                Guidance on Interoperation and Implementation Reports
                      for Advancement to Draft Standard

Abstract

   Advancing a protocol to Draft Standard requires documentation of the
   interoperation and implementation of the protocol. Historic reports
   have varied widely in form and level of content and there is little
   guidance available to new report preparers. This document updates
   the existing processes and provides more detail on what is
   appropriate in an interoperability and implementation report.

Status of This Memo

Copyright and License Notice

Table of Contents

1. Introduction

   The Draft Standard level, and requirements for standards to meet it,
   are described in [RFC2026]. For Draft Standard, not only must two
   implementations interoperate, but also documentation (the report)
   must be provided to the IETF. The entire paragraph covering this
   documentation reads:

      The Working Group chair is responsible for documenting the
      specific implementations which qualify the specification for Draft
      or Internet Standard status along with documentation about testing
      of the interoperation of these implementations. The documentation
      must include information about the support of each of the
      individual options and features. This documentation should be
      submitted to the Area Director with the protocol action request.
      (see Section 6)

   Moving documents along the standards track can be an important signal
   to the user and implementor communities, and the process of
   submitting a standard for advancement can help improve that standard
   or the quality of implementations that participate. However, the
   barriers seem to be high for advancement to Draft Standard, or at the
   very least confusing. This memo may help in guiding people through
   one part of advancing specifications to Draft Standard. It also
   changes some of the requirements made in RFC 2026 in ways that are
   intended to maintain or improve the quality of reports while reducing
   the burden of creating them.

Having and demonstrating sufficient interoperability is a gating
requirement for advancing a protocol to Draft Standard. Thus, the
primary goal of an implementation report is to convince the IETF and
the IESG that the protocol is ready for Draft Standard. This goal
can be met by summarizing the interoperability characteristics and by
providing just enough detail to support that conclusion. Side
benefits may accrue to the community creating the report in the form
of bugs found or fixed in tested implementations, documentation that
can help future implementors, or ideas for other documents or future
revisions of the protocol being tested.

Different kinds of documentation are appropriate for widely deployed
standards than for standards that are not yet deployed. Different
test approaches are appropriate for standards that are not typical
protocols: languages, formats, schemas, etc. This memo discusses how
reports for these standards may vary in Section 5.

Implementation should naturally focus on the final version of the
RFC. If there's any evidence that implementations are interoperating
based on Internet-Drafts or earlier versions of the specification, or
if interoperability was greatly aided by mailing list clarifications,
this should be noted in the report.

The level of detail in reports accepted in the past has varied
widely. An example of a submitted report that is not sufficient for
demonstrating interoperability is (in its entirety): "A partial list
of implementations include: Cray SGI Netstar IBM HP Network Systems
Convex". This report does not state how it is known that these
implementations interoperate (was it through public lab testing?
internal lab testing? deployment?). Nor does it capture whether
implementors are aware of, or were asked about, any features that
proved to be problematic. At a different extreme, reports have been
submitted that contain a great amount of detail about the test
methodology, but relatively little information about what worked and
what failed to work.

This memo is intended to clarify what an implementation report should
contain and to suggest a reasonable form for most implementation
reports. It is not intended to rule out good ideas. For example,
this memo can't take into account all process variations such as
documents going to Draft Standard twice, nor can it consider all
types of standards. Whenever the situation varies significantly from
what's described here, the IESG uses judgement in determining whether
an implementation report meets the goals above.

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT",
"SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this
document are to be interpreted as described in BCP 14 [RFC2119].

2. Content Requirements

   The implementation report MUST identify the author of the report, who
   is responsible for characterizing the interoperability quality of the
   protocol. The report MAY identify other contributors (testers, those
   who answered surveys, or those who contributed information) to share
   credit or blame. The report MAY provide a list of report reviewers
   who corroborate the characterization of interoperability quality, or
   name an active working group (WG) that reviewed the report.

   Some of the requirements of RFC 2026 are relaxed with this update:

   o The report MAY name exactly which implementations were tested. A
      requirement to name implementations was implied by the description
      of the responsibility for "documenting the specific
      implementations" in RFC 2026. However, note that usually
      identifying implementations will help meet the goals of
      implementation reports. If a subset of implementations was tested
      or surveyed, it would also help to explain how that subset was
      chosen or self-selected. See also the note on implementation
      independence below.

   o The report author MAY choose an appropriate level of detail to
      document feature interoperability, rather than document each
      individual feature. See note on granularity of features below.

   o A contributor other than a WG chair MAY submit an implementation
      report to an Area Director (AD).

   o Optional features that are not implemented, but are important and
      do not harm interoperability, MAY, exceptionally and with approval
      of the IESG, be left in a protocol at Draft Standard. See
      Section 5.6 for documentation requirements and an example of where
      this is needed.

