

Instructions to RFC Authors

Status of this Memo

This memo provides information for the Internet community. This memo does not specify an Internet standard of any kind. Distribution of this memo is unlimited.

Table of Contents

1.	Introduction	1
2.	Editorial Policy	3
3.	Format Rules	4
3a.	ASCII Format Rules	4
3b.	PostScript Format Rules	5
4.	Header	6
4a.	First Page Heading	6
4b.	Running Header	7
4c.	Running Footer	7
5.	Status Section	5
6.	Introduction Section	8
7.	References Section	9
8.	Security Considerations Section	10
9.	Author's Address Section	10
10.	Relation to other RFCs	10
11.	Protocol Standards Process	11
12.	Contact	11
13.	Distribution Lists	11
14.	RFC Index	12
15.	Security Considerations	12
16.	References	12
17.	Author's Address	12
18.	Appendix - RFC "nroff macros"	13

1. Introduction

This Request for Comments (RFC) provides information about the preparation of RFCs, and certain policies relating to the publication of RFCs.

The RFC series of notes covers a broad range of interests. The core topics are the Internet and the TCP/IP protocol suite. However, any

topic related to computer communication may be acceptable at the discretion of the RFC Editor.

Memos proposed to be RFCs may be submitted by anyone. One large source of memos that become RFCs is the Internet Engineering Task Force (IETF). The IETF working groups (WGs) evolve their working memos (known as Internet Drafts or I-Ds) until they feel they are ready for publication, then the memos are reviewed by the Internet Engineering Steering Group (IESG), and if approved sent by the IESG to the RFC Editor.

RFCs are distributed online by being stored as public access files, and a short message is sent to the distribution list indicating the availability of the memo.

The online files are copied by the interested people and printed or displayed at their site on their equipment. This means that the format of the online files must meet the constraints of a wide variety of printing and display equipment. (RFCs may also be returned via e-mail in response to an e-mail query, or RFCs may be found using information and database searching tools such as Gopher, Wais, WWW, or Mosaic.)

RFCs have been traditionally published and continue to be published in ASCII text.

While the primary RFCs is always an ASCII text file, secondary or alternative versions of RFC may be provided in PostScript. This decision is motivated by the desire to include diagrams, drawings, and such in RFCs. PostScript documents (on paper only, so far) are visually more appealing and have better readability.

PostScript was chosen for the fancy form of RFC publication over other possible systems (e.g., impress, interpress, oda) because of the perceived wide spread availability of PostScript capable printers.

However, many RFC users read the documents online and use various text oriented tools (e.g., emacs, grep) to search them. Often, brief excerpts from RFCs are included in e-mail. These practices are not yet practical with PostScript files.

PostScript producing systems are less standard than had been assumed and that several of the document production systems that claim to produce PostScript actually produce nonstandard results.

In the future, it may be necessary to identify a set of document production systems authorized for use in production of PostScript

RFCs, based on the reasonableness of the output files they generate.

2. Editorial Policy

Documents proposed to be RFCs are reviewed by the RFC Editor and possibly by other reviewers he selects.

The result of the review may be to suggest to the author some improvements to the document before publication.

Occasionally, it may become apparent that the topic of a proposed RFC is also the subject of an IETF Working Group, and that the author could coordinate with the working group to the advantage of both. The usual result of this is that a revised memo is produced as a working group Internet Draft and eventually emerges from the IETF process as a recommendation from the IESG to the RFC Editor.

In some cases it may be determined that the submitted document is not appropriate material to be published as an RFC.

In some cases it may be necessary to include in the document a statement based on the reviews about the ideas in the document. This may be done in the case that the document suggests relevant but inappropriate or unsafe ideas, and other situations.

The RFC Editor may make minor changes to the document, especially in the areas of style and format, but on some occasions also to the text. Sometimes the RFC Editor will undertake to make more significant changes, especially when the format rules (see below) are not followed. However, more often the memo will be returned to the author for the additional work.

