

GSoD 2020 Project Proposal

Wireshark - Modernize and write Extensive Wireshark User Guide(Official and Wiki documentation)

Abstract

A User documentation is designed to assist end users to use a product or service. Good user documentation is very important because it provides an avenue for users to learn how to use a software, its features, tips, tricks and also resolve common problems encountered when using the software. It also reduces support cost and is part of the corporate identity of the product: a good user documentation is a sign of healthiness of the product, the developer team.

Without a good user documentation, a user may not know how to do the above things effectively and efficiently. User Documentations can play a pivotal role in ensuring a product's success because great communication is and will always be at the heart of any business or product and a great documentation just takes that communication and puts it in a manageable framework that everyone can access for success.

Wireshark is at the top of the game with regards to documentation. The current stable download release is Wireshark is 3.2.4. and Wireshark Developer's Guide is Version 3.3.0 and Wireshark User's Guide Version 3.3.0 but it lacks several features and usage guide in the main documentation on website, wiki pages as well as on GitHub.I want to use this project to improve its user Guide to enable end users to have a seamless experience when getting started in Wireshark.

Current State

Currently, the user documentation is available on the wireshark official website [Linked here](#) . It is obsolete, incomplete, hard to navigate or find information, does not contain several user cases and references in the wireshark wiki and wireshark user guide of the official website.

Structure for the proposed documentation

I created a structure for the wireshark user Documentation. This structure needs to be accepted and modified by Mentors before implementation begins.

Project Goals

1. Restructure the documentation.
2. Complete incomplete references and documentation, various missing use case scenarios, currently missing.
3. Write Quick start guide and user friendly tutorials using QT help.
4. Remove ambiguity between wiki and User guide.
5. Remove official documentation references dependency on wiki page to make it a complete standalone guide for new users.
6. Migrate the user documentation to Gitlab/GitHub using Sphinx and ReadtheDocs.
7. Remove obsolete information.
8. Rewrite the user documentation to make it easier to understand.
9. Set it up for translation using Sphinx Internationalization.
10. Make it community driven so users can report any issues of the documentation and also be able to make changes that will be reviewed and merged by the Wireshark team.

Why is your proposed user documentation an improvement over the current one?

The proposed user documentation will be structured to improve and ensure efficiency, consistency, and peace of mind for any end user. It will contain written guides and its associated images, include instructions and explanation on how to use setup and all the necessary protocols one must know in order to work with wireshark no matter your experience in networking you will be able to understand and work around it.

Analysis

I did my research on how popular organizations work on their documentation and I found two popular platforms **Sphinx** and a documentation hosted on **Read the Docs** . Although it would be better to continue working on the current stack used by wireshark so new documentation to be similar to it like Qt Help.

The Qt Help Framework

The Qt help system includes tools for generating and viewing Qt help files. In addition, it provides classes for accessing help contents programmatically to be able to integrate online help into Qt applications.

The actual help data, meaning the table of contents, index keywords, or HTML documents, is contained in Qt compressed help files. So, one such a help file represents usually one manual or documentation set. Since most products are more

comprehensive and consist of a number of tools, one manual is rarely enough. Instead, more manuals, which should be accessible at the same time, exist.

Sphinx

Sphinx is a robust and mature solution for software documentation. It includes a number of features that writers expect, such as:

- Single source publishing, content reuse through includes, conditional includes based on content type and tags.
- Multiple mature HTML themes that provide a great user experience on mobile and desktop
- Referencing across pages, documents, and projects
- Index and Glossary support
- Internationalization support

It uses reStructuredText as its markup language, and many of its strengths come from the power and straightforwardness of reStructuredText and its ability to translate documentations.

Read the Docs

Read the Docs simplifies software documentation by automating building, versioning, and hosting of your docs for you.

- **Never out of sync**

Whenever you push code to your favorite version control system, whether that is Git, Mercurial, Bazaar, or Subversion, Read the Docs will automatically build your docs so your code and documentation are always up-to-date.

- **Multiple versions**

Read the Docs can host and build multiple versions of your docs so having a 1.0 version of your docs and a 2.0 version of your docs is as easy as having a separate branch or tag in your version control system.

- **Free and open source**

Read the Docs is free and open source and hosts documentation for nearly 100,000 large and small open source projects in almost every human and computer language.

Timeline

Before August 17

- Continue making contributions to the organization.
- Become more familiar with the user documentation and current version of Wireshark.
- Learn new techniques and technical writing skills which will be helpful while implementing the project.

Community bonding (August 17 - September 13, 2020)

- Set up a communication channel and time (due to time difference)
- Refine my goals and set expectations on both sides
- Learn more about the community and product
- Discussing the proposed documentation structure with the mentors and other members of the organization.
- Finalize the proposed features and other modifications that need to be made to the existing documentation.