   Note: Independence of implementations is mentioned in the RFC 2026
         requirements for Draft Standard status. Independent
         implementations should be written by different people at
         different organizations using different code and protocol
         libraries. If it's necessary to relax this definition, it can
         be relaxed as long as there is evidence to show that success is
         due more to the quality of the protocol than to out-of-band
         understandings or common code. If there are only two
         implementations of an undeployed protocol, the report SHOULD
         identify the implementations and their "genealogy" (which
         libraries were used or where the codebase came from). If there
         are many more implementations, or the protocol is in broad
         deployment, it is not necessary to call out which two of the

implementations demonstrated interoperability of each given
feature -- a reader may conclude that at least some of the
implementations of that feature are independent.

   Note: The granularity of features described in a specification is
         necessarily very detailed. In contrast, the granularity of an
         implementation report need not be as detailed. A report need
         not list every "MAY", "SHOULD", and "MUST" in a complete matrix
         across implementations. A more effective approach might be to
         characterize the interoperability quality and testing approach,
         then call out any known problems in either testing or
         interoperability.

3. Format

   The format of implementation and interoperability reports MUST be
   ASCII text with line breaks for readability. As with Internet-
   Drafts, no 8-bit characters are currently allowed. It is acceptable,
   but not necessary, for a report to be formatted as an Internet-Draft.

   Here is a simple outline that an implementation report MAY follow in
   part or in full:

   Title: Titles of implementation reports are strongly RECOMMENDED to
      contain one or more RFC number for consistent lookup in a simple
      archive. In addition, the name or a common mnemonic of the
      standard should be in the title. An example might look like
      "Implementation Report for the Example Name of Some Protocol
      (ENSP) RFC XXXX".

   Author: Identify the author of the report.

   Summary: Attest that the standard meets the requirements for Draft
      Standard and name who is attesting it. Describe how many
      implementations were tested or surveyed. Quickly characterize the
      deployment level and where the standard can be found in
      deployment. Call out, and if possible, briefly describe any
      notably difficult or poorly interoperable features and explain why
      these still meet the requirement. Assert any derivative
      conclusions: if a high-level system is tested and shown to work,
      then we may conclude that the normative requirements of that
      system (all sub-system or lower-layer protocols, to the extent
      that a range of features is used) have also been shown to work.

   Methodology: Describe how the information in the report was
      obtained. This should be no longer than the summary.

   Exceptions: This section might read "Every feature was implemented,
      tested, and widely interoperable without exception and without
      question". If that statement is not true, then this section
      should cover whether any features were thought to be problematic.
      Problematic features need not disqualify a protocol from Draft
      Standard, but this section should explain why they do not (e.g.,
      optional, untestable, trace, or extension features). See the
      example in Section 6.2.

   Detail sections: Any other justifying or background information can
      be included here. In particular, any information that would have
      made the summary or methodology sections more than a few
      paragraphs long may be created as a detail section and referenced.

      In this section, it would be good to discuss how the various
      considerations sections played out. Were the security
      considerations accurate and dealt with appropriately in
      implementations? Was real internationalization experience found
      among the tested implementations? Did the implementations have
      any common monitoring or management functionality (although note
      that documenting the interoperability of a management standard
      might be separate from documenting the interoperability of the
      protocol itself)? Did the IANA registries or registrations, if
      any, work as intended?

   Appendix sections: It's not necessary to archive test material such
      as test suites, test documents, questionnaire text, or
      questionnaire responses. However, if it's easy to preserve this
      information, appendix sections allow readers to skip over it if
      they are not interested. Preserving detailed test information can
      help people doing similar or follow-on implementation reports, and
      can also help new implementors.

4. Feature Coverage

   What constitutes a "feature" for the purposes of an interoperability
   report has been frequently debated. Good judgement is required in
   finding a level of detail that adequately demonstrates coverage of
   the requirements. Statements made at too high a level will result in
   a document that can't be verified and hasn't adequately challenged
   that the testing accidentally missed an important failure to
   interoperate. On the other hand, statements at too fine a level
   result in an exponentially exploding matrix of requirement
   interaction that overburdens the testers and report writers. The
   important information in the resulting report would likely be hard to
   find in the sea of detail, making it difficult to evaluate whether
   the important points of interoperability have been addressed.

   The best interoperability reports will organize statements of
   interoperability at a level of detail just sufficient to convince the
   reader that testing has covered the full set of requirements and in
   particular that the testing was sufficient to uncover any places
   where interoperability does not exist. Reports similar to that for
   RTP/RTCP (an excerpt appears below) are more useful than an
   exhaustive checklist of every normative statement in the
   specification.

