Setting up: Git/PoshGit on local machine

The only prerequisite for this set up is a Windows machine with PowerShell installed. Using Chocolatey to install all the Git-related components is optional; you can just download each one and install it manually, if required:

* Git for Windows: git-scm.com/download/win
* PoshGit: https://github.com/dahlbyk/posh-git
* git-credential-winstore: https://gitcredentialstore.codeplex.com/

Step 1. Install Chocolatey

Open a command window using Run as administrator (Locate cmd.exe and right-click)

Go to <https://chocolatey.org/>

Copy the command that is currently right there on the home page:

@powershell -NoProfile -ExecutionPolicy unrestricted -Command "iex ((new-object net.webclient).DownloadString('https://chocolatey.org/install.ps1'))" && SET PATH=%PATH%;%ALLUSERSPROFILE%\chocolatey\bin

Paste it into the command window and run it.

Once it's installed close the command window.

Step 2: Install PoshGit

Open a command window using Run as administrator

Issue:

> cinst poshgit

Or

> choco install poshgit

Step 2: Install git-credential-winstore

Open a command window using Run as administrator

Issue:

> cinst git-credential-winstore -y

winstore will store and pass the credentials automatically so we don't to need sign in for every interaction with the from remote repo.

Note: if the remote repo is on VS Online, you will need to login to VSO and enable alternate credentials in order to connect to the remote repo from a command line client. When prompted by the winstore, provide email address + alternate password.

Step 3: Set execution policy for PowerShell

If you've never used PS, need to set execution policy for PowerShell to allow it to run trusted, downloaded scripts.

Open PowerShell as an administrator (right click, run as administrator)

Issue:

> Set-executionpolicy remotesigned

Step 4: Set the startup directory for PowerShell (optional)

Locate the PowerShell startup script ($profile) called Microsoft.PowerShell\_profile.ps1 (often located in C:\Users\username\Documents\WindowsPowerShell) Note that you may need to create this file.

Open the file and use set-location to establish preferred startup directory:

Set-Location "C:\users\username"

Save your change.

Step 5: Set Git path variable

We need to set git path variable so we can execute git commands from command line – choclately should do this for you but in my case it got it wrong.

Locate git.exe (usually C:\Program Files (x86)\Git\bin)

Then:

Start | Computer – Right click – Properties | Advanced system settings | Environment variables… – highlight an entry in the System variables section and type "p" to find the path variable, click Edit.

Hit End to get to end of current path value and, leaving no gaps, paste C:\Program Files (x86)\Git\bin immediately after the final semi-colon.

Hit OK.

Step 6: Configure Git

At minimum, need to configure the display name and email for the user:

**> git config --global user.name "Your Name"**

**> git config --global user.email "name@youremail.com"**

Optionally, change editor from vi…to whatever is your preferred editor

> **git config --global core.editor Notepadplusplus.exe**

Step 7: Start using Git

The first time you interact with a remote repo, winstore (if installed) will prompt you to enter credentials into a sign in pop-up window. If you didn't install winstore, or for some reason it isn't working, the credential prompt will appear in the command line (and you'll have to sign in every time you need to access the remote repo).