2025年最新RustDesk客户端编译，百分百成功，完美运行！

2025-03-06 / 2 评论 / 10 点赞 / 1790 阅读 / 821 字

1、fork以下两个仓库

```
1. rustdesk/hbb_common
2. rustdesk/rustdesk
```

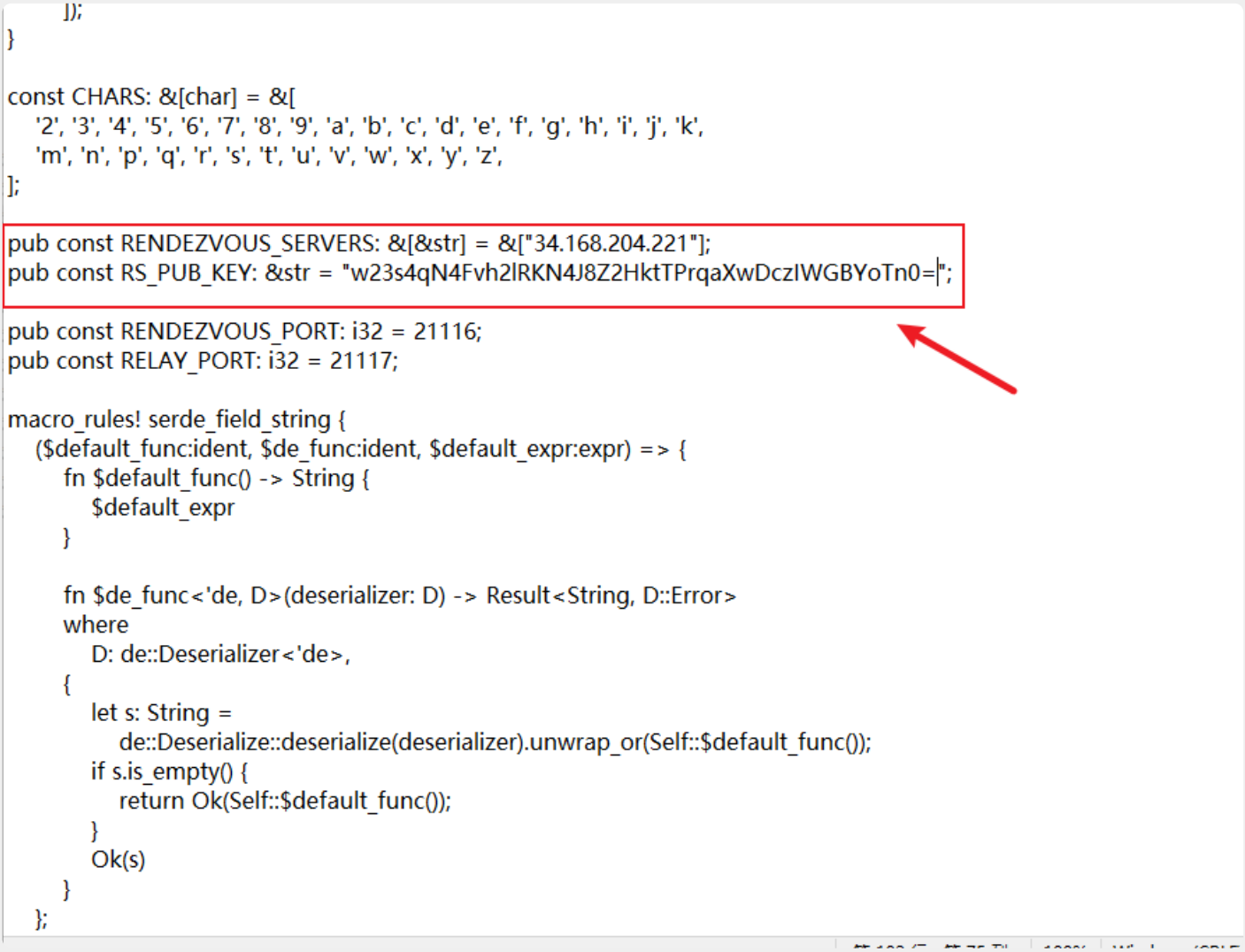
2、拉取这两个仓库到本地

```
# 拉取仓库代码到本地 请替换为你fork后的仓库地址

git clone https://github.com/kjzxs/hbb_common.git

git clone https://github.com/kjzxs/rustdesk.git
```

