IP or transport security, data-at-rest becomes a possible exposure. If the intermediate node is compromised, the data-at-rest becomes available for the bad actor. This is new and not expected on terrestrial Internet intermediate nodes.

If bad traffic is injected sufficiently to fill out the intermediate nodes buffer, then this becomes a denial-of-service attack.

If some packets are buffered in one intermediate node, If multiple paths are possible to a destination, then bad packet injection may use an alternate faster path that would result in the destination receiving the injected bad packets before the proper ones, which may become denial-of-service attacks of various kinds (destination endpoints buffer full, connection resets, ...). While this scenario can happen on terrestrial Internet, the bad traffic level and the duration window may be orders of magnitude larger than on Internet, therefore having a potentially much larger impact.

Given possible short contact windows and relatively few alternate paths, an attacker may flood a link during the whole contact window, disabling any remediation during that contact window, which means the actual impact will remain longer than the actual attack duration.

Given relatively few alternate paths, malicious injection of routes may have a larger impact than on terrestrial Internet.

Given a sparse network of few nodes and more predictable traffic patterns than terrestrial Internet, traffic analysis may become more effective, even for encrypted traffic.

Security considerations of each transport protocol are discussed in their respective transport protocol profile document.

Given low bandwidth, low number of alternate paths, high costs of links and nodes high value, access control to the deep space network and related policies should be in place. For example, at the beginning of the deployment, the deep space network shall be isolated from the current Internet by an "air gap", to disable any direct communications from the Internet to deep space. Moreover, destination IP prefix 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 tiptop forwarding node, such as celestial network edge device, shall have firewall rules to prevent inappropriate traffic from entering deep space links. If communications from Mars may only occur to Earth, but not to the Moon, then appropriate filtering based on destination IP prefixes shall be used.

11. References

11.1. Informative References

- [RFC768] Postel, J., "User Datagram Protocol", STD 6, RFC 768, DOI 10.17487/RFC0768, August 1980, <<https://www.rfc-editor.org/info/rfc768>>.
- [RFC1157] Case, J., Fedor, M., Schoffstall, M., and J. Davin, "Simple Network Management Protocol (SNMP)", RFC 1157, DOI 10.17487/RFC1157, May 1990, <<https://www.rfc-editor.org/info/rfc1157>>.
- [RFC2488] Allman, M., Glover, D., and L. Sanchez, "Enhancing TCP Over Satellite Channels using Standard Mechanisms", BCP 28, RFC 2488, DOI 10.17487/RFC2488, January 1999, <<https://www.rfc-editor.org/info/rfc2488>>.
- [RFC2760] Allman, M., Ed., Dawkins, S., Glover, D., Griner, J., Tran, D., Henderson, T., Heidemann, J., Touch, J., Kruse, H., Ostermann, S., Scott, K., and J. Semke, "Ongoing TCP Research Related to Satellites", RFC 2760, DOI 10.17487/RFC2760, February 2000, <<https://www.rfc-editor.org/info/rfc2760>>.
- [RFC3135] Border, J., Kojo, M., Griner, J., Montenegro, G., and Z. Shelby, "Performance Enhancing Proxies Intended to Mitigate Link-Related Degradations", RFC 3135, DOI 10.17487/RFC3135, June 2001, <<https://www.rfc-editor.org/info/rfc3135>>.
- [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>>.
- [RFC4340] Kohler, E., Handley, M., and S. Floyd, "Datagram Congestion Control Protocol (DCCP)", RFC 4340, DOI 10.17487/RFC4340, March 2006, <<https://www.rfc-editor.org/info/rfc4340>>.
- [RFC4655] Farrel, A., Vasseur, J.-P., and J. Ash, "A Path Computation Element (PCE)-Based Architecture", RFC 4655, DOI 10.17487/RFC4655, August 2006, <<https://www.rfc-editor.org/info/rfc4655>>.
- [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>>.

- [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>>.
- [RFC6298] Paxson, V., Allman, M., Chu, J., and M. Sargent, "Computing TCP's Retransmission Timer", RFC 6298, DOI 10.17487/RFC6298, June 2011, <<https://www.rfc-editor.org/info/rfc6298>>.
- [RFC6960] Santesson, S., Myers, M., Ankney, R., Malpani, A., Galperin, S., and C. Adams, "X.509 Internet Public Key Infrastructure Online Certificate Status Protocol - OCSP", RFC 6960, DOI 10.17487/RFC6960, June 2013, <<https://www.rfc-editor.org/info/rfc6960>>.
- [RFC7020] Housley, R., Curran, J., Huston, G., and D. Conrad, "The Internet Numbers Registry System", RFC 7020, DOI 10.17487/RFC7020, August 2013, <<https://www.rfc-editor.org/info/rfc7020>>.
- [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>>.
- [RFC8085] Eggert, L., Fairhurst, G., and G. Shepherd, "UDP Usage Guidelines", BCP 145, RFC 8085, DOI 10.17487/RFC8085, March 2017, <<https://www.rfc-editor.org/info/rfc8085>>.
- [RFC8305] Schinazi, D. and T. Pauly, "Happy Eyeballs Version 2: Better Connectivity Using Concurrency", RFC 8305, DOI 10.17487/RFC8305, December 2017, <<https://www.rfc-editor.org/info/rfc8305>>.
- [RFC8961] Allman, M., "Requirements for Time-Based Loss Detection", BCP 233, RFC 8961, DOI 10.17487/RFC8961, November 2020, <<https://www.rfc-editor.org/info/rfc8961>>.

