

Message "Support for password authentication was removed."

Asked 4 years ago Modified 5 months ago Viewed 2.5m times



I got this error on my console when I tried to use `git pull`:



1815

remote: Support for password authentication was removed on August 13, 2021.
remote: Please see <https://docs.github.com/en/get-started/getting-started-with-git/about-remote-repositories#cloning-with-https-urls> for information on currently recommended modes of authentication.
fatal: Authentication failed for ...



It's very weird, because I just followed the documentation and created a token two weeks ago on GitHub. The token expires on Tue, Oct 26, 2021. Why has this been removed today?

[git](#) [github](#) [github-api](#) [git-pull](#) [personal-access-token](#)

Share Improve this question Follow

edited Jul 27, 2023 at 14:06

asked Aug 13, 2021 at 16:49



Colonel Panic

138k 98 420 482



Daemes

17.1k 3 10 16

Sometimes it uses ssh-askpass tool from SSH_ASKPASS env variable and stores the password in wallet. To overcome this either change/remove the values in wallet or `unset SSH_ASKPASS` – Logu Sep 15, 2023 at 17:24

53 Answers

Sorted by: Highest score (default)

1 2 Next



2281

From 2021-08-13, GitHub is no longer accepting account passwords when authenticating Git operations. You need to add a **PAT (Personal Access Token)** instead, and you can follow the below method to add a PAT on your system.



Create Personal Access Token on GitHub



From your GitHub account, go to **Settings** → **Developer Settings** → **Personal Access Token** → **Tokens (classic)** → **Generate New Token** (Give your password) → **Fillup the**

form → click **Generate token** → **Copy the generated Token**, it will be something like
ghp_sFhFsSHhTzMDreGRLjmks4Tzuzgthdvsrta

Now follow the below method based on your machine:

For Windows OS

Go to **Credential Manager** from **Control Panel** → **Windows Credentials** → find **git:https://github.com** → **Edit** → On Password replace with your **GitHub Personal Access Token** → You are Done

If you don't find **git:https://github.com** → Click on **Add a generic credential** → Internet address will be **git:https://github.com** and you need to type in your username and password will be your **GitHub Personal Access Token** → Click Ok and you are done

For macOS

Click on the Spotlight icon (magnifying glass) on the right side of the menu bar. Type **Keychain access** then press the Enter key to launch the app → In Keychain Access, search for **github.com** → Find the **internet password** entry for **github.com** → Edit or delete the entry accordingly → You are done

For a Linux-based OS

For Linux, you need to configure the local GIT client with a username and email address,

```
$ git config --global user.name "your_github_username"
$ git config --global user.email "your_github_email"
$ git config -l
```

Once GIT is configured, we can begin using it to access GitHub. Example:

```
$ git clone https://github.com/YOUR-USERNAME/YOUR-REPOSITORY
> Cloning into 'YOUR-REPOSITORY'...
Username: <type your username>
Password: <type your password or personal access token (GitHub)
```

Now cache the given record in your computer to remembers the token:

```
$ git config --global credential.helper cache
```

If needed, anytime you can delete the cache record by:

```
$ git config --global --unset credential.helper
$ git config --system --unset credential.helper
```

Now try to pull with `-v` to verify

```
$ git pull -v
```

Linux/Debian (Clone as follows):

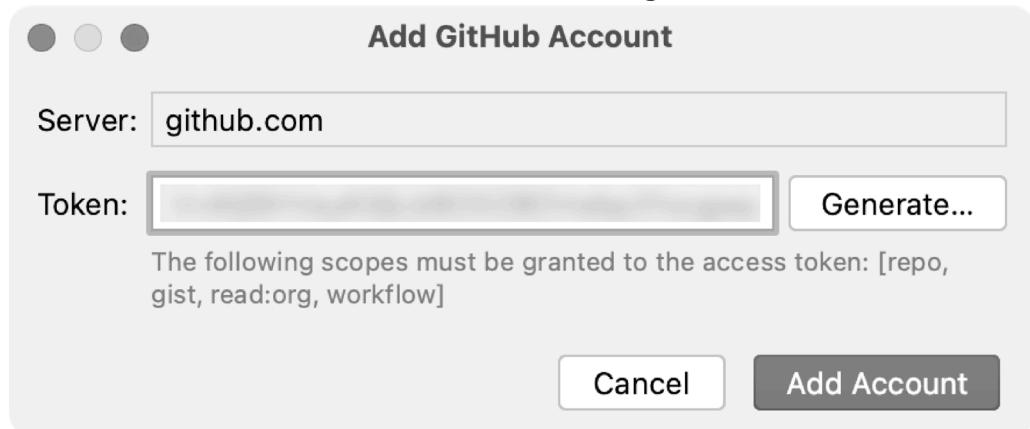
```
git clone https://<tokenhere>@github.com/<user>/<repo>.git
```

For JetBrains IDEs

You can consult the relevant help page for your IDE of choice ([IntelliJ](#), [PhpStorm](#), [WebStorm](#), [GoLand](#), [PyCharm](#), [Rider](#), [RustRover](#)) for more information on logging in. Below is a short snippet of the [IntelliJ help page](#):

Register an existing account with a token

1. Press `⌘Cmd 0 / ⌘Ctrl Alt 0` (your actual key-binds may differ), to open settings and then select **Version Control | GitHub**.
2. Click the Add button.
3. Select **Log In with Token**.
4. The token can then be inserted into the following text-field:



5. Click Add Account

Share Improve this answer Follow

edited Sep 20, 2024 at 14:42

answered Aug 14, 2021 at 6:58



Edric

27.1k

13

87

96



STA
35.3k

9

49

62

37 Be sure to save the token on some local file or in some cloud.Upvoted. – Goran_Ilic_Ilke Aug 14, 2021 at 9:37

26 Given that the OP states they already created a token two weeks ago, this doesn't address the actual problem. – 9769953 Aug 14, 2021 at 11:55

12 Ummmm, it isn't asking for a token when I clone my repo, how should I proceed then? – Prhyme Aug 14, 2021 at 15:16

12 Same experience as @Prhyme. I am not able to get git to prompt me for username and password, even trying to clone private repos, push to repos, etc. When I go to push to a repo, it continues to give the "Support for password authentication was removed" message, without prompting for anything. – SeanRtS Aug 14, 2021 at 19:08

86 For mac first time, when you are prompted for a password when you pushing the code to your repo, enter the access token, It will add the token in the keychain and it won't prompt again. – Dhiraj Himani Aug 23, 2021 at 5:38



If you're using macOS, just simply follow these steps:

626

1. Go to [this link](#): (Profile -> settings -> developers setting -> personal access tokens). (don't go to repository setting; it's your profile setting)
2. Generate a new token and copy-paste it somewhere safely.
3. Now search for an app in your Mac, named `Keychain Access`.
4. Search for `github.com` (if there are multiple GitHub logins then choose `Kind: Internet password`), double-click it.
5. Click on `Show password`, then enter your Mac's password and hit `Enter`.
6. Password should be visible by now. Now, just paste the token you generated in step 2 and click `Save changes`.

