Known issues

* On Windows 10 before 1703, or when Developer Mode is turned off, special permissions are required when cloning repositories with symbolic links, therefore support for symbolic links is disabled by default. Use git clone -c core.symlinks=true <URL> to enable it, see details [here](https://github.com/git-for-windows/git/wiki/Symbolic-Links).
* If configured to use Plink, you will have to connect with [putty](http://www.chiark.greenend.org.uk/~sgtatham/putty/) first and accept the host key.
* Some console programs, most notably non-MSYS2 Python, PHP, Node and OpenSSL, interact correctly with MinTTY only when called through winpty (e.g. the Python console needs to be started as winpty python instead of just python).
* [cURL](http://curl.haxx.se) uses $HOME/\_netrc instead of $HOME/.netrc.
* If you specify command-line options starting with a slash, POSIX-to-Windows path conversion will kick in converting e.g. "/usr/bin/bash.exe" to "C:\Program Files\Git\usr\bin\bash.exe". When that is not desired -- e.g. "--upload-pack=/opt/git/bin/git-upload-pack" or "-L/regex/" -- you need to set the environment variable MSYS\_NO\_PATHCONV temporarily, like so:

MSYS\_NO\_PATHCONV=1 git blame -L/pathconv/ msys2\_path\_conv.cc

Alternatively, you can double the first slash to avoid POSIX-to-Windows path conversion, e.g. "//usr/bin/bash.exe".

* Windows drives are normally recognised within the POSIX path as /c/path/to/dir/ where /c/ (or appropriate drive letter) is equivalent to the C:\ Windows prefix to the \path\to\dir. If this is not recognised, revert to the C:\path\to\dir Windows style.
* Git for Windows will not allow commits containing DOS-style truncated 8.3-format filenames ending with a tilde and digit, such as mydocu~1.txt. A workaround is to call git config core.protectNTFS false, which is not advised. Instead, add a rule to .gitignore to ignore the file(s), or rename the file(s).
* Many Windows programs (including the Windows Explorer) have problems with directory trees nested so deeply that the absolute path is longer than 260 characters. Therefore, Git for Windows refuses to check out such files by default. You can overrule this default by setting core.longPaths, e.g. git clone -c core.longPaths=true ....
* Some commands are not yet supported on Windows and excluded from the installation.
* As Git for Windows is shipped without Python support, git p4 (which is backed by a Python script) is not supported.
* The Quick Launch icon will only be installed for the user running setup (typically the Administrator). This is a technical restriction and will not change.
* Older versions of the Windows Explorer do *not* calculate Git for Windows' on-disk size correctly, as it is unaware of hard links. Therefore, it might look like Git for Windows takes up 1.5GB when in reality it is about a third of that.

Should you encounter other problems, please first search [the bug tracker](https://github.com/git-for-windows/git/issues) (also look at the closed issues) and [the mailing list](http://groups.google.com/group/git-for-windows), chances are that the problem was reported already. Also make sure that you use an up to date Git for Windows version (or a [current snapshot build](https://wingit.blob.core.windows.net/files/index.html)). If it has not been reported, please follow [our bug reporting guidelines](https://github.com/git-for-windows/git/wiki/Issue-reporting-guidelines) and [report the bug](https://github.com/git-for-windows/git/issues/new).

Licenses

Git is licensed under the GNU General Public License version 2.

Git for Windows also contains Embedded CAcert Root Certificates. For more information please go to <https://www.cacert.org/policy/RootDistributionLicense.php>.

This package contains software from a number of other projects including Bash, zlib, curl, tcl/tk, perl, MSYS2 and a number of libraries and utilities from the GNU project, licensed under the GNU General Public License. Likewise, it contains Perl which is dual licensed under the GNU General Public License and the Artistic License.