        10. Interoperable exchange of receiver report packets.

            o PASS: Many implementations, tested UCL rat with vat,
                     Cisco IP/TV with vat/vic.

        11. Interoperable exchange of receiver report packets when
            not receiving data (ie: the empty receiver report
            which has to be sent first in each compound RTCP packet
            when no-participants are transmitting data).

            o PASS: Many implementations, tested UCL rat with vat,
                     Cisco IP/TV with vat/vic.

         ...

         8. Interoperable transport of RTP via TCP using the
            encapsulation defined in the audio/video profile

            o FAIL: no known implementations. This has been
                     removed from the audio/video profile.

                             Excerpts from
         http://www.ietf.org/iesg/implementation/report-avt-rtp-rtcp.txt

   Consensus can be a good tool to help determine the appropriate level
   for such feature descriptions. A working group can make a strong
   statement by documenting its consensus that a report sufficiently
   covers a specification and that interoperability has been
   demonstrated.

5. Special Cases

5.1. Deployed Protocols

   When a protocol is deployed, results obtained from laboratory testing
   are not as useful to the IETF as learning what is actually working in
   deployment. To this end, it may be more informative to survey
   implementors or operators. A questionnaire or interview can elicit
   information from a wider number of sources. As long as it is known
   that independent implementations can work in deployment, it is more
   useful to discover what problems exist, rather than gather long and
   detailed checklists of features and options.

5.2. Undeployed Protocols

   It is appropriate to provide finer-grained detail in reports for
   protocols that do not yet have a wealth of experience gained through
   deployment. In particular, some complicated, flexible or powerful
   features might show interoperability problems when testers start to
   probe outside the core use cases. RFC 2026 requires "sufficient
   successful operational experience" before progressing a standard to
   Draft, and notes that:

      Draft Standard may still require additional or more widespread
      field experience, since it is possible for implementations based
      on Draft Standard specifications to demonstrate unforeseen
      behavior when subjected to large-scale use in production
      environments.

   When possible, reports for protocols without much deployment
   experience should anticipate common operational considerations. For
   example, it would be appropriate to put additional emphasis on
   overload or congestion management features the protocol may have.

5.3. Schemas, Languages, and Formats

   Standards that are not on-the-wire protocols may be special cases for
   implementation reports. The IESG SHOULD use judgement in what kind
   of implementation information is acceptable for these kinds of
   standards. ABNF (RFC 4234) is an example of a language for which an
   implementation report was filed: it is interoperable in that
   protocols are specified using ABNF and these protocols can be
   successfully implemented and syntax verified. Implementations of
   ABNF include the RFCs that use it as well as ABNF checking software.
   Management Information Base (MIB, [RFC3410]) modules are sometimes
   documented in implementation reports, and examples of that can be
   found in the archive of implementation reports.

The interoperability reporting requirements for some classes of
documents may be discussed in separate documents. See [METRICSTEST]
for example.

5.4. Multiple Contributors, Multiple Implementation Reports

If it's easiest to divide up the work of implementation reports by
implementation, the result -- multiple implementation reports -- MAY
be submitted to the sponsoring Area Director one-by-one. Each report
might cover one implementation, including:

   identification of the implementation;

   an affirmation that the implementation works in testing (or
   better, in deployment);

   whether any features are known to interoperate poorly with other
   implementations;

   which optional or required features are not implemented (note that
   there are no protocol police to punish this disclosure, we should
   instead thank implementors who point out unimplemented or
   unimplementable features especially if they can explain why); and

   who is submitting this report for this implementation.

These SHOULD be collated into one document for archiving under one
title, but can be concatenated trivially even if the result has
several summary sections or introductions.

5.5. Test Suites

Some automated tests, such as automated test clients, do not test
interoperability directly. When specialized test implementations are
necessary, tests can at least be constructed from real-world protocol
or document examples. For example:

- ABNF [RFC4234] itself was tested by combining real-world examples
   -- uses of ABNF found in well-known RFCs -- and feeding those
   real-world examples into ABNF checkers. As the well-known RFCs
   were themselves interoperable and in broad deployment, this served
   as both a deployment proof and an interoperability proof.
   [RFC4234] progressed from Proposed Standard through Draft Standard
   to Standard and is obsoleted by [RFC5234].

- Atom [RFC4287] clients might be tested by finding that they
  consistently display the information in a test Atom feed,
  constructed from real-world examples that cover all the required
  and optional features.

- MIB modules can be tested with generic MIB browsers, to confirm
  that different implementations return the same values for objects
  under similar conditions.

As a counter-example, the automated WWW Distributed Authoring and
Versioning (WebDAV) test client Litmus
(http://www.webdav.org/neon/litmus/) is of limited use in
demonstrating interoperability for WebDAV because it tests
completeness of server implementations and simple test cases. It
does not test real-world use or whether any real WebDAV clients
implement a feature properly or at all.

