

github报错（完美解决）：获取token。remote: Support for password authentication was removed on August 13, 2021.

简介：这篇文章介绍了如何在GitHub上解决因密码认证被移除而导致的推送错误，通过创建和使用个人访问令牌（token）来代替密码进行身份验证。

前言

- 好久没有上传代码到 GitHub 了，没想到突然报错了，上了网上一搜，才知道，21年8月13号更新了，不再使用密码，开始使用 token 令牌代替密码。
- 网上的资料我也尝试了一下，都不行，没办法，只能研究了，用时两小时，完美解决问题，特来记录一下。

一、需求背景环境

1. win10 系统
2. 本地创建仓库，将代码 push 到 GitHub 上，然后报错。

一、问题描述

1、截图

```
冯坚持@LAPTOP-8MBP8VBI MINGW64 /d/Users/IDEA/user_manager_front_back (main)
$ git push -u origin main
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 'https://github.com/fengfanli/front-and-back.git/'
```

2、报错信息解读

```
冯坚持@LAPTOP-8MBP8VBI MINGW64 /d/Users/IDEA/
user_manager_front_back (main)
$ git push -u origin main
```

```
remote: Support for password authentication was
remote: 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 'https://
github.com/fengfanli/front-and-back.git/'
```

解读：

远程:支持密码认证已于 2021 年 8 月 13 日删除。

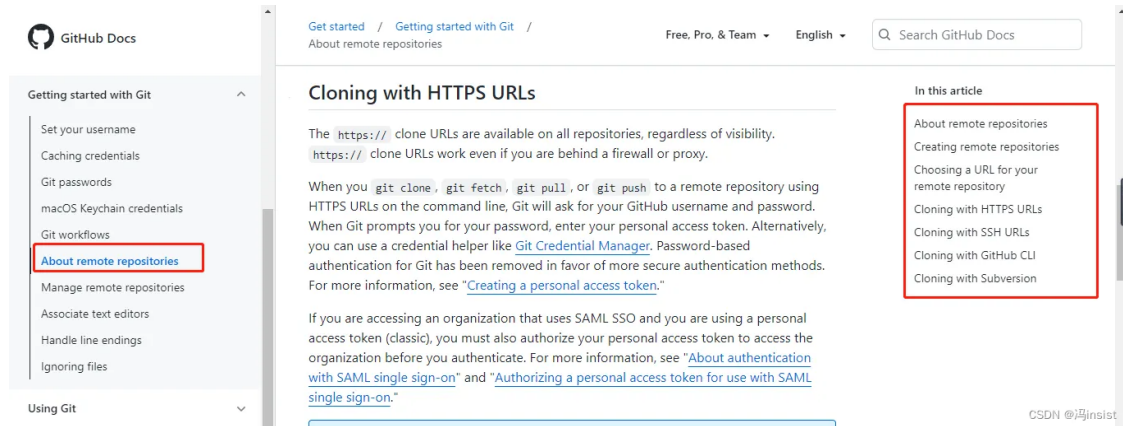
远程：请看 网址（<https://docs.github.com/en/get-started/getting-started-with-git/about-remote-repositories#cloning-with-https-urls>）有关当前推荐的身份验证模式的信息