And that's it. Enjoy!



If you're using Windows:

1. Follow steps 1 and 2 as above.
2. Search for an application in your Windows OS, named *Credential Manager* → then *Windows Credentials*.

3. Search for `github.com` and *edit* the password with the `token` you have generated on GitHub. Now enjoy!

Developer's hack (shortcode):

```
git remote set-url origin
https://<githubtoken>@github.com/<username>/<repositoryname>.git
```

While cloning:

```
git clone https://<username>:<githubtoken>@github.com/<username>/<repositoryname>.git
```

It will work on every OS (Mac, Windows, or Linux).

Cons: You have to remember or should need to do to each repository in your local. So I'll prefer everyone to use above mentioned steps.

NOTE:

For those who don't have this entry: it could be made. one way to do it is- to clone a project. then it will ask for your username and password. instead of password give it the token and then the entry would be made.

Share Improve this answer Follow

edited Aug 14, 2023 at 11:50

answered Aug 14, 2021 at 4:46



Mehdi Charife

1,063 3 11 29



kartik tyagi

7,210 2 17 34

- 3 I found that on Windows, it was necessary to *delete* the existing credential in the credential manager. Then, the next time I tried to do a `git push`, it prompted for the Personal Access Token. Simply resetting the existing credential did not work. – J. Peterson Aug 16, 2021 at 7:48
- 1 The developer hack worked good for me (I go between Github, Gitlab, and multiple accounts on each site) but it seems to me that this is a *major security downgrade* from just typing in a password, each one of these repos can push directly to Github without asking for a password at all; how is this safer? – jrh Aug 26, 2021 at 15:49
- 47 For those who does not have this entry: it should be made . one way to do it is to clone a project. then it will ask for your username and password. instead of password give it the token and then the entry would be made. – Learner Sep 15, 2021 at 17:18
- 2 That didn't work for me. [rodmclaughlin.com/...](http://rodmclaughlin.com/) – rodmclaughlin Sep 21, 2021 at 11:02
- 2 using the generated token as github password on terminal screen. – Ramazan Sağır Nov 21, 2021 at 9:01

Use *My Account → Settings → Developer settings → Personal access tokens → Generate new token*.



```
git remote set-url origin https://<token>@github.com/<username>/<repo>
```

410

[Share](#) [Improve this answer](#) [Follow](#)

edited Sep 5, 2021 at 9:11



Peter Mortensen

31.6k 22 110 134

answered Aug 14, 2021 at 9:05



Balaji

11.2k 5 61 53



-
- 16 Note that this seems to require running this command in each repo for that repo to be reset with the new credentials. Hopefully there is a global solution. – SeanRtS Aug 14, 2021 at 19:08
- 19 This one perfectly worked for me. The above one with maximum vote didn't work. – Kapilfreeman Aug 14, 2021 at 20:55
- 5 Thanks! Using token as password not work, but adding this work awesome – Syorito Hatsuki Aug 16, 2021 at 4:28
- 14 This is very insecure, not only does it allow anyone on the local system with read access to the local repo control over that repo, `git remote -v` simply discloses the key giving the attacker account wide access to your GitHub account. – Joshua Kolden Aug 17, 2021 at 20:48
- 8 @Bālājī OS key managers and browsers take great pains to secure credentials. Normal git workflow do not expose passwords or ssh private keys to other users with read access to a repo. The above advice does. It allows anyone with read access to a copy of a local repo, including for example a shared work or school filesystem, to see the user's clear text Personal Access Token. This extends to archived copies as well, such as source tarballs that retain the `.git` directory. – Joshua Kolden Aug 18, 2021 at 4:27
-



For Linux these simple steps can solve your problem

243

1. If your Git password is cached in `credential.helper`, then unset it:

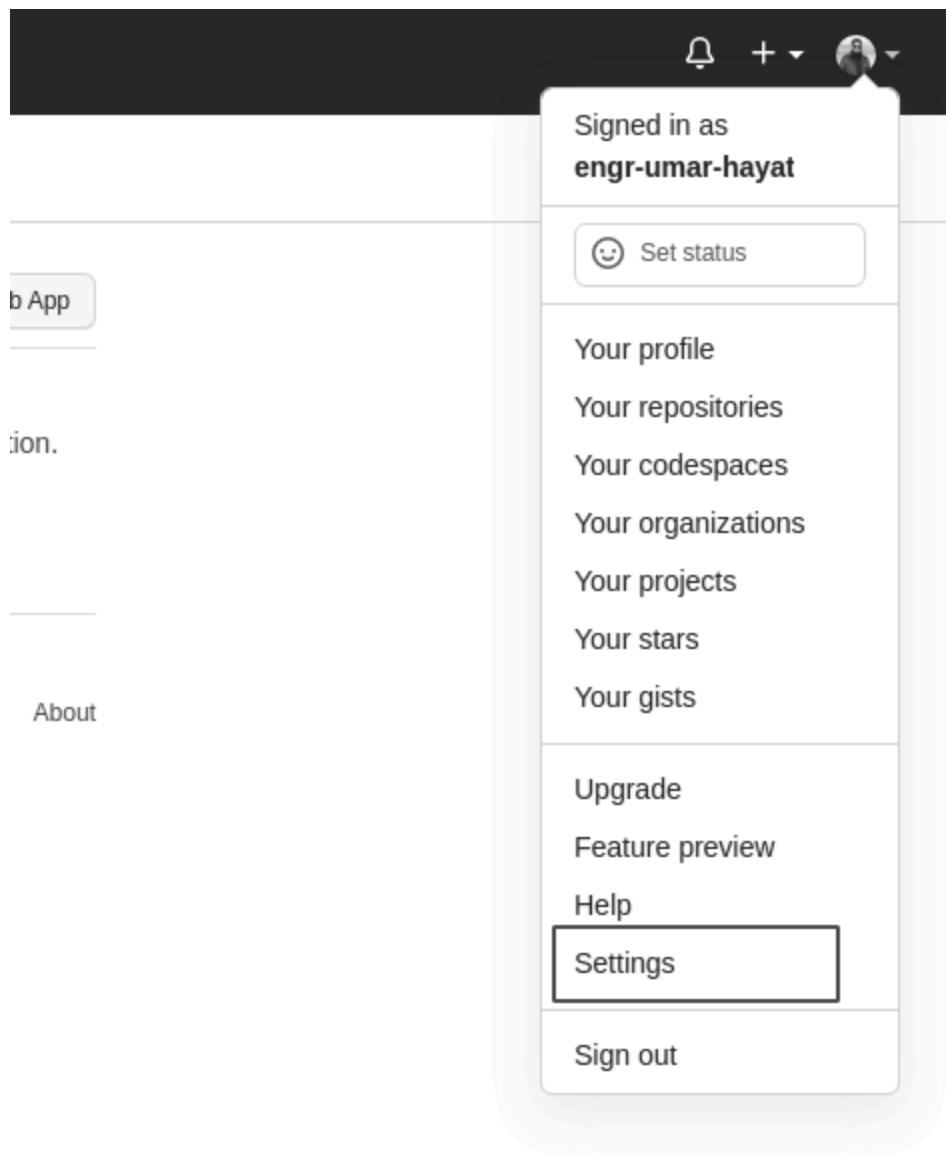
```
git config --local --unset credential.helper
```