Changes since Git for Windows v2.22.0 (June 8th 2019)

New Features

* Comes with [Git v2.23.0](https://github.com/git/git/blob/v2.23.0/Documentation/RelNotes/2.23.0.txt).
* Comes with [patch level 3](https://github.com/git-for-windows/msys2-runtime/commit/e0e7936faa74acea8cde0f89f464402515d1caad) of the MSYS2 runtime (Git for Windows flavor) based on [Cygwin 3.0.7](https://cygwin.com/ml/cygwin-announce/2019-04/msg00030.html).
* Comes with [PCRE2 v10.33](https://pcre.org/changelog.txt).
* Comes with [GNU Privacy Guard v2.2.17](https://lists.gnupg.org/pipermail/gnupg-announce/2019q3/000439.html).
* Comes with [cURL v7.65.3](https://curl.haxx.se/changes.html#7_65_3).
* Comes with [Git LFS v2.8.0](https://github.com/git-lfs/git-lfs/releases/tag/v2.8.0).
* When configuring Git for Windows to use plink, [there is now a checkbox specifically for TortoisePlink](https://github.com/git-for-windows/build-extra/pull/251).
* The FSCache feature [is now used with git checkout and git reset in sparse checkouts](https://github.com/git-for-windows/git/pull/2224).

Bug Fixes

* Git for Windows' MSYS2 runtime was [patched](https://github.com/git-for-windows/msys2-runtime/commit/c10b4185a35f494a2ff4ad2f5828540d93d56bec) to fix a bug where setting the environment variable SHELL to an empty string in a shell script would not only fail to pass that setting to non-MSYS2 processes (such as git.exe) but also completely skip all environment variables that sort after said variable.
* git clean -dfx [no longer follows NTFS junction points (also known as mount points)](https://github.com/git-for-windows/git/pull/2268).
* A [workaround](https://github.com/git-for-windows/git/pull/2253) now allows cloning to certain network drives (e.g. Isilon).
* Fixed [CVE-2019-1211](https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2019-1211) in MinGit/Portable Git by being more careful about validating the Windows-wide config.

Changes since Git for Windows v2.21.0 (February 26th 2019)

New Features

* Comes with [Git v2.22.0](https://github.com/git/git/blob/v2.22.0/Documentation/RelNotes/2.22.0.txt).
* The awk included in Git for Windows [now includes extensions](https://github.com/git-for-windows/build-extra/pull/232) such as inplace.
* The file/product version stored in the installer's .exe file [now matches the version of the included git.exe file's](https://github.com/git-for-windows/build-extra/pull/235).
* Comes with [OpenSSH v8.0p1](https://www.openssh.com/txt/release-8.0).
* Comes with [Git LFS v2.7.2](https://github.com/git-lfs/git-lfs/releases/tag/v2.7.2).
* Comes with MSYS2 runtime (Git for Windows flavor) based on Cygwin v3.x (see release notes for Cygwin [3.0.0](https://cygwin.com/ml/cygwin-announce/2019-02/msg00010.html), [3.0.1](https://cygwin.com/ml/cygwin-announce/2019-02/msg00014.html), [3.0.2](https://cygwin.com/ml/cygwin-announce/2019-03/msg00002.html), [3.0.3](https://cygwin.com/ml/cygwin-announce/2019-03/msg00008.html), [3.0.4](https://cygwin.com/ml/cygwin-announce/2019-03/msg00016.html), [3.0.5](https://cygwin.com/ml/cygwin-announce/2019-03/msg00051.html), [3.0.6](https://cygwin.com/ml/cygwin-announce/2019-04/msg00012.html), and [3.0.7](https://cygwin.com/ml/cygwin-announce/2019-04/msg00030.html)).
* There are now [experimental built-in versions of git add -i and git add -p](https://github.com/git-for-windows/git/pull/2150), i.e. those modes are now a lot faster (in particular at startup). You can opt into using them on the last installer page.
* PortableGit [now comes with a meta credential helper](https://github.com/git-for-windows/git/issues/2116), i.e. a GUI that lets the user choose *which* of the available credential helpers to use. This should help to avoid storing credentials on other people's machines when running portable Git from a thumb drive.
* Comes with [gawk v5.0.0](http://git.savannah.gnu.org/cgit/gawk.git/plain/NEWS?h=gawk-5.0.0).
* Comes with [Git Credential Manager v1.19.0](https://github.com/Microsoft/Git-Credential-Manager-for-Windows/releases/tag/1.19.0).
* Comes with [git-flow v1.12.3](https://github.com/petervanderdoes/gitflow-avh/releases/tag/1.12.3).
* Comes with [OpenSSL v1.1.1c](https://www.openssl.org/news/openssl-1.1.1-notes.html).
* Comes with [GNU Privacy Guard v2.2.16](https://lists.gnupg.org/pipermail/gnupg-announce/2019q2/000438.html), specifically [patched to handle Windows paths](https://github.com/git-for-windows/MSYS2-packages/pull/33).
* Comes with [cURL v7.65.1](https://curl.haxx.se/changes.html#7_65_1).
* Comes with [Heimdal v7.5.0](http://h5l.org/releases.html). -packages/pull/33).