即，从这个网址中去寻找方法，然后都是英文，但也只能硬着头皮去看了。

二、寻找破解之法

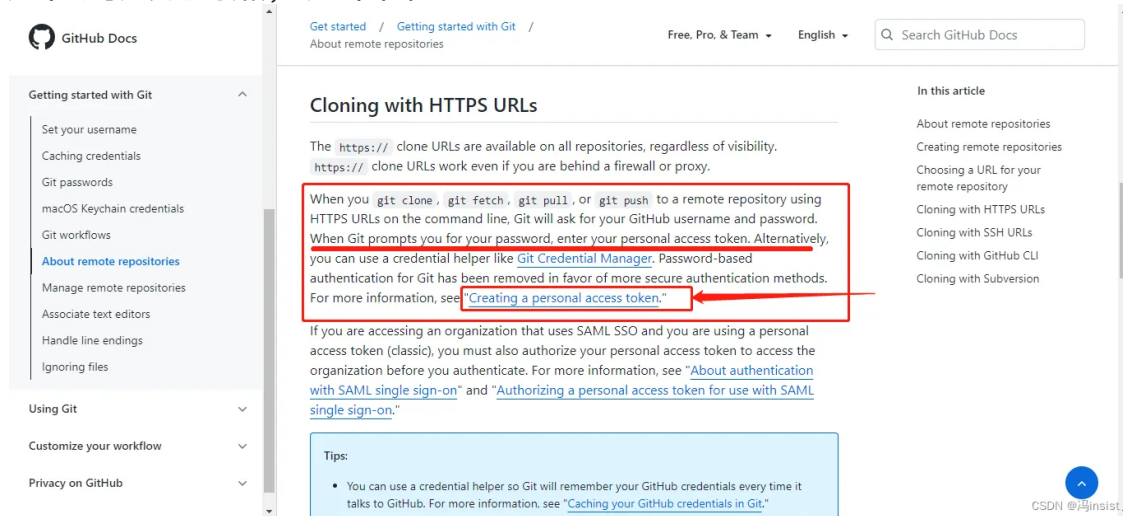
1、进入网址

进入网址：<https://docs.github.com/en/get-started/getting-started-with-git/about-remote-repositories#cloning-with-https-urls>



此页主题是：关于远程仓库

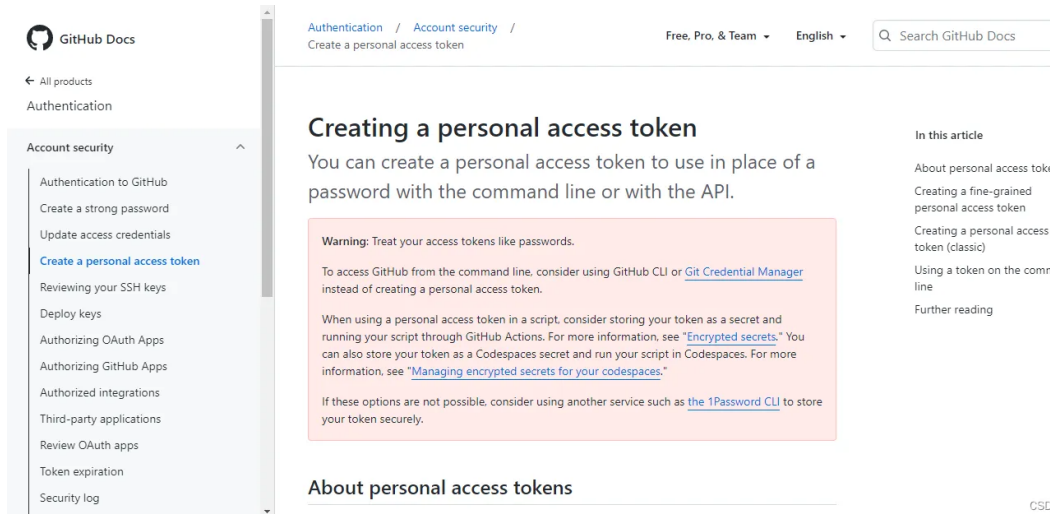
定位到关键的点，如下图：



通过上图中箭头所指的链接去 创建个人访问 token 。

2、创建个人访问 token

1. 点进入网址后如下：<https://docs.github.com/en/authentication/keeping-your-account-and-data-secure/creating-a-personal-access-token>



2. 第一句话就说出来了：

You can create a personal access token to use in place of a password with the command line or with the API.

意思是：

创建个人访问token去使用代替密码在命令行中。

3、两种 token 类型

从上面的截图中的最右边可以看到有两种 token 类型，

1. 第一种是 fine-grained personal access token，即细粒度个人访问令牌
 2. 第二种是 personal access token (classic)，即个人访问令牌（分类化）
- 从字面意义上，第一种是细粒度的 token，使用过后可以发现，可以对一个仓库设置一个 token，还可以设置每一个 permission（权限），粒度比较细。

