Getting started with GIT for lunchinators

Philip Dixon

2023-09-06

### Getting started with Git and GitHub

#### Disclaimer: This document covers a very small amount of what Git and GitHub can do

Git is a very powerful tool for collaborative work of any sort. It provides a collaborative workspace, version control and other features. I will post Nick Lyon’s (NCEAS LTER data network) presentation on GIT on a Lunchinators project. You can access that once you have GIT and R connected.

#### Our focus is using GitHub through RStudio to access code and presentations in pre-existing LunchinatoRs projects.

GitHub is a web front end to Git. We will use GitHub or RStudio to drive Git.

Side Note: There are lots of web resources on using Git and GitHub. Many ask you to open a command shell and issue git commands (things that start with git followed by –something). Those are driving Git directly. You can ignore any of those documents (unless you really want to learn about this). We will drive Git using RStudio and R packages.

RStudio provides easy ways to link to Git. This may be possible in vanilla R, but I don’t know how to do that. This is one of various reasons to use RStudio if at all possible.

#### Download and install Git

Instructions depend on your operating system (MacOS, Windows, or a flavor of UNIX) and whether you have an ISU-owned computer (usually managed by ISU admin) or a personal one (you’re the administrator).

ISU-adminstered: Start Software Center (Windows) or SelfServe (MacOS), search for Git, download and install the latest version of Git. On my Windows machine, I installed Git for Windows.

Personal: You can (I believe) use Software Center/SelfServe. Or, go to <https://git-scm.com/downloads>, download and install Git.

#### Create a GitHub account

Go to <https://github.com> and click sign up in the top right. You’ll need to provide an e-mail account. I don’t believe there is any ISU site license for GitHub, so you don’t need to use your ISU e-mail if you don’t want to. Follow the rest of the prompts to create an account.

Remember or make a copy of your password. GitHub demands super-secure passwords. I use the password generated by GitHub but make a copy of it and save it in my browser.

#### Tell RStudio where to find git

Locate where git.exe is stored on your computer. Again, MacOS and Windows are different. I searched for git.ext in the Program Files folder on my Windows machine. On my machine, it was found in c:/Program Files/Git/bin/git.exe It may be different in a Windows 11 machine and almost certainly different on a Mac.

Start RStudio, look at the menu at the top of the RStudio window and click Tools / Global Options. One of the options is Git/SVN. Click that. Look for the line labelled Git executable. It needs to be the path and filename to get to git, e.g., c:/Program Files/Git/bin/git.exe. If not, click browse, navigate to the folder, click git.exe and open. You should now see the path and filename.

#### Install and activate the gitcreds and usethis R packages

In the console window, install.packages(c(‘gitcreds’, ‘usethis’)). Both packages should install without issues.

usethis provides lots of functions to work with GIT. We only need one, create\_github\_token().

gitcreds provides a way to set your credentials for GIT

Activate both libraries:

library(usethis)
library(gitcreds)

#### Get a git personal access token (PAT)

This is different from the GitHub password. The GitHub password is used to sign in to GitHub. The Git PAT is used every time you execute any git command (or have RStudio execute a command on your behalf).

The create\_github\_token() function in the usethis library starts a browser window to generate a PAT.

create\_github\_token()

You will need to provide your e-mail address and GIT password.

The Note is an title for this PAT. You can have multiple PATs used for different purposes. My note is “Access to Lunchinators projects”.

GIT tries to be super secure, so PATs expire. The default is 30 days. You get an e-mail warning you that one is about to expire. You need to get a new PAT and register it again (next item). I change the expiration to 90 days. I don’t recommend no expiration, because you’ll probably forget you have one if you don’t use it for a while.

The scope specifies what activities this PAT will be used for. The default choices are sufficient for our purposes.

Scroll to the bottom of the page and click the green “Generate token” button.

#### NOTE: save that PAT, either in a text file or your clipboard. Once the browser window closes, you can’t get that token again.

The two blue squares to the right of the token copy the token to the clipboard. My practice is to paste that token in a text file and save that text file.

If you don’t save the PAT, just generate a new one and remember to save it.

Close the GitHub window.

#### Tell R about the PAT

gitcreds\_set()

Starts a dialog. You provide your PAT. It will be stored so R can validate your identity for git activity that RStudio does on your behalf.

You should now be able to link to Lunchinators projects stored on Git.

#### Link to a Lunchinators project

Instructions for this are in the Accessing a Lunchinators Repository document

#### What if a PAT expires?

You need to:

* get a new token, using either the link in the expiration e-mail or create\_github\_token()
* register the new token using gitcreds\_set()