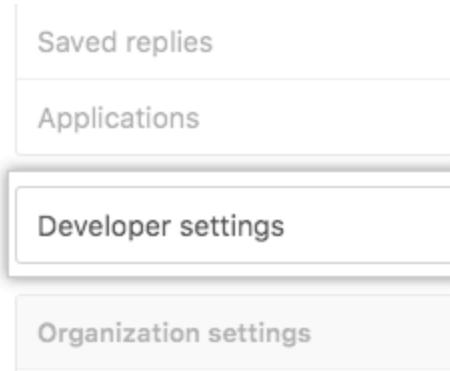
Or, If you have set your credentials globally, then:

```
git config --global --unset credential.helper
```

2. Now go to your GitHub Account settings



3. Click *Developer Settings*



4. Select *Personal Access*

[Settings / Developer settings](#)

[GitHub Apps](#)

[OAuth Apps](#)

[Personal access tokens](#)

GitHub Apps

[New GitHub App](#)

Want to build something that integrates with and extends GitHub? Register a new GitHub App to get started developing on the GitHub API. You can also read more about building GitHub Apps in our developer documentation.

© 2021 GitHub, Inc.

[Terms](#)

[Privacy](#)

[Security](#)

[Status](#)

[Docs](#)



[Contact GitHub](#)

[Pricing](#)

[API](#)

[Training](#)

[Blog](#)

[About](#)

5. Generate a token with the given permissions, e.g.,

<input checked="" type="checkbox"/> repo	Full control of private repositories
<input checked="" type="checkbox"/> repo:status	Access commit status
<input checked="" type="checkbox"/> repo_deployment	Access deployment status
<input checked="" type="checkbox"/> public_repo	Access public repositories
<input checked="" type="checkbox"/> repo:invite	Access repository invitations
<input checked="" type="checkbox"/> admin:org	Full control of orgs and teams
<input checked="" type="checkbox"/> write:org	Read and write org and team membership
<input checked="" type="checkbox"/> read:org	Read org and team membership
<input checked="" type="checkbox"/> admin:public_key	Full control of user public keys
<input checked="" type="checkbox"/> write:public_key	Write user public keys
<input checked="" type="checkbox"/> read:public_key	Read user public keys
<input checked="" type="checkbox"/> admin:repo_hook	Full control of repository hooks
<input checked="" type="checkbox"/> write:repo_hook	Write repository hooks
<input checked="" type="checkbox"/> read:repo_hook	Read repository hooks
<input checked="" type="checkbox"/> admin:org_hook	Full control of organization hooks
<input type="checkbox"/> gist	Create gists
<input type="checkbox"/> notifications	Access notifications
<input checked="" type="checkbox"/> user	Update all user data
<input checked="" type="checkbox"/> read:user	Read all user profile data
<input checked="" type="checkbox"/> user:email	Access user email addresses (read-only)
<input checked="" type="checkbox"/> user:follow	Follow and unfollow users
<input type="checkbox"/> delete_repo	Delete repositories
<input type="checkbox"/> write:discussion	Read and write team discussions
<input type="checkbox"/> read:discussion	Read team discussions
<input checked="" type="checkbox"/> admin:gpg_key	Full control of user gpg keys (Developer Preview)
<input checked="" type="checkbox"/> write:gpg_key	Write user gpg keys
<input checked="" type="checkbox"/> read:gpg_key	Read user gpg keys

6. Now `git pull` inside your Git repository

7. Provide a username and the generated token as a password

```
Username for 'https://github.com': engr-umar-hayat
Password for 'https://enr-umar-hayat@github.com': GENERATED TOKEN HERE
remote: Enumerating objects: 1814, done.
remote: Counting objects: 100% (949/949), done.
remote: Compressing objects: 100% (200/200), done.
```

That is a straightforward solution step by step.

PS: If you are annoyed by Git asking for username/token again and again, follow these three simple steps

- Run `nano ~/.git-credentials`. Remove the GitHub line and save it.
- `git config --global credential.helper store`
Risky as physically the token is saved in file `~/.git-credentials`
- Run `git pull` and provide the username and password only once

It will not ask for the username and access token again and again now!

Share Improve this answer Follow

edited Sep 5, 2021 at 9:14

answered Aug 14, 2021 at 8:51



Peter Mortensen

31.6k 22 110 134



Umar Hayat

5,061 2 18 28

4 Those are a *lot* of permissions. What are you using that token for? – Bergi Aug 14, 2021 at 13:15

1 It all depends on the use case you want to use that token for. So select those permissions according to your use case. – Umar Hayat Aug 14, 2021 at 15:37

1 It only worked when I followed this. – ScottyBlades Oct 26, 2021 at 21:53

Is there a way to save it to a custom location? Perhaps it can help reduce the risk? – Zack Light Jan 18, 2022 at 0:29

1 Do NOT do that with permissions. – Leandro Bardelli May 6, 2022 at 19:32

For Windows OS,

211

GitHub has made changes in password authentication. If you are trying to access Git by username and password then it does not allow you. So use a **personal access token** instead of a password to access Git everywhere.

Here are the steps to generate **personal access tokens**.

Click here for Token - <https://github.com/settings/tokens>

Step 1 - Open GitHub and log in with your credentials.

Step 2 - Click on the **Setting** menu.

Step 3 - From the **Setting** menu click on **Developer Settings**

Step 4 - From the **Developer Settings** menu, click on **Personal access token**

Step 5 - From the **Personal access token**, click on the **Generate new Token** button.

Step 6 - Now fill up required details like **Note**, **Expiration**, **Select scopes**. And then click on the **Generate Token** button.

Step 7 - After that, a new token has been generated. Copy that generated token and use this token to access Git with username and token.

If you are using the Windows operating system then please follow the below step.

Open **Control Panel** → **User Accounts** → **Manage your credentials** → **Windows Credentials**.

It will show all generic credentials. Find your GitHub URL and click on that. Now click on the **edit** button. And then add the **personal access token** generated from **GitHub** into the password field. And click on the **Save** button.

Now you can access Git.

If you are accessing Git in Android Studio, if asked for a password then add the GitHub **personal access token** instead of your password everywhere.

Share Improve this answer Follow

edited Jan 9, 2024 at 5:58

answered Aug 14, 2021 at 11:03



Jaydip Meghapara
2,833 1 17 14

108 This message means that you're using a password instead of a personal access token to access GitHub over HTTPS, and that's no longer allowed. GitHub has disabled password authentication because it's common for people to accidentally leak their passwords, and while a personal access token can be restricted to limit the damage, a password cannot.

If you haven't explicitly entered your password at a prompt, then it's likely you have a credential manager which is saving your password and sending it without prompting you.

You can follow the directions for clearing your credential manager [listed in the Git FAQ](#):

```
$ echo url=https://account@github.com | git credential reject
```

You should use this same URL, but replace `account` with your own username (e.g., in my case, it would look like `echo url=https://bk2204@github.com`).