3、修改本地仓库hbb_common下的src/config.rs，修改位置在101-102行

Explore our developer-friendly HTML to PDF API Printed using PDFCrowd HTML to PDF

```
    ]);
}

const CHARS: &[char] = &[
    '2', '3', '4', '5', '6', '7', '8', '9', 'a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j', 'k',
    'm', 'n', 'p', 'q', 'r', 's', 't', 'u', 'v', 'w', 'x', 'y', 'z',
];

pub const RENDEZVOUS_SERVERS: &[&str] = &["34.168.204.221"];
pub const RS_PUB_KEY: &str = "w23s4qN4Fvh2lRKN4J8Z2HktTPrqaXwDczIWGBYoTn0=";

pub const RENDEZVOUS_PORT: i32 = 21116;
pub const RELAY_PORT: i32 = 21117;

macro_rules! serde_field_string {
    ($default_func:ident, $de_func:ident, $default_expr:expr) => {
        fn $default_func() -> String {
            $default_expr
        }

        fn $de_func<'de, D>(deserializer: D) -> Result<String, D::Error>
        where
            D: de::Deserializer<'de>,
        {
            let s: String =
                de::Deserialize::deserialize(deserializer).unwrap_or(Self::$default_func());
            if s.is_empty() {
                return Ok(Self::$default_func());
            }
            Ok(s)
        }
    };
```



4、推送修改后的代码

```
cd hbb_common/

git add .

git commit -m "修改中继服务器域名和key"

git push
```

Explore our developer-friendly HTML to PDF API Printed using PDFCrowd HTML to PDF

```
HL@DESKTOP-RNOIL7E MINGW64 ~/Desktop/hbb_common (main)
$ git add .

HL@DESKTOP-RNOIL7E MINGW64 ~/Desktop/hbb_common (main)
$ git commit -m "修改中继服务器域名和key"
[main b530d78] 修改中继服务器域名和key
 1 file changed, 2 insertions(+), 2 deletions(-)

HL@DESKTOP-RNOIL7E MINGW64 ~/Desktop/hbb_common (main)
$ git push
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 4 threads
Compressing objects: 100% (4/4), done.
Writing objects: 100% (4/4), 465 bytes | 465.00 KiB/s, done.
Total 4 (delta 3), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (3/3), completed with 3 local objects.
To https://github.com/kjzxs/hbb_common.git
   7cf11f7..b530d78  main -> main

HL@DESKTOP-RNOIL7E MINGW64 ~/Desktop/hbb_common (main)
$
```

5、来到你克隆到本地的rustdesk目录，首先删除原来的子模块

```
git rm -rf libs/hbb_common  # 确保是在 rustdesk 仓库的根目录下执行
git commit -m "删除子模块"
git push
```

```
HL@DESKTOP-RNOIL7E MINGW64 ~/Desktop/rustdesk (master)
$ git add .

HL@DESKTOP-RNOIL7E MINGW64 ~/Desktop/rustdesk (master)
$ git commit -m "删除子模块"
[master 807377ac9] 删除子模块
 2 files changed, 4 deletions(-)
 delete mode 160000 libs/hbb_common

HL@DESKTOP-RNOIL7E MINGW64 ~/Desktop/rustdesk (master)
$ git push
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 4 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (4/4), 336 bytes | 336.00 KiB/s, done.
Total 4 (delta 1), reused 1 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To https://github.com/kjzxs/rustdesk.git
   8f6886192..807377ac9  master -> master
```

6、**重新添加子模块：** 重新将 hbb_common 子模块添加到你的 rustdesk 仓库，并指定你 fork 后的仓库地址

```
# 重新添加子模块，注意替换为你的仓库地址
git submodule add https://github.com/kjzxs/hbb_common libs/hbb_commo
```