5.6. Optional Features, Extensibility Features

Optional features need not be shown to be implemented everywhere.
However, they do need to be implemented somewhere, and more than one
independent implementation is required. If an optional feature does
not meet this requirement, the implementation report must say so and
explain why the feature must be kept anyway versus being evidence of
a poor-quality standard.

Extensibility points and versioning features are particularly likely
to need this kind of treatment. When a protocol version 1 is
released, the protocol version field itself is likely to be unused.
Before any other versions exist, it can't really be demonstrated that
this particular field or option is implemented.

6. Examples

Some good, extremely brief, examples of implementation reports can be
found in the archives:

    http://www.ietf.org/iesg/implementation/report-ppp-lcp-ext.html

    http://www.ietf.org/iesg/implementation/report-otp.html

In some cases, perfectly good implementation reports are longer than
necessary, but may preserve helpful information:

    http://www.ietf.org/iesg/implementation/report-rfc2329.txt

    http://www.ietf.org/iesg/implementation/report-rfc4234.txt

6.1. Minimal Implementation Report

   A large number of SMTP implementations support SMTP pipelining,
   including: (1) Innosoft's PMDF and Sun's SIMS. (2) ISODE/
   MessagingDirect's PP. (3) ISOCOR's nPlex. (4) software.com's
   post.office. (5) Zmailer. (6) Smail. (7) The SMTP server in
   Windows 2000. SMTP pipelining has been widely deployed in these
   and other implementations for some time, and there have been no
   reported interoperability problems.

   This implementation report can also be found at
   http://www.ietf.org//iesg/implementation/report-smtp-pipelining.txt
   but the entire report is already reproduced above. Since SMTP
   pipelining had no interoperability problems, the implementation
   report was able to provide all the key information in a very terse
   format. The reader can infer from the different vendors and
   platforms that the codebases must, by and in large, be independent.

   This implementation report would only be slightly improved by a
   positive affirmation that there have been probes or investigations
   asking about interoperability problems rather than merely a lack of
   problem reports, and by stating who provided this summary report.

6.2. Covering Exceptions

   The RFC2821bis (SMTP) implementation survey asked implementors what
   features were not implemented. The VRFY and EXPN commands showed up
   frequently in the responses as not implemented or disabled. That
   implementation report might have followed the advice in this
   document, had it already existed, by justifying the interoperability
   of those features up front or in an "exceptions" section if the
   outline defined in this memo were used:

   VRFY and EXPN commands are often not implemented or are disabled.
   This does not pose an interoperability problem for SMTP because
   EXPN is an optional features and its support is never relied on.
   VRFY is required, but in practice it is not relied on because
   servers can legitimately reply with a non-response. These
   commands should remain in the standard because they are sometimes
   used by administrators within a domain under controlled
   circumstances (e.g. authenticated query from within the domain).
   Thus, the occasional utility argues for keeping these features,
   while the lack of problems for end-users means that the
   interoperability of SMTP in real use is not in the least degraded.

7. Security Considerations

   This memo introduces no new security considerations.

8. References

8.1. Normative References

   [RFC2119]   Bradner, S., "Key words for use in RFCs to Indicate
                   Requirement Levels", BCP 14, RFC 2119, March 1997.

8.2. Informative References

   [METRICSTEST]   Bradner, S. and V. Paxson, "Advancement of metrics
                   specifications on the IETF Standards Track", Work
                   in Progress, July 2007.

   [RFC2026]   Bradner, S., "The Internet Standards Process --
                   Revision 3", BCP 9, RFC 2026, October 1996.

   [RFC3410]   Case, J., Mundy, R., Partain, D., and B. Stewart,
                   "Introduction and Applicability Statements for
                   Internet-Standard Management Framework", RFC 3410,
                   December 2002.

   [RFC4234]   Crocker, D., Ed. and P. Overell, "Augmented BNF for
                   Syntax Specifications: ABNF", RFC 4234, October 2005.

   [RFC4287]   Nottingham, M., Ed. and R. Sayre, Ed., "The Atom
                   Syndication Format", RFC 4287, December 2005.

   [RFC5234]   Crocker, D. and P. Overell, "Augmented BNF for Syntax
                   Specifications: ABNF", STD 68, RFC 5234, January 2008.

Authors' Addresses

   Lisa Dusseault
   Messaging Architects

   EMail: lisa.dusseault@gmail.com

   Robert Sparks
   Tekelec
   17210 Campbell Road
   Suite 250
   Dallas, Texas 75254-4203
   USA

   EMail: RjS@nostrum.com