Then, the next time you try to push or pull, Git will prompt you for a username and password. For the username, enter your GitHub username, and for the password, generate a new personal access token [on the appropriate settings page](#) and paste it into the password field. If you're working from the command line, you may not see any indication that the password was successfully pasted; this is normal, so just hit `Enter` afterwards.

That will save the personal access token in your credential manager for the next time, assuming you have one set up. If you're not sure if you have one set up, run `git config credential.helper` and see if it outputs anything.

If you don't have one set up, you should add one so that you don't have to memorize your token. Run one of the following commands, depending on operating system:

- `git config --global credential.helper manager` on Windows;
- `git config --global credential.helper osxkeychain` on macOS;
- `git config --global credential.helper libsecret` on Linux (if available); or
- `git config --global credential.helper store` on Linux if `libsecret` isn't available.

Then you can try the push or pull again until you're no longer prompted.

Share Improve this answer Follow

edited Sep 4, 2021 at 22:28

answered Aug 13, 2021 at 17:01



Peter Mortensen

31.6k 22 110 134



bk2204

78.4k 7 109 124

1 Thanks for the info, I just tested `git config --global credential.helper osxkeychain` and `git config --global credential.helper` but nothing is happening in my console, so I will try the first solution. Just one question, what is the `url=https://account@github.com` ? The link of my repot or my github? – Daemes Aug 13, 2021 at 17:22

3 check out [stackoverflow.com/questions/68776300/...](https://stackoverflow.com/questions/68776300/) to change the password on a mac – Evan Aug 13, 2021 at 19:22

2 The technique I've mentioned above will clear the password on any system so Git will prompt for it again. That's why we documented it in the FAQ, because the directions are different for each individual OS and credential helper and it's too hard to write a single answer that applies to all systems when it involves the GUI. – bk2204 Aug 13, 2021 at 19:56

2 `echo url=https://account@github.com | git credential reject` was just what I needed. Thank you! – Nolan Strait Aug 17, 2021 at 18:39

This was the way to fix things, thanks! Apparently I had done everything right according to the numerous reminders, but nowhere did it actually say *how to start using the tokens...* – JoonasD6 Sep 5, 2021 at 5:25



Simplest solution (May 2022):

70

1. Create a new token at [Personal access tokens](#)
2. Copy token (Windows: `Ctrl` + `C`, macOS: `Cmd` + `C`, or click copy icon)
3. Try to push your local repository: `git push`
4. Enter your GitHub user name
5. Paste the token as your password

[Share](#) [Improve this answer](#) [Follow](#)

edited Jun 26, 2022 at 10:18

answered Jan 29, 2022 at 19:03



Peter Mortensen

user3025289

31.6k 22 110 134

I cannot verify that this *all* that is needed, but I it very well might. You see, based on answers above I put the PAT into the Keychain Access (Mac) and then I pasted the PAT for password on the command line when prompted during my 'git clone' command. Maybe I didn't need to put PAT in Keychain Access. I can say that if I do another clone I am *not* prompted for auth. *If* this answer is *all* one needs to do then this is the best answer. – steve Jan 6, 2023 at 19:11

 Generate an access token in GitHub from *Settings* → *Developer settings*.

43 If you have cloned your repository in the past and made it as *origin*, then you can change the authentication so,

 `git remote set-url origin https://<token>@github.com/<username>/<repo>.git`

  If you are going to clone repository after 13 August 2021, then you can use the normal clone command and set the generated access token in the password field.

Share Improve this answer Follow

edited Sep 5, 2021 at 9:16



Peter Mortensen

31.6k 22 110 134

answered Aug 14, 2021 at 14:41



Abdelrahman Elayashy

588 4 14

And it should be "<repo>.git". – G.Vanem Aug 15, 2021 at 12:04

1 If you're like me and have years worth of git repos that you used password based auth on. I have made this: `find . -type f -path '*/.git/*' -name 'config' -exec sed -i 's%https://github.com/%git@github.com:%g' {} \;` It will recursively search in the `.git` folder for any file called config and find and replace the https URL with the git@github one. I'd recommend only doing this on simple repos where you know you have plain vanilla remotes. I haven't tested how it will work on different setups. YMMV! – robmmsmt Aug 17, 2021 at 2:05 

I tried removing the username and it worked: `git remote set-url origin https://<token>@github.com/<repo>.git` – doubleo46 Aug 20, 2021 at 7:07

 For **Ubuntu**, use the following steps

38

- At <https://github.com/settings/tokens>, go and generate a token.
- `git push`
- `username: user_github_username`
- `password: add_generated_token` instead of the password.

 Share Improve this answer Follow

edited Sep 5, 2021 at 9:09



Peter Mortensen

31.6k 22 110 134

answered Aug 14, 2021 at 17:29



Javed A. Baloch

1,128 15 23

A one-command simple solution to solve it

25

If your computer has no SSH key added to the GitHub account, I add information for you to do it at the end of the answer. You should do it first.

 After push failed, then do this:



```
git remote set-url origin git@github.com:{user_id}/{project_name}.git
```

And push again. Then it works.

Let me show my case in the following.

(And I will guide you on how to do your case.)

At the first, when I add, commit, and push, then I meet this issue:

```
milo@MacBook-Pro ~/git/open-idea-pool (master) $ touch test.md
milo@MacBook-Pro ~/git/open-idea-pool (master) $ git add .
milo@MacBook-Pro ~/git/open-idea-pool (master) $ git commit -m "Add test.md"
[master 55b6486] Add test.md
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 test.md
milo@MacBook-Pro ~/git/open-idea-pool (master) $ git push -u origin master
remote: Support for password authentication was removed on August 13, 2021. Please use a personal access token instead.
remote: Please see https://github.blog/2020-12-15-token-authentication-requirements-for-git-operations/ for more information.
fatal: unable to access 'https://github.com/milochen0418/open-idea-pool/': The requested URL returned error: 403
```

And then, my current Git log is the following.

```
milo@MacBook-Pro ~/git/open-idea-pool (master) $ git log
commit 55b6486471282a9840d00d1ad6a76a035a8ee3dc (HEAD -> master)
Author: milochen0418 <milo_chen@mail2000.com.tw>
Date:   Sat Aug 14 21:35:57 2021 +0800

  Add test.md

commit 142716a28ce3b81066ad3e4a123a0a41e2949742 (origin/master, origin/HEAD)
Author: HackMD <no-reply@hackmd.io>
Date:   Wed Aug 7 18:40:00 2019 +0000
```

In the final, this is my way to solve the issue.

```
milo@MacBook-Pro ~/git/open-idea-pool (master) $ git remote set-url origin git@github.com:milochen0418/open-idea-pool.git
milo@MacBook-Pro ~/git/open-idea-pool (master) $ git push -u origin master
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 4 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 272 bytes | 272.00 KiB/s, done.
Total 3 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To github.com:milochen0418/open-idea-pool.git
  142716a..55b6486  master -> master
Branch 'master' set up to track remote branch 'master' from 'origin'.
milo@MacBook-Pro ~/git/open-idea-pool (master) $ git log
commit 55b6486471282a9840d00d1ad6a76a035a8ee3dc (HEAD -> master, origin/master, origin/HEAD)
Author: milochen0418 <milo_chen@mail2000.com.tw>
Date:   Sat Aug 14 21:35:57 2021 +0800

  Add test.md
```

In my case,

1. **{project_name}** <-> open-idea-pool
2. **{user_id}** <-> milochen0418 is the
3. **{branch_name}** <-> master

(your *branch_name* maybe is *main*, but not *master*)

When I push failed, the only thing I need is this one command:

```
git remote set-url origin git@github.com:{user_id}/{project_name}.git
```

Then I push it again by:

```
git push -u origin {branch_name}
```

For the example of my case,

```
git remote set-url origin git@github.com:milochen0418/open-idea-pool.git
git push -u origin master
```

It works.