Explore our developer-friendly HTML to PDF API Printed using PDFCrowd HTML to PDF

```
n
git add .
git commit -m "重新添加子模块"
git push
```

```
HL@DESKTOP-RNOIL7E MINGW64 ~/Desktop/rustdesk (master)
$ git submodule add https://github.com/kjzxs/hbb_common libs/hbb_common
Cloning into 'C:/Users/HL/Desktop/rustdesk/libs/hbb_common'...
remote: Enumerating objects: 117, done.
remote: Counting objects: 100% (55/55), done.
remote: Compressing objects: 100% (35/35), done.
remote: Total 117 (delta 38), reused 21 (delta 20), pack-reused 62 (from 3)
Receiving objects: 100% (117/117), 106.52 KiB | 302.00 KiB/s, done.
Resolving deltas: 100% (50/50), done.
warning: in the working copy of '.gitmodules', LF will be replaced by CRLF the r
ext time Git touches it

HL@DESKTOP-RNOIL7E MINGW64 ~/Desktop/rustdesk (master)
```

```
HL@DESKTOP-RNOIL7E MINGW64 ~/Desktop/rustdesk (master)
$ git add .
warning: in the working copy of '.gitmodules', LF will be replaced by CRLF the n
ext time Git touches it

HL@DESKTOP-RNOIL7E MINGW64 ~/Desktop/rustdesk (master)
$ git commit -m "重新添加子模块"
[master 77a4749b2] 重新添加子模块
 2 files changed, 4 insertions(+)
 create mode 160000 libs/hbb_common

HL@DESKTOP-RNOIL7E MINGW64 ~/Desktop/rustdesk (master)
$ git push
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 4 threads
Compressing objects: 100% (4/4), done.
Writing objects: 100% (4/4), 457 bytes | 457.00 KiB/s, done.
Total 4 (delta 2), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To https://github.com/kjzxs/rustdesk.git
   807377ac9..77a4749b2  master -> master
```

7、**强制更新子模块：** 这会强制将子模块更新到你 hbb_common 仓库的最新 commit。

```
git submodule sync
git submodule update --init --recursive
```

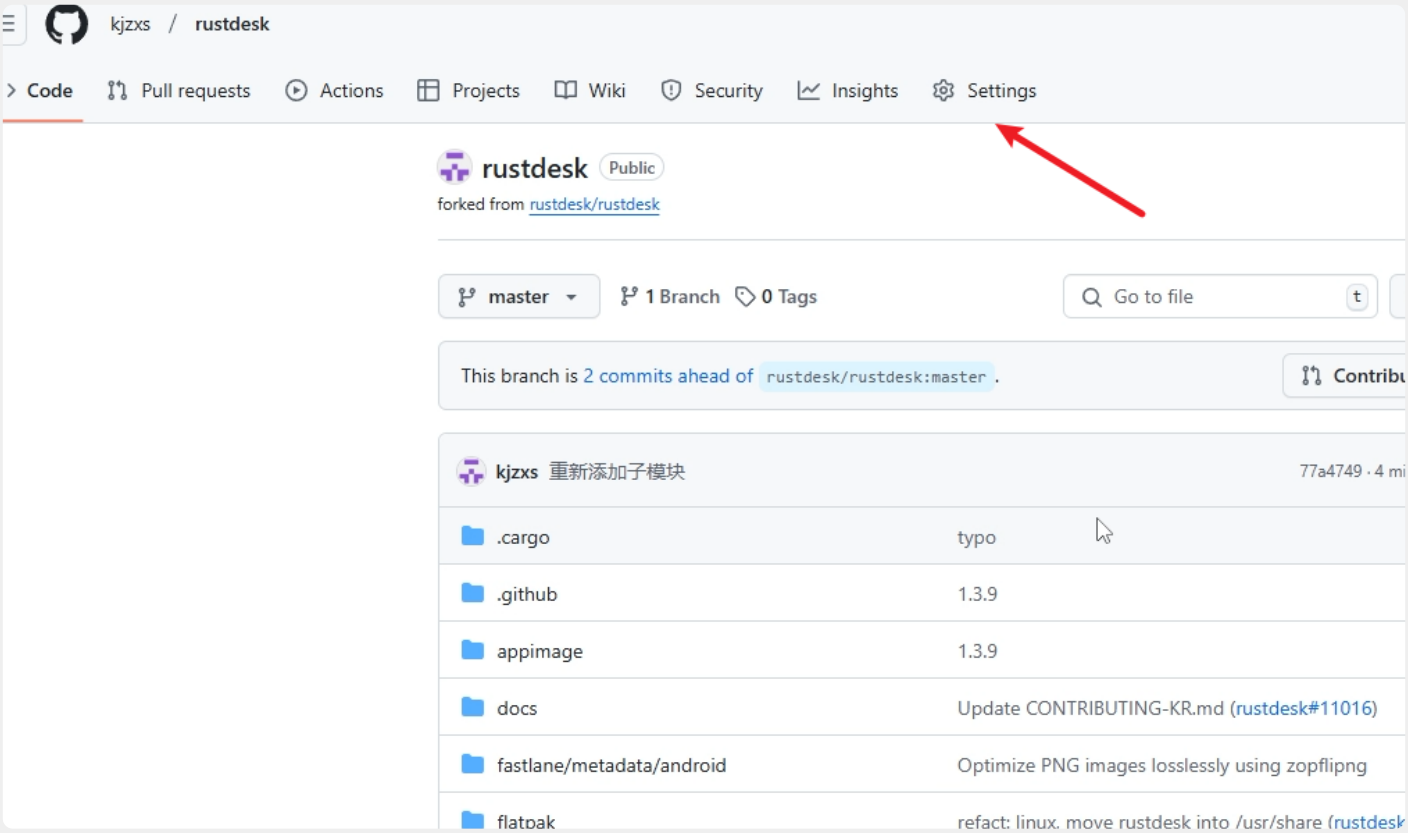
```
HL@DESKTOP-RNOIL7E MINGW64 ~/Desktop/rustdesk (master)
$ git submodule sync
Synchronizing submodule url for 'libs/hbb_common'

HL@DESKTOP-RNOIL7E MINGW64 ~/Desktop/rustdesk (master)
$ git submodule update --init --recursive
```