Documents intended to become RFCs specifying standards track protocols must be approved by the IESG before being sent to the RFC Editor. The established procedure is that when the IESG completes work on a document that is to become a standards track RFC the communication will be from the Secretary of the IESG to the RFC Editor. Generally, the documents in question are Internet Drafts. The communication usually cites the exact Internet Draft in question (by file name). The RFC Editor must assume that only that file is to be processed to become the RFC. If the authors have small corrections to the text, they should be sent to the RFC Editor separately (or as a "diff"), do not send a new version of the document.

In some cases, authors prepare alternate secondary versions of RFCs in fancy format using PostScript. Since the ASCII text version of the RFC is the primary version, the PostScript version must match the

text version. The RFC Editor must decide if the PostScript version is "the same as" the ASCII version before the PostScript version can be published.

The effect of this is that the RFC Editor first processes the ASCII version of the memo through to publication as an RFC. If the author wishes to submit a PostScript version at that point that matches the ASCII version (and the RFC Editor agrees that it does), then the PostScript version will be installed in the RFC repositories and announced to the community.

Due to various time pressures on the RFC Editorial staff the time elapsed between submission and publication can vary greatly. It is always acceptable to query (ping) the RFC Editor about the status of an RFC during this time (but not more than once a week). The two weeks preceding an IETF meeting are generally very busy, so RFCs submitted shortly before an IETF meeting are most likely to be published after the meeting.

3. Format Rules

To meet the distribution constraints, the following rules are established for the two allowed formats for RFCs: ASCII and PostScript.

The RFC Editor attempts to ensure a consistent RFC style. To do this the RFC Editor may choose to reformat the RFC submitted. It is much easier to do this if the submission matches the style of the most recent RFCs. Please do look at some recent RFCs and prepare yours in the same style.

You must submit an editable online document to the RFC Editor. The RFC Editor may require minor changes in format or style and will insert the actual RFC number.

Most of the RFCs are processed by the RFC Editor with the unix "nroff" program using a very simple set of the formatting commands (or "requests") from the "ms" macro package (see the appendix). If a memo submitted to be an RFC has been prepared by the author using nroff, it is very helpful to let the RFC Editor know that when it is submitted.

3a. ASCII Format Rules

The character codes are ASCII.

Each page must be limited to 58 lines followed by a form feed on a line by itself.

Each line must be limited to 72 characters followed by carriage return and line feed.

No overstriking (or underlining) is allowed.

These "height" and "width" constraints include any headers, footers, page numbers, or left side indenting.

Do not fill the text with extra spaces to provide a straight right margin.

Do not do hyphenation of words at the right margin.

Do not use footnotes. If such notes are necessary, put them at the end of a section, or at the end of the document.

Use single spaced text within a paragraph, and one blank line between paragraphs.

Note that the number of pages in a document and the page numbers on which various sections fall will likely change with reformatting. Thus cross references in the text by section number usually are easier to keep consistent than cross references by page number.

RFCs in ASCII Format may be submitted to the RFC Editor in e-mail messages (or as online files) in either the finished publication format or in NROFF. If you plan to submit a document in NROFF please consult the RFC Editor first.

3b. PostScript Format Rules

Standard page size is 8 1/2 by 11 inches.

Margin of 1 inch on all sides (top, bottom, left, and right).

Main text should have a point size of no less than 10 points with a line spacing of 12 points.

Footnotes and graph notations no smaller than 8 points with a line spacing of 9.6 points.

Three fonts are acceptable: Helvetica, Times Roman, and Courier. Plus their bold-face and italic versions. These are the three standard fonts on most PostScript printers.

Prepare diagrams and images based on lowest common denominator PostScript. Consider common PostScript printer functionality and

memory requirements.

The following PostScript commands should not be used:
initgraphics, erasepage, copypage, grestoreall, initmatrix,
initclip, banddevice, framedevice, nulldevice and renderbands.

Note that the number of pages in a document and the page numbers on which various sections fall will likely differ in the ASCII and the PostScript versions. Thus cross references in the text by section number usually are easier to keep consistent than cross references by page number.

These PostScript rules are likely to be changed and expanded as experience is gained.

RFCs in PostScript Format may be submitted to the RFC Editor in e-mail messages (or as online files). If you plan to submit a document in PostScript please consult the RFC Editor first.

Note that since the ASCII text version of the RFC is the primary version, the PostScript version must match the text version. The RFC Editor must decide if the PostScript version is "the same as" the ASCII version before the PostScript version can be published.