Documentation development (September 14, 2020)

Doc development officially begins!

Week 1 (14 September - 20 September)

This week, I will focus on completing incomplete references and missing documentation on the Wireshark official documentation.

Week 2 (21 September - 27 September)

This week, I will remove official documentation references dependency on the wiki page to make it a complete standalone guide for new users.

Week 3 (28 September - 4 October)

Discover and remove ambiguity between wiki and official documentation

Week 4 (5 October - 11 October)

Review all user complaints in the GUI for increasing clarity and consistency

Week 5 (12 October - 18 October)

Write documentation for common use cases, such as decrypting HTTPS traffic with Wireshark and Extracting files from HTTP traffic, discuss and developing more such use cases

Week 6 (19 October - 25 October)

Starting working with “Quickstart” guide for new Wireshark users to help them launch Wireshark and understand basic packet details as briefly and effectively as possible.

Week 7 (26 October - 1 November)

Continue working in the “Quickstart guide” and add tutorials and demo wherever needed and necessary via QThelp

Week 8 (2 November- 8 November)

This week, I will focus on setting up a Gitlab/GitHub repository which I will use to push and pull all the changes I have made, then start working on the Overview section of the documentation.

- If a Gitlab/GitHub repository for the documentation has already been created by the wireshark community before the beginning of the documentation period, I will fork the repository so that I can be making changes to my branch directly before creating a pull request to the main repository to be reviewed and merged by the mentor. If not, I will create a repository for the documentation on Gitlab (of course after taking permission from the mentor).
- Add a README file to Gitlab/GitHub to help future contributors understand how to contribute to Wireshark user documentation.
- Connect the repository to **Read the Docs** documentation hosting site.
- Commit the newly made changes to my branch and create a pull request for review.

Week 9 (9 November - 15 November)

Proofread and review the existing guides for clarity and accuracy. And complete and reflect on the changes i made till now

Week 10 (16 November - 22 November)

This week, I will continue improving the documentation on **User Guides** and proofread the entire documentation.

- Preferences (Settings)
- Prepare the documentation for translation.
- Make the documentation more community friendly so more and more enthusiasts can start contributing.

Week 11 (23 November - 30 November)

This week I will put all my attention on proofreading the documentation for typos and errors.

Next - November 30 - December 5, 2020 at 18:00 UTC

Will work on the final project reports, also known as final work products

Why this project?

I have always had a conviction that writing code, solving problems and building software reaches its full potential when you also have the ability to enlighten others about it through writing. I first listened about Wireshark from one of my professor in the computer network lab. I personally really like the SharkFest Mission and SharkFest GOALS and how it can help the day to day working of a network engineer.

Why am I the right person for this project?

I believe myself to be the right person for this project because

- I have past experience in working on several research papers and documentation of personal projects and I can use any version control system, so carrying out commands on GitHub will not be a problem. Moreover, what drives me is working on projects that create value to people.
- I am a developer myself and i know what exactly a developer needs when working around the project, and what amount of trouble they have to go through if there is no proper documentation.
- I believe that If you want someone to do something in the most efficient way possible, you document it. By documenting your processes, you ensure efficiency, consistency, and peace of mind for anyone involved.
- I know the core concepts of networking and protocols used and at every layer of the model. I also wish to work as a developer and contribute to the wireshark community at one point in life.

Personal information

- **Name:** Rishabh Arya.

- **Display Name:** Rine
- **Email address:** rishabharya2799@gmail.com
- **Location:** Dehradun,India(GMT+5:30)
- **GitHub:** <https://github.com/RishabhArya>
- **About me:** Undergraduate student currently pursuing Computer Science with Spec. in Open Source and Open Standard from UPES,India.

Information about your technical writing experience

- **Technical writing experience:**
 1. Technical Research paper - Currently writing Research paper on Task Consolidation, VM Migration and SLA Violation, on CloudSim(Open Source) using the materialistic algorithm, which I worked on and developed during my last semester in university.
 2. Medium Articles
 3. Personal Projects
- **Link to my resume/CV:**
 1. LinkedIn - <https://www.linkedin.com/in/rishabh-arya-b64634165/>
 2. Personal website - <http://www.rishabharya.me/#>
- **Link of my work samples:**
 1. Medium Blog - <https://medium.com/@rishabharya2799>
 2. Personal Projects - <https://github.com/RishabhArya>
- **Additional information:** I am a programmer first and I the know the value of good documentation over the years I have worked on various projects and on a good number of technologies from C to Python, SQL to NoSQL databases, Android to Web technologies with good command in Data Structures and Algorithms. I love open source, community and I am fascinated with working with good developers.I