-- By the way, this is the process of creating an SSH key to GitHub account.

You can refer to these two links to do it. The information here supports Mac, Windows, and Linux.

1. [Issue on adding SSH key to GitHub](#)
2. [Adding a new SSH key to your GitHub account](#)

Furthermore, if you want to clone a new project, you can do the following command:

```
git clone git@github.com:{user_id}/{project_name}.git
```

Share Improve this answer Follow

edited Sep 13, 2021 at 11:37

answered Aug 14, 2021 at 14:07



Milo Chen

4,025 4 23 39

3 I bet this works only because you have an SSH key added to the github account. – HolyBlackCat Aug 14, 2021 at 15:17

1 Maybe add the process of creating an SSH key to the answer? – HolyBlackCat Aug 14, 2021 at 15:32

Thanks a lot. I will add some simple processes of creating an SSH key tomorrow. GitHub official doc also introduces how to do it, but that's is too complicated for junior people. – Milo Chen Aug 14, 2021 at 15:35

I've done to add the process of creating an SSH key to github account by refer two excellent links.
– Milo Chen Aug 15, 2021 at 4:57

1 Advisable to use `ssh` instead of using `username/password(token)` approach. Good answer. Instead of `https://github.com/<repo_url>` use `git@github.com:<repo_url>` – Francois Nov 8, 2021 at 13:48

 You can force your machine to just use SSH instead of HTTPS:

18 `git config --global url."git@github.com:".insteadOf "https://github.com/"`



Share Improve this answer Follow

edited Jun 26, 2022 at 10:02

answered Apr 21, 2022 at 1:03



Peter Mortensen

31.6k 22 110 134



Maf

734 1 9 27

3 this is actually the easiest solution; i wonder if there is no caveats in this – JPdL Sep 7, 2022 at 5:50

1 SSH is even safer than HTTPS for authentication because you'll never need to send your secret over the internet. The other part (git server in this case) only knows your public key and you just need to keep safe your private key in your machine. That's why strong authentication use such approach. You could also protect your private key with something like chmod 400 your_key . Where 4 is the permission for you to read only your key without the risk of modifying it and 0 permissions on your secret for any user in your group and 0 for the rest of the world as well. – Maf Sep 11, 2022 at 9:40

2 I had the same symptoms as OP, and this answer solved it for me, i.e opening the Git config file and replace the HTTPS address with the SSH one. – Björn Larsson Mar 3, 2023 at 12:43

1 This had to be the answer with the most votes... – SdSaati Aug 7, 2023 at 3:13



If you're using macOS and do not find the github.com entry in the KeyChain access:

17

- Try to do a Git action
- It'll ask for the user name
- Enter your GitHub user name
- Generate a new key from Personal access tokens
- In the password field, enter this newly generated token value
- Now you can see a new entry of github.com inside KeyChain Access → *login*

Share Improve this answer Follow

edited Jan 1, 2022 at 17:20

answered Dec 2, 2021 at 2:59



Peter Mortensen

31.6k 22 110 134



vikramvi

3,745 11 52 84



For Mac, go and create your token. You can only see it once. Copy it and store it securely.

16

Open up Terminal and run: `gh auth login`

*`gh` can be installed using Homebrew

Answer the questions. Make sure you pick HTTPS when asked.



```
mac.ca (AbstractDjango) → gh auth login
? What account do you want to log into? GitHub.com
? You're already logged into github.com. Do you want to re-authenticate? Yes
? What is your preferred protocol for Git operations? HTTPS
? Authenticate Git with your GitHub credentials? Yes
? How would you like to authenticate GitHub CLI? Paste an authentication token
Tip: you can generate a Personal Access Token here https://github.com/settings/tokens
The minimum required scopes are 'repo', 'read:org', 'workflow'.
? Paste your authentication token: *****
- gh config set -h github.com git_protocol https
✓ Configured git protocol
✓ Logged in as diek
```

Share Improve this answer Follow

edited Jan 1, 2022 at 17:26



Peter Mortensen

31.6k 22 110 134

answered Nov 16, 2021 at 16:05



diek

695 7 18

Go to *Settings* → *Developer settings* → generate a new token if it does not exist.

15

Settings / Developer settings

GitHub Apps

OAuth Apps

Personal access tokens

Edit personal access token

Make sure to copy your personal access token now. You won't be able to see it again!

Note

test

What's this token for?

Expiration

This token expires on Tue, Sep 21 2021. To set a new expiration date, you must regenerate the token.

Then set the token:

```
git remote set-url origin https://<token>@github.com/<username>/<repo>.git/
```

Share Improve this answer Follow

edited Sep 5, 2021 at 9:30



Peter Mortensen

31.6k 22 110 134

answered Aug 22, 2021 at 13:53



FBA Gimhana

339 3 2

First create the token on GitHub:

13

1. Click on the profile picture and in the menu select **Settings**
2. Select **Developer Settings** in the left menu at the bottom

 3. Select **Personal Access tokens** in the left menu (third option)

 4. Select **Generate new token** and follow the next steps.

 If you have not installed GitHub CLI, you would not find it in your keychain Access. Therefore you should install it first.

 1. Install GitHub CLI for macOS

```
brew install gh
```

For Windows:

```
winget install gh
scoop install gh
choco install gh
```

 2. On the command line, enter `gh auth login`, follow the prompts, and provide the following answers

? What account do you want to log into? **GitHub.com**

? What is your preferred protocol for Git operations? **HTTPS**

? Authenticate Git with your GitHub credentials? **Yes**

? How would you like to authenticate GitHub CLI? Paste an authentication token

 3. Paste the token you created on GitHub and press .

 Share  Improve this answer  Follow

edited Sep 5, 2021 at 10:08

answered Sep 4, 2021 at 10:01



Peter Mortensen

31.6k 22 110 134



Pascal Nitcheu

839 10 8

Generating a GitHub personal access token (PAT)

 9

1. Log in to your GitHub account and open *Settings* → *Developer Settings* Personal Access Tokens.

2. Click on **Generate New Token**.

 Do not forget to copy and the token after generation. It is accessible only once.