4. Headers and Footers

There is the first page heading, the running headers, and the running footers.

4a. First Page

Please see the front page of this memo for an example of the front page heading. On the first page there is no running header. The top of the first page has the following items:

Network Working Group

The traditional heading for the group that founded the RFC series. This appears on the first line on the left hand side of the heading.

Request for Comments: nnnn

Identifies this as a request for comments and specifies the number. Indicated on the second line on the left side. The actual number is filled in at the last moment before publication by the RFC Editor.

Author

The author's name (first initial and last name only) indicated on the first line on the right side of the heading.

Organization

The author's organization, indicated on the second line on the right side.

Date

This is the Month and Year of the RFC Publication. Indicated on the third line on the right side.

Updates or Obsoletes

If this RFC Updates or Obsoletes another RFC, this is indicated as third line on the left side of the heading.

Category

The category of this RFC, one of: Standards Track, Informational, or Experimental. This is indicated on the third (if there is no Updates or Obsoletes indication) or fourth line of the left side.

Title

The title appears, centered, below the rest of the heading.

If there are multiple authors and if the multiple authors are from multiple organizations the right side heading may have additional lines to accommodate them and to associate the authors with the organizations properly.

4b. Running Headers

The running header in one line (on page 2 and all subsequent pages) has the RFC number on the left (RFC NNNN), the (possibly a shortened form) title centered, and the date (Month Year) on the right.

4c. Running Footers

The running footer in one line (on all pages) has the author's last name on the left and the page number on the right ([Page N]).

5. Status Section

Each RFC must include on its first page the "Status of this Memo" section which contains a paragraph describing the type of the RFC.

The content of this section will be one of the three following statements.

Standards Track

"This document specifies an Internet standards track protocol for the Internet community, and requests discussion and suggestions for improvements. Please refer to the current edition of the "Internet Official Protocol Standards" (STD 1) for the standardization state and status of this protocol. Distribution of this memo is unlimited."

Experimental

"This memo defines an Experimental Protocol for the Internet community. This memo does not specify an Internet standard of any kind. Discussion and suggestions for improvement are requested. Distribution of this memo is unlimited."

Informational

"This memo provides information for the Internet community. This memo does not specify an Internet standard of any kind. Distribution of this memo is unlimited."

6. Introduction Section

Each RFC should have an Introduction section that (among other things) explains the motivation for the RFC and (if appropriate) describes the applicability of the protocol described.

Some example paragraphs are:

Protocol

This protocol is intended to provide the bla-bla service, and be used between clients and servers on host computers. Typically the clients are on workstation hosts and the servers on mainframe hosts.

or

This protocol is intended to provide the bla-bla service, and be used between special purpose units such as terminal servers or routers and a monitoring host.

Discussion

The purpose of this RFC is to focus discussion on particular problems in the Internet and possible methods of solution. No proposed solutions in this document are intended as standards for the Internet. Rather, it is hoped that a general consensus will emerge as to the appropriate solution to such problems, leading eventually to the adoption of standards.

Interest

This RFC is being distributed to members of the Internet community in order to solicit their reactions to the proposals contained in it. While the issues discussed may not be directly relevant to the research problems of the Internet, they may be interesting to a number of researchers and implementers.

Status Report

In response to the need for maintenance of current information about the status and progress of various projects in the Internet community, this RFC is issued for the benefit of community members. The information contained in this document is accurate as of the date of publication, but is subject to change. Subsequent RFCs will reflect such changes.

These paragraphs need not be followed word for word, but the general intent of the RFC must be made clear.

7. References Section

Nearly all RFCs contain citations to other documents, and these are listed in a References section near the end of the RFC. There are many styles for references, and the RFCs have one of their own. Please follow the reference style used in recent RFCs. See the reference section of this RFC for an example. Please note that for protocols that have been assigned STD numbers, the STD number must be included in the reference.

8. Security Considerations Section

All RFCs must contain a section near the end of the document that discusses the security considerations of the protocol or procedures that are the main topic of the RFC.

9. Author's Address Section

Each RFC must have at the very end a section giving the author's address, including the name and postal address, the telephone number, (optional: a FAX number) and the Internet e-mail address.