- [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>>.
- [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>>.
- [RFC9260] Stewart, R., Txen, M., and K. Nielsen, "Stream Control Transmission Protocol", RFC 9260, DOI 10.17487/RFC9260, June 2022, <<https://www.rfc-editor.org/info/rfc9260>>.
- [RFC9293] Eddy, W., Ed., "Transmission Control Protocol (TCP)", STD 7, RFC 9293, DOI 10.17487/RFC9293, August 2022, <<https://www.rfc-editor.org/info/rfc9293>>.
- [RFC9717] Li, T., "A Routing Architecture for Satellite Networks", RFC 9717, DOI 10.17487/RFC9717, January 2025, <<https://www.rfc-editor.org/info/rfc9717>>.
- [STD5] Internet Standard 5, <<https://www.rfc-editor.org/info/std5>>.
At the time of writing, this STD comprises the following:
- Postel, J., "Internet Protocol", STD 5, RFC 791, DOI 10.17487/RFC0791, September 1981, <<https://www.rfc-editor.org/info/rfc791>>.
- Postel, J., "Internet Control Message Protocol", STD 5, RFC 792, DOI 10.17487/RFC0792, September 1981, <<https://www.rfc-editor.org/info/rfc792>>.

Mogul, J., "Broadcasting Internet Datagrams", STD 5, RFC 919, DOI 10.17487/RFC0919, October 1984, <<https://www.rfc-editor.org/info/rfc919>>.

Mogul, J., "Broadcasting Internet datagrams in the presence of subnets", STD 5, RFC 922, DOI 10.17487/RFC0922, October 1984, <<https://www.rfc-editor.org/info/rfc922>>.

Mogul, J. and J. Postel, "Internet Standard Subnetting Procedure", STD 5, RFC 950, DOI 10.17487/RFC0950, August 1985, <<https://www.rfc-editor.org/info/rfc950>>.

Deering, S., "Host extensions for IP multicasting", STD 5, RFC 1112, DOI 10.17487/RFC1112, August 1989, <<https://www.rfc-editor.org/info/rfc1112>>.

[STD86] Internet Standard 86, <<https://www.rfc-editor.org/info/std86>>. At the time of writing, this STD comprises the following:

Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", STD 86, RFC 8200, DOI 10.17487/RFC8200, July 2017, <<https://www.rfc-editor.org/info/rfc8200>>.

[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.ietf-masque-quic-proxy]

Pauly, T., Rosenberg, E., and D. Schinazi, "QUIC-Aware Proxying Using HTTP", Work in Progress, Internet-Draft, draft-ietf-masque-quic-proxy-08, 2 March 2026, <<https://datatracker.ietf.org/doc/html/draft-ietf-masque-quic-proxy-08>>.

[I-D.ietf-netconf-over-quic]

Dai, J., Yu, S., Cheng, W., Blanchet, M., and P. Andersson, "NETCONF over QUIC", Work in Progress, Internet-Draft, draft-ietf-netconf-over-quic-07, 18 January 2026, <<https://datatracker.ietf.org/doc/html/draft-ietf-netconf-over-quic-07>>.

[I-D.ietf-schc-architecture]

Pelov, A., Thubert, P., and A. Minaburo, "Static Context Header Compression (SCHC) Architecture", Work in Progress, Internet-Draft, draft-ietf-schc-architecture-05, 17 October 2025, <<https://datatracker.ietf.org/doc/html/draft-ietf-schc-architecture-05>>.

[I-D.ietf-tvr-schedule-yang]

Qu, Y., Lindem, A., Kinzie, E., Fedyk, D., and M. Blanchet, "YANG Data Model for Scheduled Attributes", Work in Progress, Internet-Draft, draft-ietf-tvr-schedule-yang-08, 9 February 2026, <<https://datatracker.ietf.org/doc/html/draft-ietf-tvr-schedule-yang-08>>.

[I-D.li-tiptop-address-space]

Li, T. and M. Eubanks, "IP Address Space for Outer Space", Work in Progress, Internet-Draft, draft-li-tiptop-address-space-01, 4 February 2026, <<https://datatracker.ietf.org/doc/html/draft-li-tiptop-address-space-01>>.

