Network Working Group   C. Author
Request for Comments: 9999   Independent Submission
Category: Informational   March 2026
ISSN: 2070-1721

A Lightweight Protocol for Human-Readable Documentation on the World Wide Web

Status of This Memo

This document is not an Internet Standards Track specification; it is published for informational purposes.

This document is a product of an independent submission. It has been approved for publication by the RFC Editor, but is not a candidate for any level of Internet Standard; see Section 2 of RFC 7841.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at https://www.rfc-editor.org/info/rfc9999.

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.

Table of Contents

1. Introduction 4
2. Conventions and Definitions 5
3. Protocol Specification 6
3.1. Block Types 6
3.2. Requirement Keywords 6
3.3. Message Format 7
3.4. Transport Handshake 7
3.5. System Architecture 8
3.6. Session State Machine 8
3.7. ABNF Grammar 8
4. Security Considerations 9
5. IANA Considerations 10
6. References 10
6.1. Normative References 10
6.2. Informative References 10
Appendix A. Example Document 11
Acknowledgements 11
Author's Address 11

1. Introduction

This document specifies a lightweight protocol for creating, exchanging, and rendering human-readable documentation in the style of IETF Request for Comments (RFC) documents. The protocol is intended to provide a structured, machine-parseable format that preserves the traditional typographic conventions established by the RFC Series.

The key motivations for this work include the growing need for structured documentation that can be both rendered for human consumption and processed by automated tools. Traditional plain-text RFC formatting, while venerable, lacks the semantic markup necessary for modern toolchains.

NOTE: This is an informational note. The protocol described herein is purely illustrative and is not intended for production deployment without further review.

The remainder of this document is organized as follows. Section 2 establishes terminology and conventions. Section 3 describes the core protocol mechanisms. Section 4 discusses security considerations, and Section 5 covers IANA registration requirements.

2. Conventions and Definitions

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.

Document Block
: A discrete semantic unit within an RFC document, such as a heading, paragraph, table, or diagram. Each block has a type and structured JSON content.

Renderer
: A software component that transforms a sequence of document blocks into a visual presentation suitable for human reading.

Sort Order
: A non-negative integer that determines the position of a block within its parent article. Blocks are rendered in ascending sort order.

Content Envelope
: The JSON object associated with each block that carries the block-type-specific payload.

3. Protocol Specification

3.1. Block Types

The protocol defines a set of block types that cover the full range of structural elements found in traditional RFC documents. Implementations MUST support all block types listed in this section. Each block type has a well-defined content schema.

- Structural blocks: header, title, heading, page_break, toc, boilerplate
- Text blocks: paragraph, note, raw, code_block
- List blocks: bullet_list, definition_list
- Visual blocks: ascii_table, packet_diagram, figure, state_machine, timeline
- Metadata blocks: reference, author_address, abnf

3.2. Requirement Keywords

Keyword	Meaning
MUST	Absolute requirement of the specification
MUST NOT	Absolute prohibition of the specification
SHOULD	Recommended but not strictly required
SHOULD NOT	Not recommended but not strictly prohibited
MAY	Truly optional; interoperability must be handled

Table 1: RFC 2119 Requirement Keywords

3.3. Message Format

Document exchange messages use a binary header inspired by the IPv4 packet format. The header carries version information, flags, and the total length of the payload in octets. All multi-octet fields are in network byte order (big-endian).

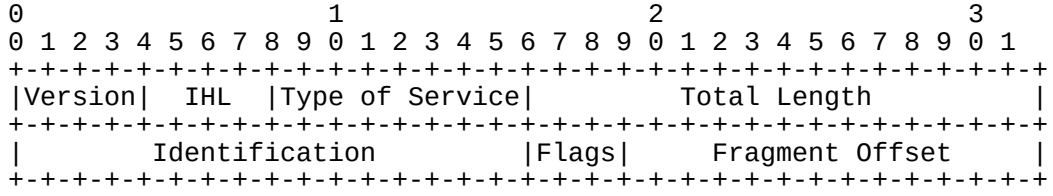


Figure 1: Document Exchange Header

3.4. Transport Handshake

Before exchanging documents, peers perform a three-way handshake to establish session parameters and negotiate supported block types. The handshake follows the familiar TCP SYN/SYN-ACK/ACK pattern.

