CLI   Manual   Release notes

gh auth login

```
gh auth login [flags]
```

Authenticate with a GitHub host.

The default hostname is `github.com` . This can be overridden using the `--hostname` flag.

The default authentication mode is a web-based browser flow. After completion, an authentication token will be stored securely in the system credential store. If a credential store is not found or there is an issue using it gh will fallback to writing the token to a plain text file. See `gh auth status` for its stored location.

Alternatively, use `--with-token` to pass in a token on standard input. The minimum required scopes for the token are: `repo` , `read:org` , and `gist` .

Alternatively, gh will use the authentication token found in environment variables. This method is most suitable for "headless" use of gh such as in automation. See `gh help environment` for more info.

To use gh in GitHub Actions, add `GH_TOKEN: ${{ github.token }}` to `env` .

The git protocol to use for git operations on this host can be set with `--git-protocol` , or during the interactive prompting. Although login is for a single account on a host, setting the git protocol will take effect for all users on the host.

Specifying `ssh` for the git protocol will detect existing SSH keys to upload, prompting to create and upload a new key if one is not found. This can be skipped with `--skip-ssh-key` flag.

Options

-p , --git-protocol <string>
  The protocol to use for git operations on this host: {ssh|https}

-h , --hostname <string>
  The hostname of the GitHub instance to authenticate with

--insecure-storage
  Save authentication credentials in plain text instead of credential store

-s , --scopes <strings>
  Additional authentication scopes to request

--skip-ssh-key
  Skip generate/upload SSH key prompt

-w , --web
  Open a browser to authenticate

--with-token
  Read token from standard input

Examples

```
# Start interactive setup
$ gh auth login

# Authenticate against github.com by reading the token from a file
$ gh auth login --with-token < mytoken.txt

# Authenticate with specific host
$ gh auth login --hostname enterprise.internal
```

list
remove
search
upgrade

gist
clone
create
delete
edit
list
rename
view

gpg-key
add
delete
list

help
environment
exit-codes
formatting
mintty
reference

issue
close
comment
create
delete
develop
edit
list
lock
pin
reopen
status
transfer
unlock
unpin
view

label
clone
create
delete
edit
list

org
list

pr
checkout
checks
close
comment
create
diff
edit
list
lock
merge
ready
reopen
review
status
unlock
update-
branch
view

See also

- gh auth

project
close
copy
create
delete
edit
field-
create
field-
delete
field-list
item-add
item-
archive
item-
create
item-
delete
item-edit
item-list
link
list
mark-
template
unlink
view

release
create
delete-
asset
delete
download
edit
list
upload
view

repo
archive
clone
create
delete
deploy-key
deploy-key
add
deploy-key
delete
deploy-key
list
edit
fork
gitignore
gitignore
list
gitignore
view
license
license
list
license
view
list
rename
set-
default
sync
unarchive
view

ruleset

check
list
view

run
cancel
delete
download
list
rerun
view
watch

search
code
commits
issues
prs
repos

secret
delete
list
set

ssh-key
add
delete
list

status

variable
delete
get
list
set

workflow
disable
enable
list
run
view

GitHub

Product	Platform	Support	Company
Features	Developer API	Help	About
Security	Partners	Community Forum	Blog
Enterprise	GitHub Desktop	Expert Services	Careers
Customer stories	GitHub Mobile	Status	Press
Pricing		Contact GitHub	Shop
Resources			