Explore our developer-friendly HTML to PDF API Printed using PDFCrowd HTML to PDF

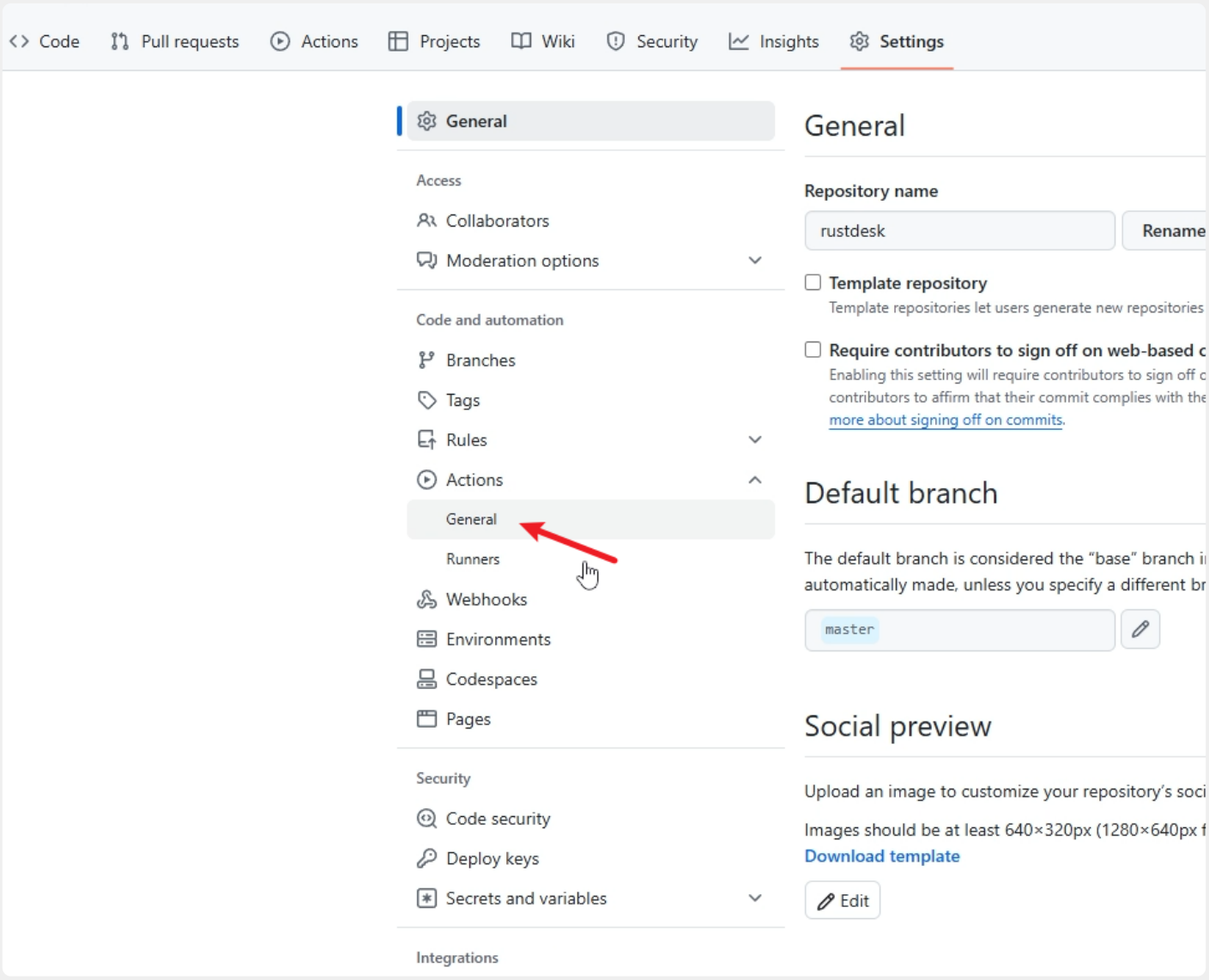
8、提交本地rustdesk仓库的修改

```
1. git add .

2. git commit -m "更新子模块"

3. git push
```