Windows

1. Open *Control Panel* → *User Accounts* → *Credential Manager*.

2. Now, look for `git:https://github.com`.

3. Click on *Edit*.

4. Replace the password with the GitHub Personal Access Token.

MAC

1. Click on Spotlight Icon and search for `Keychain Access`.
2. Inside Keychain Access, search for `github.com`.
3. Now, look for the internet password entry for `github.com`.
4. Update the entry.

Share Improve this answer Follow

edited Sep 4, 2021 at 22:55

answered Aug 14, 2021 at 11:40



Peter Mortensen

31.6k 22 110 134



Aayush

121 5

I found that on Windows, it was necessary to *delete* the existing credential in the credential manager. Then, the next time I tried to do a `git push`, it prompted for the Personal Access Token. Simply resetting the existing credential did not work. – J. Peterson Aug 16, 2021 at 7:51

Mac users -- a simple solution.

9

Set up the personal access token in GitHub:

Settings → Developer Settings → Personal access token

Open *Keychain* → type "GitHub"

Change the password to the personal access token.

Try commit/pushing.

Select "Always allow" and business is as usual.

Share Improve this answer Follow

edited Sep 5, 2021 at 8:56

answered Aug 14, 2021 at 17:14



Peter Mortensen

31.6k 22 110 134



Vasilios Kambouras

109 4

I was using `git pull` on Linux with a password before.

9

If that is the case, the only thing you need to do is use **token** in place of **password**.

Generate an access token by going to [Personal access tokens](#).

Here is example of `git pull`

git pull

```
Username for 'https://github.com': gitusername
Password for 'https://yourusername@github.com': //Then enter generated token
```

Share Improve this answer Follow

edited Sep 5, 2021 at 22:54

answered Sep 2, 2021 at 9:28



VK321

6,003 5 48 53

In my case after doing "git pull", just do "git pull" a second time and you should get prompted for the username and password automatically. It shouldn't require you go to delete your password in the OS settings. – P.Brian.Mackey Sep 9, 2021 at 15:27

To those using **Sourcetree** with an *existing repository* you must update your repository URL like so `https://<your_token>@github.com/username/repo.git`

8

This was taken from [this answer](#).

Share Improve this answer Follow

edited Jan 1, 2022 at 17:09

answered Dec 4, 2021 at 20:04



Peter Mortensen

31.6k 22 110 134



Aggressor

13.5k 24 110 188

`git clone GIT USER NAME:<your token>@github.com/username/repo.git` – Beatle Refractor Dec 29, 2021 at 18:24

If you want to clone for the first time:

7

`git clone https://<repository_owner>:<personal_access_token>@github.com/<repository_owner>/<repo>.git`

Share Improve this answer Follow

edited Nov 30, 2022 at 22:04

answered Nov 5, 2021 at 15:40



Peter Mortensen

31.6k 22 110 134



Álvaro Agüero

4,850 1 47 41

It worked!

6

After getting the token key as said [here](#):

[Create Personal Access Token on GitHub From your GitHub account](#)



go to Settings => Developer Settings => Personal Access Token => Generate New Token (Give your password) => Fillup the form => click Generate token => Copy the generated Token (it will be something like `ghp_sFhFsSHhTzMDrjmk54TzuzgthdVfsrta`)

use the following in your terminal:

```
git clone https://your_username:your_github_token@github.com/username/private-repo.git
```

Share Improve this answer Follow

edited Jan 1, 2022 at 17:25



Peter Mortensen

31.6k 22 110 134

answered Nov 27, 2021 at 5:51



vagdevi k

1,725 1 14 25



In Linux, you can generate a new access token in GitHub and directly replace it in place of the password in the `~.git-credentials` file.

4

The password section starts with `:` and ends with `@`.



Share Improve this answer Follow

edited Sep 5, 2021 at 7:48



Peter Mortensen

31.6k 22 110 134

answered Aug 14, 2021 at 17:02



Shriraj Hegde

1,102 11 21



A quick solution for Windows users

4

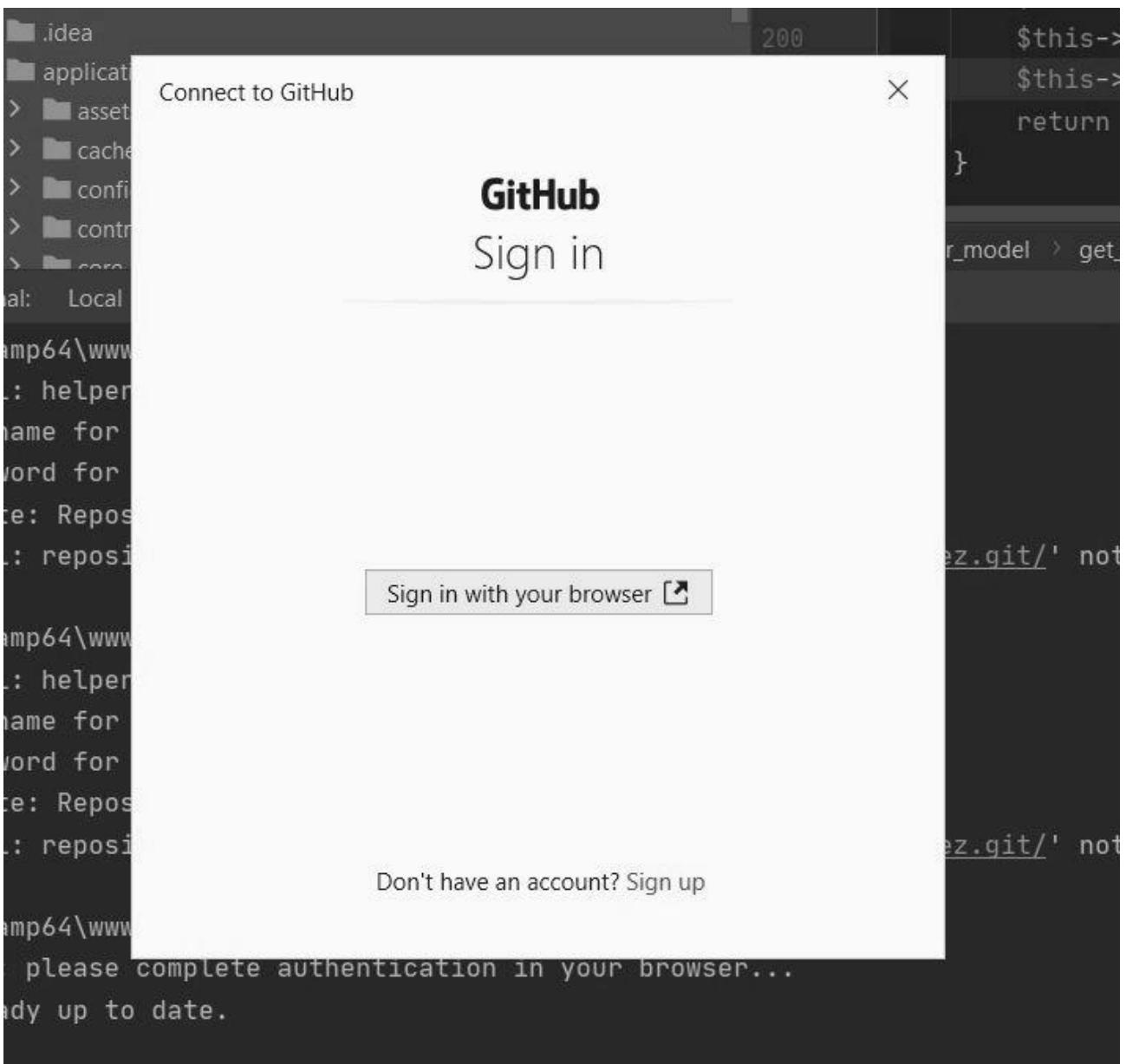
Open [Credential Manager](#) and just remove the already-saved credentials for Git: <https://github.com>.



