ï»¿

Working Group GitHub Administration

Abstract

   The use of GitHub in IETF working group processes is increasing.
   This document describes uses and conventions for working groups that
   are considering starting to use GitHub. It does not mandate any
   processes and does not require changes to the processes used by
   current and future working groups not using GitHub.

Table of Contents

1. Introduction

Many IETF working groups and participants make use of GitHub in different ways as part of their work on IETF documents. Some others are interested in having their working groups use GitHub to facilitate the development of working group documents, but they are unfamiliar with how to get started or unclear about which conventions to follow. Some other working groups use or plan to use other code-repository services such as GitLab and Bitbucket, which have different properties than GitHub.

This document specifies a set of administrative processes and conventions for IETF working groups to use if they choose as a working group to use GitHub to facilitate their work. The specifications in this document are not directed at working groups or individuals that are already using GitHub to do IETF work. Practices vary among existing working groups, and some of them are not consistent with the conventions proposed here: that is fine. The goal of the specifications in this document is not to require uniformity in current practice, but to help working groups get started using GitHub in a reviewed and validated way, if desired.

2. Administrative Process and Conventions

   This section specifies an administrative process and conventions to support the creation and management of GitHub organizations for working groups and single-document repositories in a uniform way. The steps may be done manually by the IETF Secretariat, or they may be automated. See <https://github.com/richsalz/ietf-gh-scripts> and <https://github.com/martinthomson/i-d-template> for working examples of automation that is in use in some working groups.

   In this document the question of whether processes should be manual or automated is deliberately left unspecified, since these are implementation details that the IETF Secretariat and Tools Team will address.

   Most of the conventions below are drawn from [RFC8874].

2.1. Creation of GitHub Organizations

   This document specifies that there be a facility in the IETF Datatracker (<https://datatracker.ietf.org/>) interface to allow an area director (AD) or working group chair to request the creation of a GitHub organization for a particular working group. Ideally, this facility would appear both as part of the working group chartering UI and the working group page UI.

   When an area director or working group chair makes a request to create a GitHub organization, the following process would be initiated:

   1. Create a GitHub organization for the working group.

   2. Name the organization in the format ietf-wg-<wgname>...

   3. Initialize the organization by designating the IETF Secretariat and the area directors in the working group's area as owners. If the responsible AD for the working group is from another area, that AD will be an owner as well.

   4. Initialize the organization with a team that has administrator access. This team will consist of the working group chairs and working group secretary, if one exists.

   After the organization is created, the URL for the organization would be added to the working group's page in the Datatracker.

   Steps 3 and 4 above imply that the GitHub identities of the organization owners and administrators are known. Recording GitHub identities in the Datatracker (see <https://trac.tools.ietf.org/tools/ietfdb/ticket/2548>) would

facilitate this. The person requesting the organization would need
to be notified if the GitHub identities of any of the people meant to
be owners or administrators were not available.

2.2. Migration of an Existing Organization

If a working group already has an organization, it would be useful to
be able to make it have the same management as one would get by going
through the steps in Section 2.1. That is, it would be good to be
able to run Steps 3 and 4 from Section 2.1 so that the rest of the
activities in this section, such as personnel changes, work the same
way as for organizations that were created as specified herein.

2.3. Personnel Changes

When there are personnel changes in the area or the working group,
those changes would be reflected in the GitHub organization. There
should be an ability in the Datatracker to specify that personnel
changes have occurred.

2.4. Working Group Closing

When a working group is closed, the team with administrative access
would be removed, and the owner list would be returned to the
Secretariat and current ADs at the time of closing. The organization
summary and the repositories within the organization would be updated
to indicate that they are no longer under development. Later, the
owner list could become just the Secretariat, or it might include
others chosen by the Secretariat or the IESG.