10. Relation to other RFCs

Sometimes an RFC adds information on a topic discussed in a previous RFC or completely replaces an earlier RFC. There are two terms used for these cases respectively, UPDATES and OBSOLETES. A document that obsoletes an earlier document can stand on its own. A document that merely updates an earlier document cannot stand on its own; it is something that must be added to or inserted into the previously existing document, and has limited usefulness independently. The terms SUPERSEDES and REPLACES are no longer used.

UPDATES

To be used as a reference from a new item that cannot be used alone (i.e., one that supplements a previous document), to refer to the previous document. The newer publication is a part that will supplement or be added on to the existing document; e.g., an addendum, or separate, extra information that is to be added to the original document.

OBSOLETES

To be used to refer to an earlier document that is replaced by this document. This document contains either revised information, or else all of the same information plus some new information, however extensive or brief that new information is; i.e., this document can be used alone, without reference to the older document.

For example:

On the Assigned Numbers RFCs the term OBSOLETES should be used since the new document actually incorporate new information (however brief) into the text of existing information and is more up-to-date than the older document, and hence, replaces it and makes it OBSOLETE.

In lists of RFCs or the RFC-Index (but not on the RFCs themselves) the following may be used with early documents to point to later documents.

OBSOLETE-BY

To be used to refer to the newer document(s) that replaces the older document.

UPDATED-BY

To be used to refer to the newer section(s) which are to be added to the existing, still used, document.

11. Protocol Standards Process

See the current "Internet Official Protocol Standards" (STD 1) memo for the definitive statement on protocol standards and their publication [1].

The established procedure is that when the IESG completes work on a document that is to become a standards track RFC the communication will be from the Secretary of the IESG to the RFC Editor. Generally, the documents in question are Internet Drafts. The communication usually cites the exact Internet Draft (by file name) in question. The RFC Editor must assume that only that file is to be processed to become the RFC. If the authors have small corrections to the text, they should be sent to the RFC Editor separately (or as a "diff"), do not send a new version of the document.

12. Contact

To contact the RFC Editor send an email message to

"RFC-Editor@ISI.EDU".

13. Distribution Lists

The RFC announcements are distributed via two mailing lists: the "IETF-Announce" list, and the "RFC-DIST" list. You don't want to be on both lists.

To join (or quit) the IETF-Announce list send a message to IETF-Request@cnri.reston.va.us.

To join (or quit) the RFC-DIST list send a message to RFC-Request@NIC.DDN.MIL.

14. RFC Index

Several organizations maintain RFC Index files, generally using the file name "rfc-index.txt". The contents of such a file copied from one site may not be identical to that copied from another site.

15. Security Considerations

This RFC raises no security issues (however, see Section 6).

16. References

[1] Postel, J., "Internet Official Protocol Standards", STD 1, RFC 1540, Internet Architecture Board, October 1993.

17. Author's Address

Jon Postel
USC/Information Sciences Institute
4676 Admiralty Way
Marina del Rey, CA 90292

Phone: 310-822-1511
Fax: 310-823-6714
EMail: Postel@ISI.EDU

18. Appendix - RFC "nroff macros"

Generally, we use the very simplest nroff features. We use the "ms" macros. So, "nroff -ms input-file > output-file". However, we could not get nroff to do the right thing about putting a form feed after the last visible line on a page and no extra line feeds before the first visible line of the next page. We want:

```
last visible line on page i
^L
first visible line on page i+1
```

So, we invented some hacks to fix this including a "sed" script called "fix.sh" and a "c" program we called "pg" (pg is called from fix). So the command to process the file becomes:

```
nroff -ms input-file | fix.sh > output-file
```

Now as to the nroff features we actually use, I'll append a sample memo, prepared in RFC style.