The screenshot shows the Windows Control Panel Credential Manager. The left sidebar has links for Control Panel Home, User Accounts, and See also (User Accounts). The main area shows a list of credentials under 'Generic Credentials'. One credential is selected: 'git:https://github.com' with details: Internet or network address: githttps://github.com, User name: Ahmadkhan12345566, Password: (redacted), Persistence: Local computer. Buttons for Edit and Remove are shown. Other generic credentials listed include MicrosoftAccount, MSIX-Skype, and various devices. A 'Certificate-Based Credentials' section is also present with a link to add one.

Category	Credential Name	Modified
Generic Credentials	git:https://github.com	Today
	MicrosoftAccount: user=ahmadkhan12345566@outloo...	8/16/2021
	MSIX-Skype for Desktop MSA/live:ahmadkhan12345566	6/15/2021
	MSIX-Skype for Desktop/live:ahmadkhan12345566	6/15/2021
	virtualapp/didlogical	8/5/2021
	SSO_POP_Device	Today
	XboxLive	Today
Certificate-Based Credentials		
Add a certificate-based credential		

After this step, run the command again for pull/push in the terminal. GitHub will ask to log in with your default browser automatically (**make sure you are logged in with GitHub in the default browser**).



After successful login, we got this message:



Authentication Succeeded

You may now close this tab and return to the application.

For more help related to this, you can also watch [this video](#).

Share Improve this answer Follow

edited Sep 5, 2021 at 9:28

answered Aug 20, 2021 at 14:56



Peter Mortensen

31.6k 22 110 134



Arslan Ahmad khan

5,894 1 31 35

 The best solution I have found so far:

4

1. Install GitHub CLI, [brew install gh](#) or [check how to install for other OSes](#)
2. Type `gh auth login` in your terminal
3. Follow through like this:

What account do you want to log into?

Choose GitHub.com

What is your preferred protocol for Git operations?

Choose HTTPS

Authenticate Git with your GitHub credentials?

Choose Yes

How would you like to authenticate GitHub CLI?

Choose Login with a web browser

(Copy the one-time code and paste in the browser)

Press 'Enter' to open github.com in your browser

Share Improve this answer Follow

edited Sep 5, 2021 at 10:10

answered Sep 4, 2021 at 8:48



Peter Mortensen

31.6k 22 110 134



Ash Singh

4,957 2 28 31

In case someone is looking for way to install brew on Arch based distros:

gist.github.com/fardjad/114ebf50a0dd031418bb63b3b134db51 – Harry Adel Sep 7, 2021 at 21:32 

After getting the token key, you can just skip all steps and go with this:

4

```
git clone https://your_username:your_github_token@github.com/username/private-repo.git
```

Share Improve this answer Follow

edited Jun 26, 2022 at 10:02

answered Apr 14, 2022 at 6:16



Peter Mortensen

31.6k 22 110 134



Suraj

894 11 9

This is the most useful answer for me. The original poster (like me) *already had a token* - it just did not work as expected. All the answers starting with "generate a token" are therefore a bit beside the point. My existing token works fine in other places, so I don't see the need to start all over again. And all the answers based on `git remote set-url` store your token in a way that is not terribly secure. My repo wouldn't accept the token at the password prompt, even with a fresh clone, but this worked. Afterwards I just deleted the line from the bash history - no spillage :-) – llude Mar 4, 2024 at 10:02

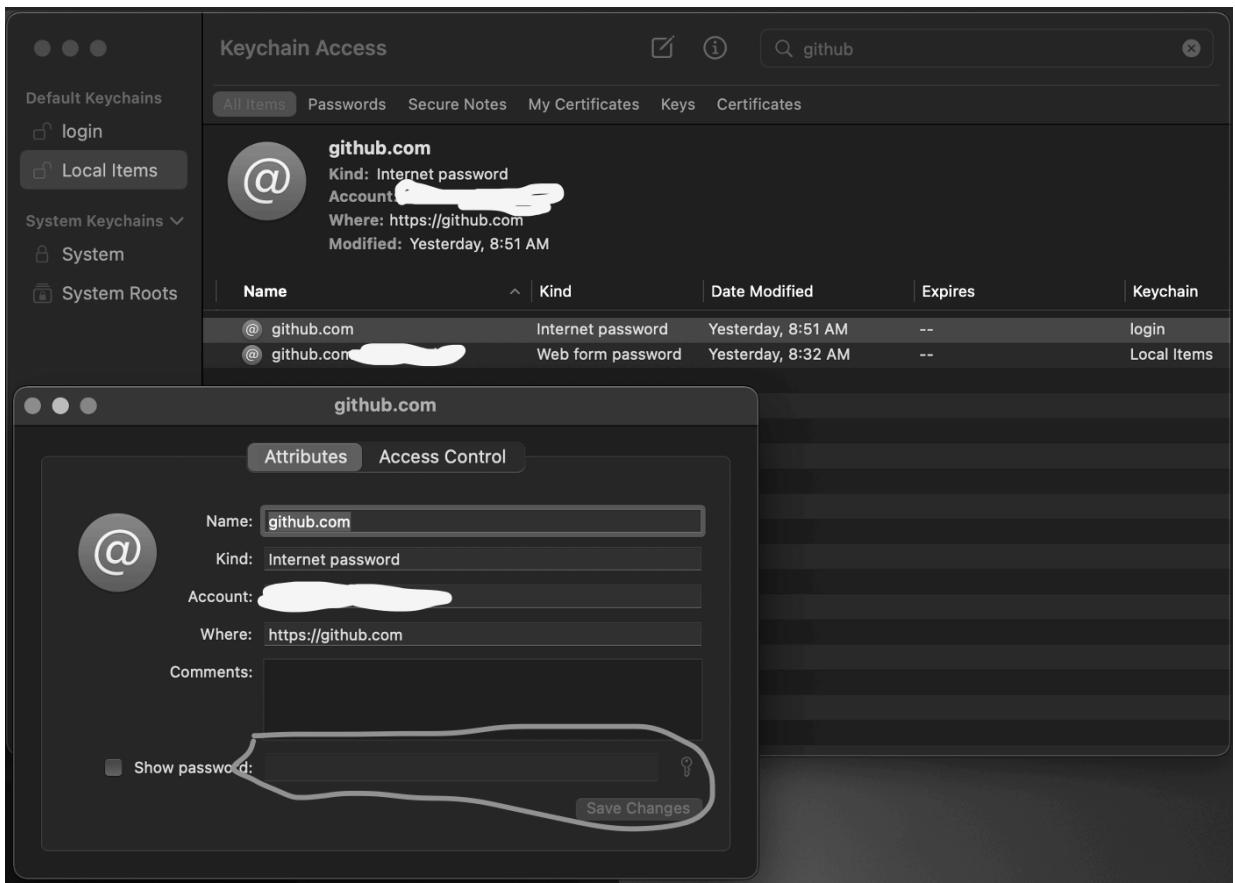
4

Support for password authentication was removed on August 13, 2021. Please use a personal access token instead.