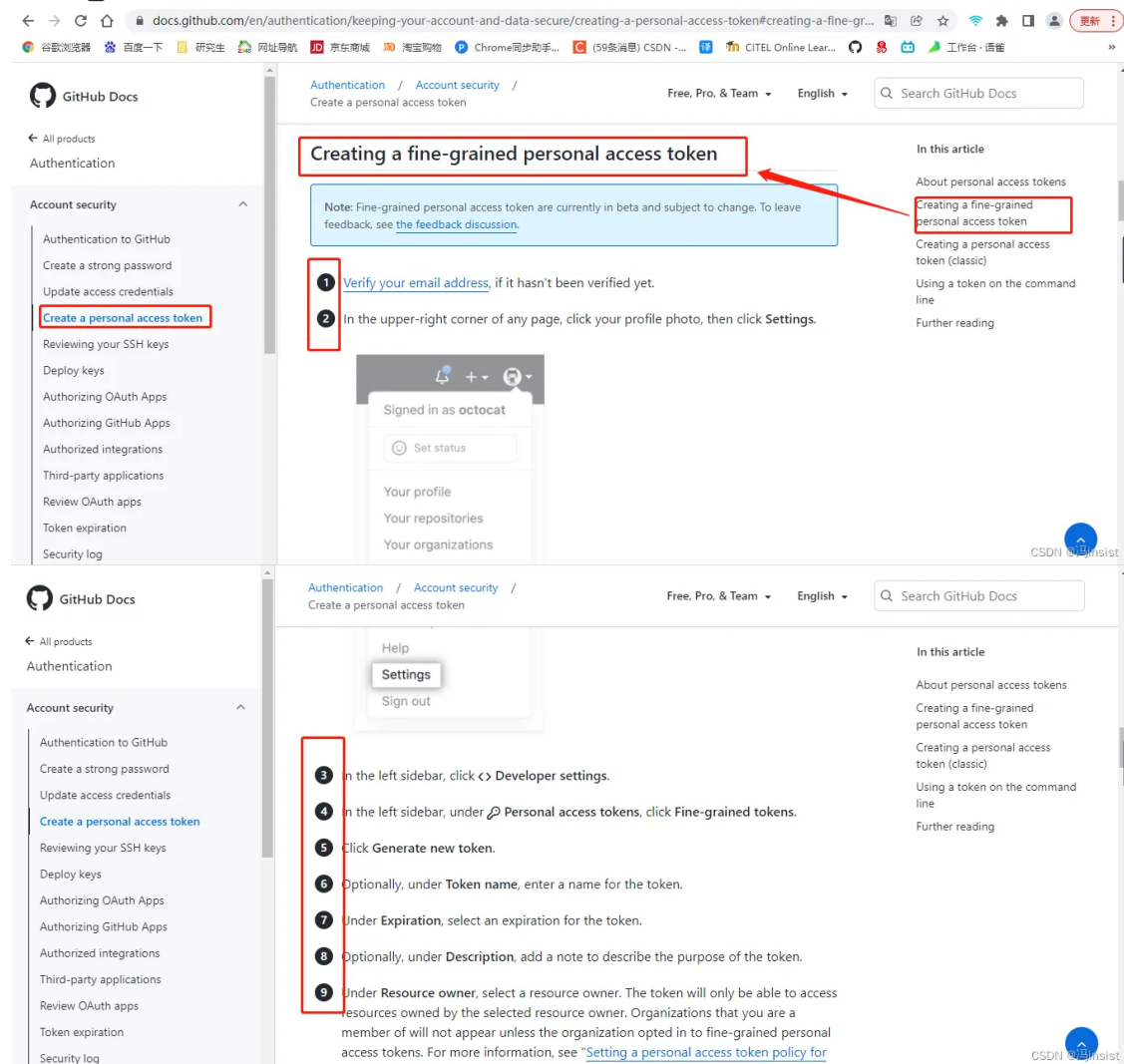
- 第二个没有细研究，有会的大佬可以评论指正，一起学习

三、创建细粒度个人访问令牌

1. 观看网址

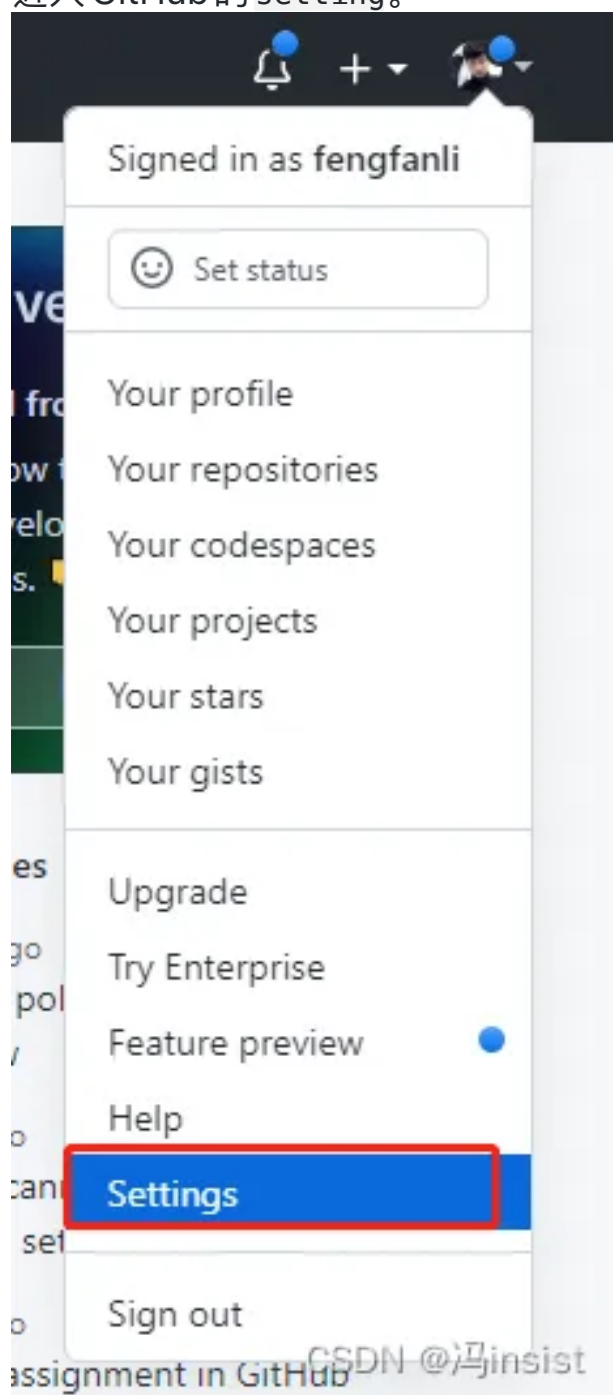
从下面的网址中可以看出，已经给出创建过程，但都是英文，直接看

3.2 吧





2、创建过程

1. 进入GitHub的setting。




2. 最左边的最下面，进入Developer settings


 Packages

 GitHub Copilot

 Pages


 Saved replies

Security


 Code security and analysis

Integrations


 Applications

 Scheduled reminders

Archives

 Security log

 Sponsorship log

 Developer settings

CSDN @冯insist

3. 点击 Personal access tokens 中的 Fine-grained tokens

Settings / Developer settings

GitHub Apps

OAuth Apps

Personal access tokens

Fine-grained tokens

Beta

Tokens (classic)

CSDN @冯insist

4. 点击 Generate new token

Settings / Developer settings / Personal access tokens

GitHub Apps

OAuth Apps

Personal access tokens

Fine-grained tokens

Beta

Tokens (classic)

Fine-grained personal access tokens

Beta

Generate new token

Need an API token for scripts or testing? [Generate a personal access token](#) for quick access to the GitHub API.

Search or jump to... Pull requests Issues Marketplace Explore

GitHub Apps

OAuth Apps

Personal access tokens

Fine-grained tokens

Beta

Tokens (classic)

New fine-grained personal access token

Beta

Create a fine-grained, repository-scoped token suitable for personal API use and for using Git over HTTPS.

Token name

token_for_commit_code

A unique name for this token. May be visible to resource owners.

Expiration *

30 days

The token will expire on Thu, Dec 8 2022

Description

What is this token for?

Resource owner

fengfanli

CSDN

Repository access

☐ Public Repositories (read-only)

☒ **All repositories**
This applies to all current and future repositories owned by the resource owner.
Also includes public repositories (read-only).

☐ Only select repositories
Select at least one repository. Max 50 repositories.
Also includes public repositories (read-only).

Permissions

Read our [permissions documentation](#) for information about specific permissions.