[I-D.many-deepspace-ip-assessment]

Blanchet, M., Huitema, C., and D. Bogdanovi, "Revisiting the Use of the IP Protocol Stack in Deep Space: Assessment and Possible Solutions", Work in Progress, Internet-Draft, draft-many-deepspace-ip-assessment-02, 10 September 2024, <<https://datatracker.ietf.org/doc/html/draft-many-deepspace-ip-assessment-02>>.

[I-D.ietf-tiptop-usecase]

Blanchet, M., Eddy, W., and M. Eubanks, "IP in Deep Space: Key Characteristics, Use Cases and Requirements", Work in Progress, Internet-Draft, draft-ietf-tiptop-usecase-01, 21 January 2026, <<https://datatracker.ietf.org/doc/html/draft-ietf-tiptop-usecase-01>>.

[I-D.many-tiptop-quic-profile]

Blanchet, M. and W. Eddy, "QUIC Profile for Deep Space", Work in Progress, Internet-Draft, draft-many-tiptop-quic-profile-02, 1 March 2026, <<https://datatracker.ietf.org/doc/html/draft-many-tiptop-quic-profile-02>>.

[I-D.gomez-tiptop-coap]

Gomez, C. and S. Aguilar, "CoAP in Space", Work in Progress, Internet-Draft, draft-gomez-tiptop-coap-00, 30 September 2025, <<https://datatracker.ietf.org/doc/html/draft-gomez-tiptop-coap-00>>.

[I-D.many-tiptop-dns]

Blanchet, M., "Deployment and Use of the Domain Name System(DNS) in Deep Space", Work in Progress, Internet-Draft, draft-many-tiptop-dns-02, 2 April 2026, <<https://datatracker.ietf.org/doc/html/draft-many-tiptop-dns-02>>.

[IPoverCCSDSSpaceLinks]

Consultative Committee on Space Data Systems (CCSDS), "IP OVER CCSDS SPACE LINKS, Blue Book 702", September 2012, <<https://public.ccsds.org/Pubs/702x1b1c2.pdf>>.

[SCPS-TP] Consultative Committee on Space Data Systems (CCSDS), "Space Communications Protocol Specification (SCPS) - Transport Protocol (SCPS-TP), Blue Book 714.0-B-2", October 2006.

[SANAIPEHeaderRegistry]

"Internet Protocol Extension Header", <https://sanaregistry.org/r/ipe_header/>.

[CCSDS_AOS]

Consultative Committee on Space Data Systems (CCSDS), "AOS Space Data Link Protocol, Blue Book 732.0-B-4", October 2021, <<https://public.ccsds.org/Pubs/732x0b4.pdf>>.

[CCSDS_TM] Consultative Committee on Space Data Systems (CCSDS), "TM Space Data Link Protocol, Blue Book 132.0-B-3", October 2021, <<https://public.ccsds.org/Pubs/132x0b3.pdf>>.

[CCSDS_TC] Consultative Committee on Space Data Systems (CCSDS), "TC Space Data Link Protocol, Blue Book 232.0-B-4", October 2021, <<https://public.ccsds.org/Pubs/232x0b4e1c1.pdf>>.

[CCSDS_PROX1]

Consultative Committee on Space Data Systems (CCSDS), "Proximity-1 Space Link Protocol—Data Link Layer, Blue Book 211.0-B-6", July 2020, <<https://public.ccsds.org/Pubs/211x0b6e1.pdf>>.

- [CCSDS_USLP] Consultative Committee on Space Data Systems (CCSDS), "Unified Space Data Link Protocol, Blue Book 732.1-B-2", October 2021, <<https://public.ccsds.org/Pubs/732x1b2s.pdf>>.
- [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>>.
- [curl] "Curl", <<https://curl.se>>.
- [CCSDSWEB] CCSDS, "Consultative Committee for Space Data Systems", <<https://ccsds.org>>.
- [ntn6g3gpp] 3GPP, "Non-Terrestrial Networks (NTN)", <<https://www.3gpp.org/technologies/ntn-overview>>.
- [quic-sim] Blanchet, M., "Deep Space IP: Some simulation results", July 2024, <<https://deepspaceip.github.io/meetings/ietf120/ietf120-deepspaceip-simulation-results.pdf>>.

Acknowledgements

This work started by reassessing the use of the whole IP stack in the context of deep space discussed in [I-D.many-deepspace-ip-assessment] where early contributors are acknowledged.

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: Padma Pillay-Esnault, Marius Feldmann, Britta Hale, Carles Gomez Montenegro, Meiling Chen, Marshall Eubanks, Eric Rescorla, Johnson Liu.

Authors' Addresses

Marc Blanchet
Viagenie
Canada
Email: marc.blanchet@viagenie.ca

Wesley Eddy
Aalyria Technologies
United States of America
Email: wes@aalyria.com

Tony Li
Hewlett Packard Enterprise
United States of America
Email: tony.li@tony.li