Solution

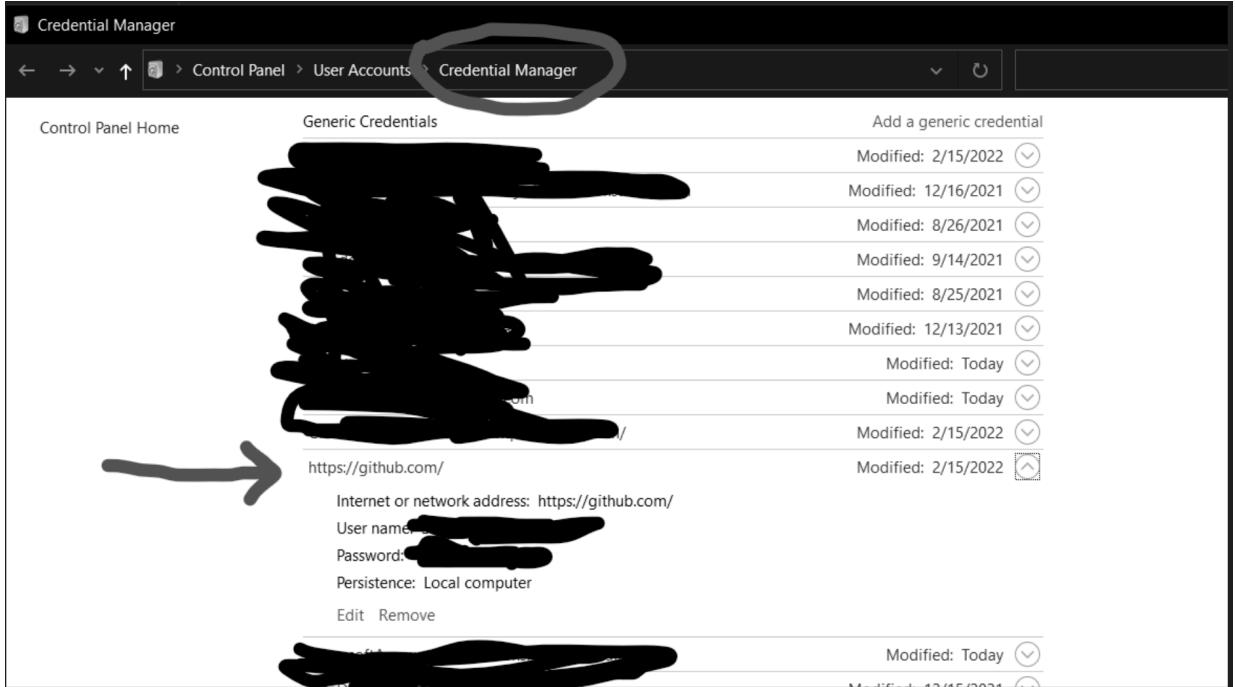
1. [Create a GitHub personal access token \(PAT\)](#) and copy it.
2. For macOS, add it to KeyChain Access under GitHub.

Spotlight Search → type *KeyChain* → select *KeyChain Access* → search for `github.com` → paste your [PAT](#)



For Windows, add the PAT to Windows credentials for your user.

Search → type *Credential Manager* → Add your PAT to *github.com*



3. If the repository is part of an organization, you will also need to sign in to that organization with your personal access token so that the organization will recognize it. If you're already signed in, sign out first.
4. git clone as usual :-)

Share Improve this answer Follow

edited Jun 26, 2022 at 10:17



Peter Mortensen

31.6k 22 110 134

answered Feb 16, 2022 at 21:35



Super Jade

6,489 8 49 70

Android Studio Arctic Fox Solution (Windows 10)

4

I was authenticated in [Android Studio 4.2 \(Arctic Fox\)](#) (latest version) using a personal access token, but still, I was getting this error on August 14, 2021:

```
remote: Support for password authentication was removed on August 13, 2021. Please use a
personal access token instead.
```

I generated a new access token on [github.com](#), removed my GitHub account from Android Studio, and added the account again with the new access token. But still, it was showing the same error.

Reading other solutions on Stack Overflow, I downloaded the GitHub CLI, and added my GitHub account in the Windows command prompt with the access token successfully, and tried to push the repository from Android Studio again, which again failed.

Then following [this article](#), I did the following at the command line:

```
cd <project-directory>
git remote set-url origin https://<TOKEN>@github.com/<user_name>/<repo_name>.git
```

I pushed again from Android Studio, and it finally worked!

Share Improve this answer Follow

edited Oct 27, 2024 at 22:55



Peter Mortensen

31.6k 22 110 134

answered Aug 14, 2021 at 9:31



Raw Hasan

1,426 2 16 33

3

As a security precaution, GitHub automatically removes personal access tokens that haven't been used in a year. To provide additional security, we highly recommend adding expiration to your personal access tokens.

Just follow [Creating a token](#).

Share Improve this answer Follow

edited Sep 5, 2021 at 9:06



Peter Mortensen

31.6k 22 110 134

answered Aug 19, 2021 at 11:30



vaibhav

297 2 3

In three commands:

First create a token as indicated in the official documentation: [Creating a personal access token](#)

3 Then reset the Git credentials

 `git config --global --unset credential.helper`

 Clone or pull your repositories:

 `'git pull' or 'git clone'`

Then enter the token as your password.

If you want to save your token, you can use:

`git config --global credential.helper cache`

 Share Improve this answer Follow

edited Sep 5, 2021 at 9:08



Peter Mortensen

31.6k 22 110 134

answered Aug 19, 2021 at 19:00



Badr Bellaj

13.1k 3 46 47

 Run these commands inside your project directory if the project was cloned before 13 Aug 2020:

3

 `git remote set-url <remote_name> <remote_url>`
`git pull`

 Type your *username* and *personal access token* as the *password*.

 Share Improve this answer Follow

edited Sep 5, 2021 at 9:17



Peter Mortensen

31.6k 22 110 134

answered Aug 14, 2021 at 18:17



Jirawat Yuktawathin

47 2

Example - `git remote set-url origin github.com/myName/GitTest.git` – Akitha_MJ Aug 17, 2021 at 17:05 

1  Next



Protected question. To answer this question, you need to have at least 10 reputation on this site (not counting the association bonus). The reputation requirement helps protect this question from spam and non-answer activity.

Start asking to get answers

Find the answer to your question by asking.

[Ask question](#)

Explore related questions

[git](#) [github](#) [github-api](#) [git-pull](#) [personal-access-token](#)

See similar questions with these tags.