The sed script fix.sh is:

```
sed -e 's/FORMFEED\[Page/      \[Page/' $* | pg -n5
```

The pg program is:

```
~~~Beginning of pg program~~~
```

```
/* $Header$
 *
 * Remove N lines following any line that contains a form feed (^L).
 * (Why can't this be done with awk or sed?)
 *
 * OPTION:
 *   -n#      Number of lines to delete following each ^L (0 default).
 * $Log$
 */
#include <stdio.h>
#define FORM_FEED      '\f'
#define OPTION        "n:N:"      /* for getopt() */

extern char *optarg;
extern int optind;

main(argc, argv)
int      argc;
char    *argv[];
```

```

{
    int    c,                /* next input char */
          nlines = 0;       /* lines to delete after ^L */
    void  print_and_delete(); /* print line starting with ^L,
                              then delete N lines */

/* Process option (-nlines) */

    while ((c = getopt(argc, argv, OPTION)) != EOF)
        switch(c)
        {
            case 'n' :
            case 'N' : nlines = atoi(optarg);
                       break;
        }
/* READ AND PROCESS CHARS */

    while ((c = getchar()) != EOF)
        if (c == FORM_FEED)
            print_and_delete(nlines); /* remove N lines after this one */
        else
            putchar(c);               /* we write the form feed */
    exit(0);
}

/* Print rest of line, then delete next N lines. */

void print_and_delete(n)
int  n;                /* nbr of lines to delete */
{
    int  c,            /* next input char */
          cntr = 0;    /* count of deleted lines */

    while ((c = getchar()) != '\n') /* finish current line */
        putchar(c);
    putchar('\n');                 /* write the last CR */
    putchar(FORM_FEED);

    for ( ; cntr < n; cntr++)
        while ((c = getchar()) != '\n')
            if (c == EOF)
                exit(0);          /* exit on EOF */
        putchar(c);              /* write that last CR */
}

~~~~End of pg program~~~~

```

.pl 10.0i
.po 0
.ll 7.2i
.lt 7.2i
.nr LL 7.2i
.nr LT 7.2i
.ds LF Waitzman
.ds RF FORMFEED[Page %]
.ds CF
.ds LH RFC 1149
.ds RH 1 April 1990
.ds CH IP Datagrams on Avian Carriers
.hy 0
.ad 1
.in 0

Network Working Group
Request for Comments: 1149

D. Waitzman
BBN STC
1 April 1990

.ce
A Standard for the Transmission of IP Datagrams on Avian Carriers

.ti 0
Status of this Memo

.fi
.in 3
This memo describes an experimental method for the encapsulation of IP datagrams in avian carriers. This specification is primarily useful in Metropolitan Area Networks. This is an experimental, not recommended standard. Distribution of this memo is unlimited.

.ti 0
Overview and Rational

Avian carriers can provide high delay, low throughput, and low altitude service. The connection topology is limited to a single point-to-point path for each carrier, used with standard carriers, but many carriers can be used without significant interference with each other, outside of early spring. This is because of the 3D ether space available to the carriers, in contrast to the 1D ether used by IEEE802.3. The carriers have an intrinsic collision avoidance system, which increases availability. Unlike some network technologies, such as packet radio, communication is not limited to line-of-sight distance. Connection oriented service is available in some cities, usually based upon a central hub topology.

.ti 0
Frame Format

The IP datagram is printed, on a small scroll of paper, in hexadecimal, with each octet separated by whitestuff and blackstuff. The scroll of paper is wrapped around one leg of the avian carrier. A band of duct tape is used to secure the datagram's edges. The bandwidth is limited to the leg length. The MTU is variable, and paradoxically, generally increases with increased carrier age. A typical MTU is 256 milligrams. Some datagram padding may be needed.

Upon receipt, the duct tape is removed and the paper copy of the datagram is optically scanned into a electronically transmittable form.

.ti 0
Discussion

Multiple types of service can be provided with a prioritized pecking order. An additional property is built-in worm detection and eradication. Because IP only guarantees best effort delivery, loss of a carrier can be tolerated. With time, the carriers are self-regenerating. While broadcasting is not specified, storms can cause data loss. There is persistent delivery retry, until the carrier drops. Audit trails are automatically generated, and can often be found on logs and cable trays.

.ti 0
Security Considerations

.in 3
Security is not generally a problem in normal operation, but special measures must be taken (such as data encryption) when avian carriers are used in a tactical environment.

.ti 0
Author's Address

.nf
David Waitzman
BBN Systems and Technologies Corporation
BBN Labs Division
10 Moulton Street
Cambridge, MA 02238

Phone: (617) 873-4323

EMail: dwaitzman@BBN.COM