Network Working Group
Internet-Draft
Expires: April 14, 2006

L. Masinter
Adobe Systems
October 11, 2005

Formalizing IETF Interoperability Reporting
draft-ietf-newtrk-interop-reports-00.txt

Status of this Memo

Copyright Notice

Abstract

This document suggests another way of reforming IETF standards process by formalizing the mechanism for interoperability reporting, as a way of facilitating standards development. It establishes two kinds of reports: a 'Protocol Feature Set', which lays out the set of features from IETF specifications that constitute a protocol, and a 'Protocol Implementation Report', which is submitted by an individual or group to report on implementation and interoperability testing.

1. Introduction

The basic idea is to create formal structures for

- ("Protocol Feature Set") Describing the set of specifications, and the features within them, that constitute a single "protocol", from the point of view of testing interoperability. (See below for format & publication process.)
- ("Protocol Implementation Report") Creating a standard way that individuals can report on implementation and interoperability testing of a protocol. (See below for format & publication process.)

These structures can be used to enhance the IETF standards process in the following ways:

- Working groups (or individuals) preparing specifications for new protocols may also prepare the initial Protocol Feature Set. The IETF should publish these if they represent rough consensus.
- Working groups preparing specifications for updating existing protocols or adding new options of features to an existing protocol may prepare a proposed extension to an existing published Protocol Feature Set. Again, updated Protocol Feature Sets that represent community (rough) consensus should be published.
- Individuals or groups who have an implementation of a protocol, and those who have tested interoperability between independent implementations may prepare implementation reports (which may include reports of successful interoperability).
- Implementation reports may contain comments about existing specifications. Groups interested in updating existing specifications to facilitate their advancement in standards status may use comments within implementation reports to give weight to "running code"; they may use the lack of implementation of particular features as motiviation for removal of those features in subsequent updates.
- The IETF may use the existence of reports of successful interoperability by multiple independent implementations of every feature within a specification as evidence for advancing that specification. Note that specifications may require updates in order to make them suitable for advancement, as in current practice.
- Implementation reports may also include assertions about widespread deployment of the implementations, and the IETF may use these reports as part of the basis for judgement of widespread deployment of protocol implementations as a basis for advancement of specifications.

2. Format of Protocol Feature Set

Requirements:

- List of referenced technical specifications.
- List of features, where a feature is a specification with chapter, section, paragraph, or quoted text.
- A feature description may contain additional explanatory text, to clarify or otherwise elaborate the feature.
- A feature description should indicate whether implementation is REQUIRED or optional for the protocol.
- Protocols may define multiple roles (e.g., client/server/proxy). Protocol Feature Set can include sets of roles, and feature specifications can identify the roles for which the feature is appropriate.
- May include references to other Protocol Feature Sets which are REQUIRED or OPTIONAL
- Could be specified as an XML-based format, with text format automatically derived, and both XML and text published.

3. Scope and Granularity of Protocol Feature Set

There is a great deal of judgement needed about how details to get in the protocol feature set. At the coarsest granularity, a feature set could have a single feature, which listed a single specification, at least for protocols with no options. How difficult it is to create the Protocol Feature Set depends a great deal on the quality of the original technical specifications. Protocol Feature Sets require rough consensus before they are published. However, rough consensus may be judged by the willingness of implementors to prepare Protocol Implementation Reports using the Protocol Feature Set framework.

4. Format of Protocol Implementation Report

Requirements:

- May not cover entire PFS.
- Identity of implementation. Relationship to other previously reported implementations, if any.
- If any IPR noted for any technical specification referenced in PFS, relationship of source of implementation to owner of IPR and/or other exercises of license process.
- Optionally, assertions about deployment
- Version history, for implementation report updates
- Identity of author, relationship to implementation, IPR.
- If implementation report has been reviewed by someone else (working group chair, interoperability event host), identity of reviewer.
- For each feature:

    * Was feature implemented?
    * If so, has feature been sucessfully tested as interoperable with at least one independent implementation?
    * Optionally, the identity of the other implementations against which interoperability was successfully tested
    * For asymmetric protocols (e.g., client/server, or different roles), repeat for each role played.
    * Optionally, a short comment about the way in which the feature had to be interpreted to be interoperable. This shouldn't be a place to publish a long article, however.
    * Could be specified as an XML representation, with text format automatically derived, to facilitate tools for automatic merging and summarizing implementation reports.
- If Protocol Feature Set contains references to other Protocol Feature Sets, the Protocol Implementation Report may also reference corresponding Protocol Implementation Reports.
- QUESTION: Is it a requirement to allow for anonomous implementation reports, where the implementation is not specifically identified? In some cases, interop events allow for this because product managers don't want competitors to use their implemetation reports in negative marketing.

5. Process for publication of Protocol Feature Set

Authored against template. Should be reviewed by working group (if active) or IESG. Perhaps IETF last call not necessary? After all, proof is in whether there are actually any implementations willing to report on it.

Updates to a Protocol Feature Set could be proposed by listing the proposed delta. In general, if specifications change, feature sets should be extended, not updated, unless there was some mistake. That is, the "feature" corresponds to the documented feature.

6. Process for publication of Protocol Implementation Report

Preferably produced by someone responsible for the implementation. Perhaps could be reported by someone else, as long as actual implementor can update. May be updated at any time, old reports are still available. Updates can include new information or correction to old information. Perhaps there could be a mechanism for publishing comments on implementation reports.

7. Tools for viewing Protocol Feature Sets and Protocol Implementation Reports

If the format for submission of both kinds of reports are in XML, there could be tools for generating HTML and plain text versions of these reports.

8. Tools for combining information for combined reports

To facilitate seeing the "whole picture", it would be useful to have some tools that would take the information in the published Protocol Feature Sets and Protocol Implementation Reports and generate implementation reports that could summarize, for each feature of a given protocol,

- Whether it was not implemented
- How many implementations there were.
- How many implementations reported interoperability with an independent implementation.
- The list of all comments about the feature.

9. Updating IETF processes

Once we have provided a way of formalized interoperability reporting, we could consider ways in which IETF RFC 2026 standards process could be updated to make use of these. For example, we could consider automating progression of specifications from Proposed Standard to Draft Standard if sufficient combined interoperability reports existed. We would need to be clear about the minimum requirement for implementation reports. Alternatively, we could consider removing "Draft Standard" as a formal approval step; and instead (automatically) report which Standards Track documents had adequate interoperability reports. Since the IESG does not currently evaluate the accuracy of interoperability reporting, it would make it clearer that the judgment about the maturity of a protocol specification and its interoperable implementation is left to the reader of the specification and its interoperability reports. This would also simplify the decisions about "downreference", since references from widely implemented specifications to those with mixed implementation would not result in confusion. Finally, we could change the judgment of "full standard" from a judgement about the protocol specification to a judgement about what constitutes "widespread deployment" and whether the implementations reported had reached that status.

10. Comparison with ISD and SRD

Note that this section will be removed if this proposal advances.

The idea for formalizing interoperability reporting was based on the

ideas from ISDs and SRDs that we should have a single document that pulls together all of the specifications of a single "protocol". However, basing the full description of what constitutes a single "protocol" on the operational need to test interoperability creates a better justification for putting energy into the task, motivates a different category of individuals to work on it, and gives it an operational criteria for judging success.

I imagine that a PFS wouldn't take much more work to author than an ISD.

11. Acknowledgements

Thanks to Sam and others who helped flesh out the idea.

12. Informative References

Author's Address

Larry Masinter
Adobe Systems
345 Park Ave
San Jose, CA 95110
US

Phone: +1 408 536 3024
Email: LMM@acm.org
URI: http://larry.masinter.net

Intellectual Property Statement

Disclaimer of Validity

Copyright Statement

Acknowledgment

Funding for the RFC Editor function is currently provided by the Internet Society.