Repository permissions
Repository permissions permit access to repositories and related resources. >

Account permissions
User permissions permit access to resources under your personal GitHub account. >

update lighthouse action workflow tries.

Account permissions
User permissions permit access to resources under your personal GitHub account. >

Overview

23 permissions for all of your repositories >

0 Account permissions >

This token will expire December 8, 2022.

Generate token Cancel

This token will be ready for use immediately.

© 2022 GitHub, Inc. [Terms](#) [Privacy](#) [Security](#) [Status](#) [Docs](#) [Contact GitHub](#) [Pricing](#) [API](#) [Training](#) [Blog](#) [About](#) CSD

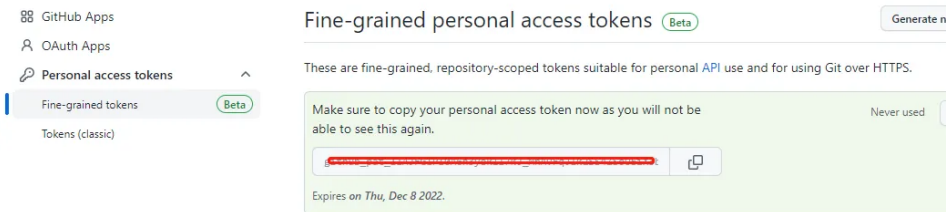
5. 细节如下：

- Token name：令牌名称
- Expiration：令牌过期时间
- Description：描述
- Resource owner：资源的 owner

Repository access：仓库访问，设置所有仓库使用一个 token，还是一个仓库一个 token，我这里选择第二个，所有仓库使用一个 token

- Permissions: 权限, Repository permissions 中的权限我都选中了。
- 最后点击创建即可, 生成下面的 token, 使用 token 替换 push 命令中的密码即可 (token 自己保存, 刷新后就没了)

[Settings](#) / [Developer settings](#) / Personal access tokens



四、使用 token 测试

1. 创建一个仓库, 会看到这些命令, 按顺序执行即可, 在最后一个命令输入你的用户名, 密码就是上面创建的 token。

```
git init
```

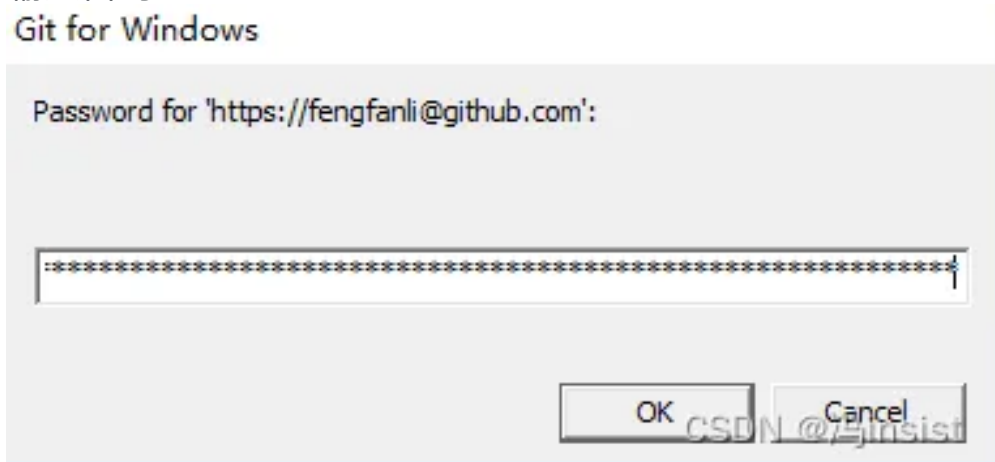
2. `git add README.md`
3. `git commit -m "first commit"`
4. `git branch -M main`
5. `git remote add origin https://github.com/fengfanli/front-and-back.git`
6. `git push -u origin main`

18. 执行最后一个命令

输入用户名:



19. 输入密码:



20. push 成功

```
冯坚持@LAPTOP-8MBP8VBI MINGW64 /d/Users/IDEA/user_manager_front_back (main)
$ git push -u origin main
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), 316 bytes | 316.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/fengfanli/front-and-back.git
   0f87448..a488cbb  main -> main
branch 'main' set up to track 'origin/main'.
```

CSDN @冯坚持