

# What makes for a quality RFC?

An invited talk to the MPLS WG

Adrian Farrel

[adrian@olddog.co.uk](mailto:adrian@olddog.co.uk)

IETF-89 London, March 2014

# Context

- The MPLS chairs asked me to talk about what I look for when reviewing drafts as responsible AD
  - I review every MPLS WG draft:
    - shortly after the WG requests publication
    - before I issue IETF Last Call
    - to find issues and problems that might be found in IETF Last Call or IESG review
    - to make sure that I am happy to support the document in discussions with the IESG
  - I typically find a variety of issues
    - Minor editorial
    - Major editorial (clarity, motivation, description of behavior)
    - Minor technical (bits missing, typographical)
    - Major technical (I don't think it works)
  - Recent stats show a lot of docs need work after my review

# Some rough stats

- Categories
  - No issues found
  - Minor issues can be taken in IETF last call
  - Manageable issues just need a new revision
  - Major issues mean returning the I-D to the WG
    - Includes additional WG last call

Publication Requests	24*
No issues	2
Minor issues	2
New revision needed	12
Returned to WG	5

\* I authored three drafts not shown in other rows

# Compare with Other WGs?

- Far more documents from the MPLS WG
- Far more comments and issues from me on MPLS WG documents
- What is going on?
  - I know the topic better?
  - Too much work for the WG to focus and review?
  - Just not enough emphasis on quality?
  - Documents are pre-implementation?
  - I have a personal grudge against you?

# Objectives of this talk

- Reduce my work load 😊
  - I really would prefer to read an I-D and say “yes”
  - You should be applying the polish, not me
- Speed up the processing of your drafts
  - My review, the emails, the discussion, the changes, the subsequent additional WG last call all take time
- Stop surprising you
  - It is not practical for me to review I-Ds earlier in the process, but I can tell you what I will look for
- Generally move drafts “up a category”

# Format is important

- Reduce work done by RFC Editor
  - Stop them introducing accidental errors
  - Make sure that what is published is what you intended
- Usually only minor editorial issues
  - But these are easy to check and there is no excuse for not just getting it right
- Resources
  - idnits (<http://www.ietf.org/tools/idnits/>)
  - RFC Editor guidance (<http://www.rfc-editor.org/styleguide.html>)
  - RFC Editor tutorial (<http://www.ietf.org/edu/documents/tutorial76.pdf>)

# Essential ingredients 1 of 2

- Security section
  - Don't just write the least you can get away with!
  - Think about attacks on and using your protocol
    - Are you using an old protocol in a new way?
    - Are you exposing new information for snooping?
    - You want your protocol to be strong and safe
  - Have you used and referenced existing security docs (such as RFC 5920, RFC 6941, and RFC 6952?)
- Resources
  - RFC 3552 (guidelines == requirements!)
  - RFC 4107 (when do you need to change your keys?)
  - Early Security Directorate reviews
  - Specific requests for help

# Essential ingredients 2 of 2

- Manageability section
  - Not a requirement, but have you thought about it and said so?
  - What to configure, how to manage, defaults, ...
- Resources
  - RFC 5706 for *guidelines*
  - RFC 6123 for examples
  - Checklists for Ops Directorate reviewers (Appendix in RFC 5706)
  - Early Ops Directorate reviews
  - Specific requests for help
- IANA considerations
  - You cannot make them clear enough
  - Write down exactly what you want to see in the finished registry
  - Don't include other stuff in the IANA section
  - Use tags not values in the text
    - Disambiguate the tags (e.g., TBD1, TBD2...)
- Resources
  - RFC 5226
  - RFC 3692 (for Experimental code points)
  - Existing registries at <http://www.iana.org/protocols>



# Pet peeves

- Have the right reviews been done?
  - Other WGs working on similar topics, same protocols, etc.
  - Experts (MIB, Security, Management)
- Consistent use of RFC 2119 language
- Are experiments properly described?
- Backwards and forwards compatibility must be described or explicitly ruled out
- Resources
  - Checklists from Routing ADs
    - <http://trac.tools.ietf.org/area/rwg/trac/wiki/WikiStart>

# Good English is helpful

- I know this is not easy
  - Don't ask me to write my I-Ds in Swedish
- But you want your I-D to be as clear and easy to read as possible
- The RFC Editor *will* fix things, but...
  - Every change risks introducing a technical error
  - The RFC Editor can only guess at your real meaning
  - Example: draft-ietf-mpis-mldp-hsmp just been edited
    - RFC Editor introduced errors fixing the English
    - Authors assumed the change was just a fix in English and didn't spot the change in meaning
- Resources
  - Other people in the WG
    - This is a real chance for newcomers to become involved
  - RFC Editor workshops on I-D writing
    - <https://www.ietf.org/edu/documents/tutorial76.pdf>
    - <http://trac.tools.ietf.org/group/iesg/trac/wiki/DocumentLanguageEditing>
  - idspell <http://tools.ietf.org/tools/idspell/webservice>

# Did you say what you meant to say?

- A knowledgeable reviewer (or an author) will parse confusing text sympathetically
  - Could a newcomer read and implement from your spec?
- Issues
  - Sometimes issues of good English
  - Sometimes writing style
    - Avoid complex sentences that save words
    - Use more words in simple sentences
    - Use bullet points
- Resources
  - MPLS Review Team
  - Working group last call
  - WG chairs
  - WG shepherds

# Why have you written this draft?

- This may be the hardest thing to say
  - If you don't know then how will the reader?
- Try to state the problem you are solving
  - Clearly, in a few words, with a picture
  - Say why you are not using existing mechanisms
    - Preferably without being rude or disparaging
- Problem statement != use cases
  - Well sometimes it is, but...
  - Do not just collect a set of sections showing “You could also use my great idea for all these things”
    - That is not why you wrote the draft, it is why someone else might like your idea

# Does the solution work?

- It's kind of key, isn't it?
- Getting the main path right is relatively easy
  - While we do not engineer for edge-cases, they have to be covered
- I should not say “I would not have done it like this”
  - Or at least, if I do, it is not a blocking comment, but just a flag for the WG to think about
  - Please push back if I do this too much!
  - Your answer can be:
    - We discussed it, but we prefer our approach

# Has it been implemented?

- This is not a requirement in the MPLS WG
- But if no implementation, why do you want an RFC?
- Some WGs achieve multiple interoperating implementations before requesting publication
- Implementation gives:
  - Validation of technical content
  - Proof that the solution does work
  - Proof of readability and comprehension
  - Firm indication of support and intent
  - A nice warm feeling for the AD
- Resources
  - RFC 6982

# Summary

- MPLS WG is doing a lot of great work
  - Thanks!
- Sometimes you let yourselves down in the last details
  - Recall that your names will forever be associated with what is published
  - Slow down a bit and get it right
- It does not scale for me to be your back-stop