Bug Fixes

* Git for Windows' updater [is now accessible](https://github.com/git-for-windows/build-extra/pull/234), i.e. it can be read by a screen reader.
* git update-git-for-windows (i.e. the auto updater of Git for Windows) now [reports correctly when it failed to access the GitHub API](https://github.com/git-for-windows/build-extra/pull/239).
* Git for Windows' updater [no longer runs into GitHub API rate limits](https://github.com/git-for-windows/build-extra/pull/242) (this used to be quite common in enterprise scenarios, where many users would share one IP as far as GitHub is concerned).
* gitk [no longer fails with "filename too long"](https://github.com/git-for-windows/git/pull/2170) when there are 1,000+ branches/tags.
* A bug which on occasion caused lengthy rebase runs to crash without error message [was fixed](https://github.com/git-for-windows/git/pull/2182).
* Two workarounds from the Git for Windows 1.x era (concerning reading credentials via GUI and fetching via git://) [were considered obsolete](https://github.com/git-for-windows/git/pull/2178).
* git difftool --no-index [can now be run outside of Git worktrees](https://github.com/git-for-windows/git/pull/2175).
* git rebase -i used to get confused when an exec command created new commits and then appended pick lines for them. This [has been fixed](https://github.com/git-for-windows/git/pull/2121).
* During a run of git rebase --rebase-merges, the output of git status [now shows label lines correctly](https://github.com/git-for-windows/git/pull/2185), i.e. with the labels' names instead of the commit hash they point to.
* We [now avoid problems updating the commit graph](https://github.com/git-for-windows/git/pull/2198) when gc.writeCommitGraph=true.

Changes since Git for Windows v2.20.1 (December 15th 2018)

New Features

* Comes with [Git v2.21.0](https://github.com/git/git/blob/v2.21.0/Documentation/RelNotes/2.21.0.txt).
* The custom editor setting in the installer [has been improved substantially](https://github.com/git-for-windows/build-extra/pull/221).
* Comes with [Git Credential Manager v1.18.4.0](https://github.com/Microsoft/Git-Credential-Manager-for-Windows/releases/tag/1.18.4.0).
* Comes with [cURL v7.64.0](https://curl.haxx.se/changes.html#7_64_0).
* Comes with [git-flow v1.12.0](https://github.com/petervanderdoes/gitflow-avh/releases/tag/1.12.0).
* git archive [no longer requires gzip to generate .tgz archives](https://github.com/git-for-windows/git/pull/2077) (this means in particular that it works in MinGit).
* System-wide Sublime Text installations [are now detected](https://github.com/git-for-windows/build-extra/commit/396b283cc6231589b0b034d4ca4b241b25163e9a) and offered on the editor wizard page.
* Comes with [Git LFS v2.7.1](https://github.com/git-lfs/git-lfs/releases/tag/v2.7.1).

Bug Fixes

* The Git CMD deprecation [was further clarified](https://github.com/git-for-windows/build-extra/pull/222) to mention that the *Start Menu item* is deprecated, not using Git from CMD.
* Certain drivers/anti-malware caused git.exe to hang, which [has been fixed](https://github.com/git-for-windows/MINGW-packages/pull/32).
* git stash [now works](https://github.com/git-for-windows/git/issues/2006) after staging files with git add -N.
* A problem with difftool and more than a handful modified files [has been fixed](https://github.com/git-for-windows/git/pull/2026).
* The regression where git-cmd <command> would not execute the command [was fixed](https://github.com/git-for-windows/git/issues/2039).
* Portable Git [can be launched via network paths again](https://github.com/git-for-windows/git/issues/2036).
* FSCache works again [on network drives](https://github.com/git-for-windows/git/issues/2022), in particular [when Windows 8.1 or older](https://github.com/git-for-windows/git/issues/1989) are involved.
* Partially hidden text in the Path options page in the installer [is no longer hidden](https://github.com/git-for-windows/git/issues/2049).
* Fixes [an obscure git svn hang](https://github.com/git-for-windows/git/issues/1993).
* The installer [now configures editors so that the built-in rebase can use them](https://github.com/git-for-windows/git/issues/2011).