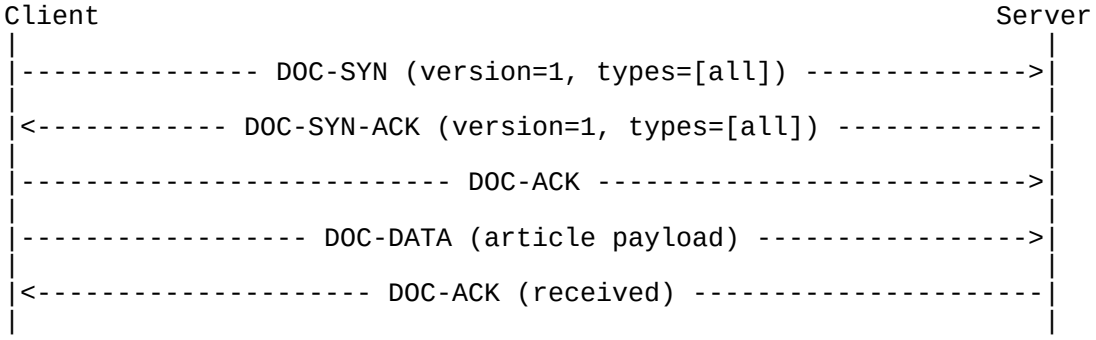


Figure 2: Document Exchange Handshake

3.5. System Architecture

The protocol stack consists of four layers. The application layer provides the document editing interface. The presentation layer handles block serialization. The session layer manages peer handshakes, and the transport layer provides reliable delivery.

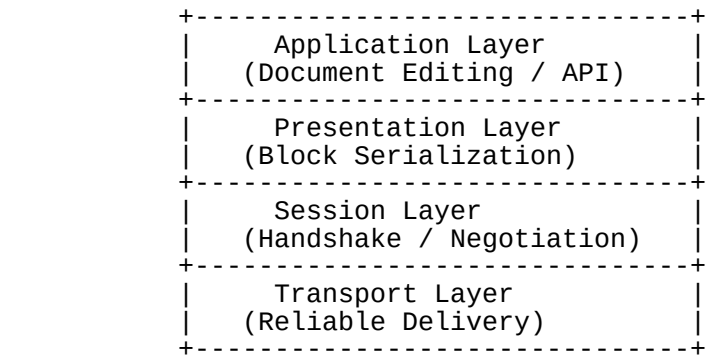
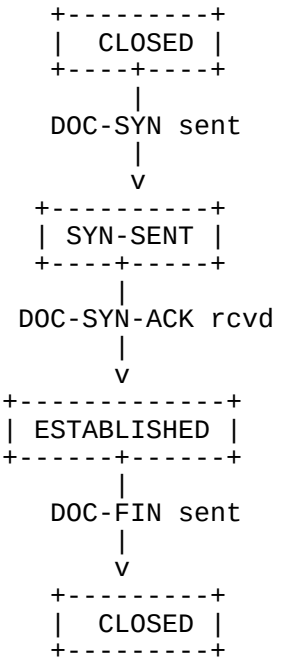


Figure 3: Protocol Layer Model

3.6. Session State Machine



3.7. ABNF Grammar

```
document      = header title *block
block         = heading / paragraph / list / table / diagram / reference
heading       = level number text anchor-id
level         = 1*2DIGIT
number        = 1*DIGIT *("." 1*DIGIT)
HTTP-Version  = "HTTP" "/" 1*DIGIT "." 1*DIGIT
```

4. Security Considerations

Implementations that render document blocks in a web browser MUST take care to sanitize all content to prevent cross-site scripting (XSS) attacks. Block content MUST NOT be inserted into the DOM as raw HTML. The use of Content Security Policy headers is RECOMMENDED.

The following Content-Security-Policy header is RECOMMENDED for deployments serving rendered documents:

```
Content-Security-Policy:
    default-src 'none';
    script-src 'self';
    style-src 'self' 'unsafe-inline';
    img-src 'self' data:;
    font-src 'self';
    connect-src 'self';
    frame-ancestors 'none';
```

Document payloads SHOULD be transmitted over TLS 1.3 or later. Implementations MUST reject connections using TLS versions prior to 1.2. The session handshake does not provide its own cryptographic protection and relies entirely on the underlying transport security.

5. IANA Considerations

This document has no IANA actions. The block types defined in Section 3.1 are maintained by the protocol implementors and do not require IANA registration at this time. Future extensions that introduce new block types SHOULD consider registering them with IANA under a dedicated registry.

6. References

6.1. 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>.

6.2. Informative References

[RFC7991] Hoffman, P., "The "xml2rfc" Version 3 Vocabulary", RFC 7991, DOI 10.17487/RFC7991, December 2016 <https://www.rfc-editor.org/info/rfc7991>.

Appendix A. Example Document

This appendix provides a minimal example of a conformant document consisting of a header block, a title block, a single heading, and a paragraph. Implementations MAY use this as a test vector for their parsers.

```
{
  "blocks": [
    {"type": "header", "content": {"left_lines": ["RFC 0000"], "right_lines":
["Author"]}},
    {"type": "title", "content": {"lines": ["Minimal Example"]}},
    {"type": "paragraph", "content": {"text": "Hello, world."}}
  ]
}
```

Acknowledgements

The author wishes to thank the IETF community for decades of work on the RFC format, which inspired the design of this protocol. Special thanks to the xml2rfc maintainers for their tireless efforts in tooling support.

Author's Address

C. Author
Independent Submission
1 Example Street
Singapore 000000
SG

Email: author@example.com
URI: https://example.com/rfc9999