2.5. Creation of Document Repository

There are many different scenarios and configurations where it might
be useful to have automation or established administrative
conventions for repositories within WG organizations, such as:

* Creating a new repository for an individual draft (at the
   discretion of the WG chair);

* Creating a new repository for an already adopted working group
   draft;

* Migrating an existing document repository into the WG
   organization; and

* Creating a new repository that contains multiple drafts.

As an incremental step, this document specifies that there be a
facility in the Datatracker interface to allow an administrator of an
ietf-wg-<wgname> organization to request the creation of a new
repository within that organization for a single document. The
document authors would be identified as collaborators. The
repository name would be the draft name. Ideally, the repository
would be configured with a skeleton draft file, default CONTRIBUTING,
LICENSE, and README files, and continuous integration support, in the
vein of <https://github.com/martinthomson/i-d-template>. Performing
this step would automatically inform the IETF Secretariat that this
repository should be backed up as described in Section 3.2.

2.6. Listing Related Repositories

The IETF Datatracker should allow users to add links to repositories
(for GitHub and other repository services) on working group,
document, and user pages. At the time of this writing, this feature
was under development.

3. Working Group Process

[RFC8874] contains discussion of the different possible ways that a
working group can use GitHub and the large number of decisions
associated with doing so. This section specifies a basic set of

administrative policies for working groups to follow and the
administrative support needed to carry out those policies.

3.1. Contributions

At a minimum, every repository created in a working group
organization needs to incorporate into its CONTRIBUTING file the
boilerplate text at <https://trustee.ietf.org/license-for-open-
source-repositories.html> from the IETF license file for open-source
repositories. The CONTRIBUTING file can contain other information as
well (see <https://github.com/ietf/repo-files/tree/master/
contributing-samples> for examples).

It would be useful if the user data in the Datatracker could list (at
a minimum) the GitHub account of the user so that their contributions
could be tracked more easily.

Some working groups choose to have more than one draft in a
repository, particularly for drafts that are tightly linked with
significant cross-references. In such a case, the README for the
repository needs to say so clearly, so that a participant understands
that changes might be made to multiple drafts at once.

3.2. Backing Up and Archiving GitHub Content

IETF working group mailing lists are automatically backed up by the
IETF Secretariat, and the archives are publicly available. All
official interactions in a WG must be archived.

Working group GitHub content also needs to be backed up and publicly
archived. This document specifies using the Git protocol
[git-protocol] itself for both of these tasks.

Every IETF working group repository on GitHub will have a mirror
repository of the same name on a server maintained by the IETF
Secretariat. Every hour, a service will use the "git fetch" command
on every GitHub repository that is being tracked. The mirror
repository will allow anyone to read the repository.

Note that this system will not back up GitHub issues or pull
requests. These should be backed up as well; the GitHub API allows
for this. The IETF Secretariat should back up those at the same time
as it is backing up the GitHub repositories.

The steps in Section 2.5 inform the IETF Secretariat which
repositories should be backed up. Working group chairs and area
directors should also be able to request that the IETF Secretariat
back up additional repositories that are related to IETF working
groups.

4. Security Considerations

An attacker who can change the contents of Internet-Drafts,
particularly late in a working group's process, can possibly cause
unnoticed changes in protocols that are eventually adopted.

There is a risk of data loss due to centralization of data in one
service. This is recognized and mitigated by the plan described in
Section 3.2.

5. IANA Considerations

This document has no IANA actions.

6. Informative References

[git-protocol]
          Chacon, S. and B. Straub, "Git on the Server - The
          Protocols", in Pro Git, 2014, <https://git-
          scm.com/book/en/v2/Git-on-the-Server-The-Protocols#The-
          Git-Protocol>.

   [RFC8874] Thomson, M. and B. Stark, "Working Group GitHub Usage
              Guidance", RFC 8874, DOI 10.17487/RFC8874, August 2020,
              <https://www.rfc-editor.org/info/rfc8874>.

Authors' Addresses

   Alissa Cooper
   Cisco

   Email: alcoop@cisco.com

   Paul Hoffman
   ICANN

   Email: paul.hoffman@icann.org