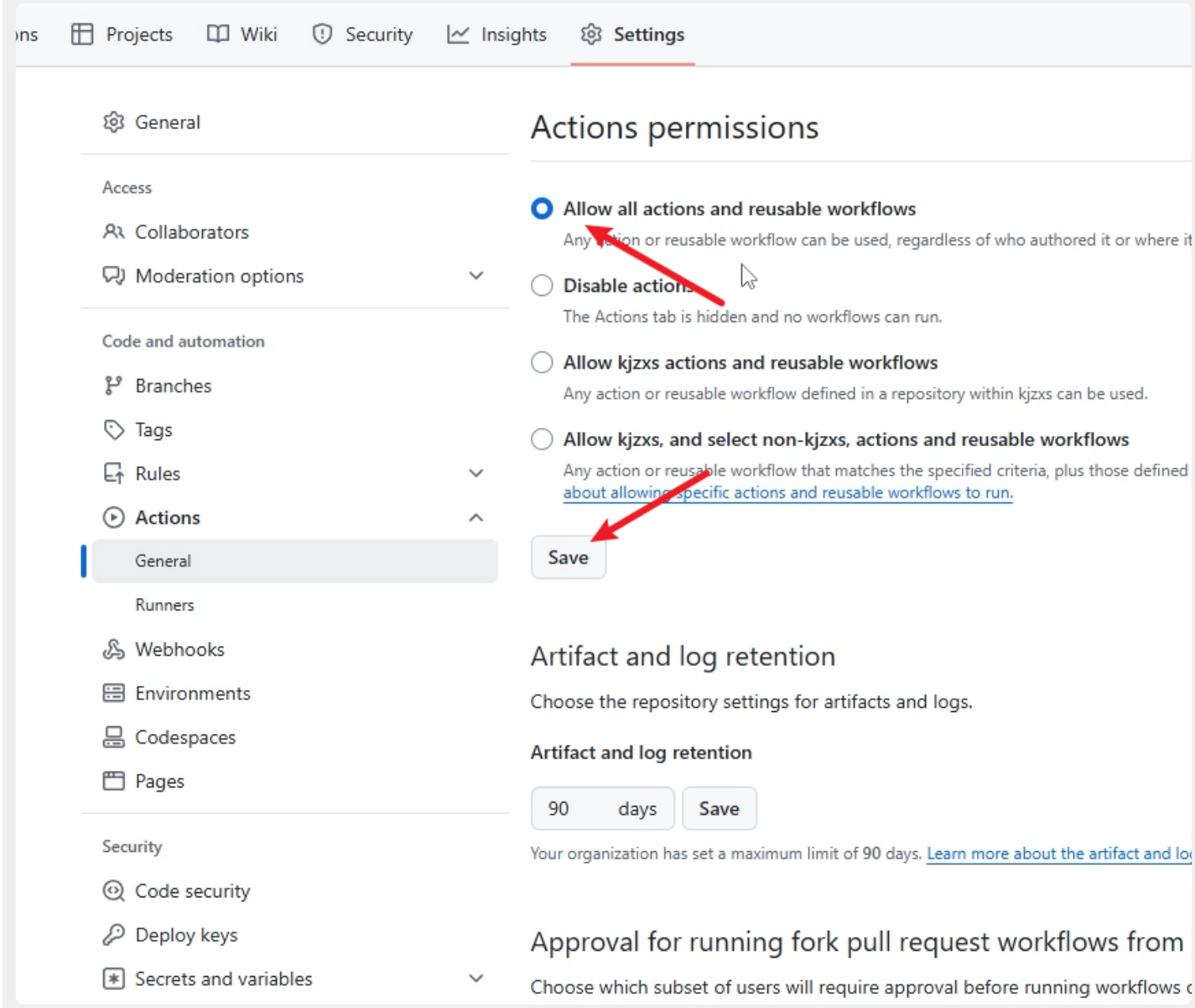
9、在github项目的settings中设置权限



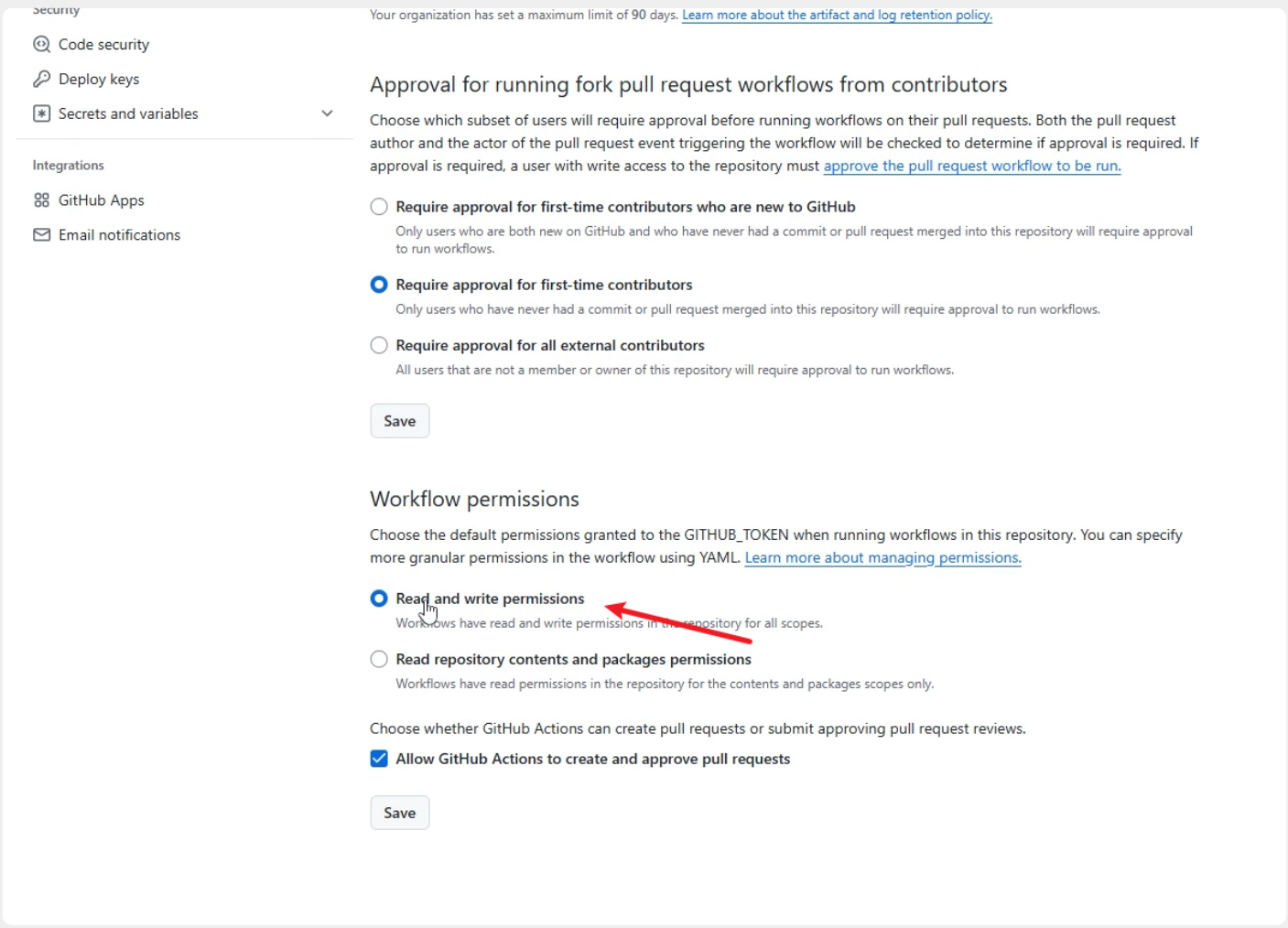
Explore our developer-friendly HTML to PDF API Printed using PDFCrowd HTML to PDF



Explore our developer-friendly HTML to PDF API
Printed using PDFCrowd
HTML to PDF

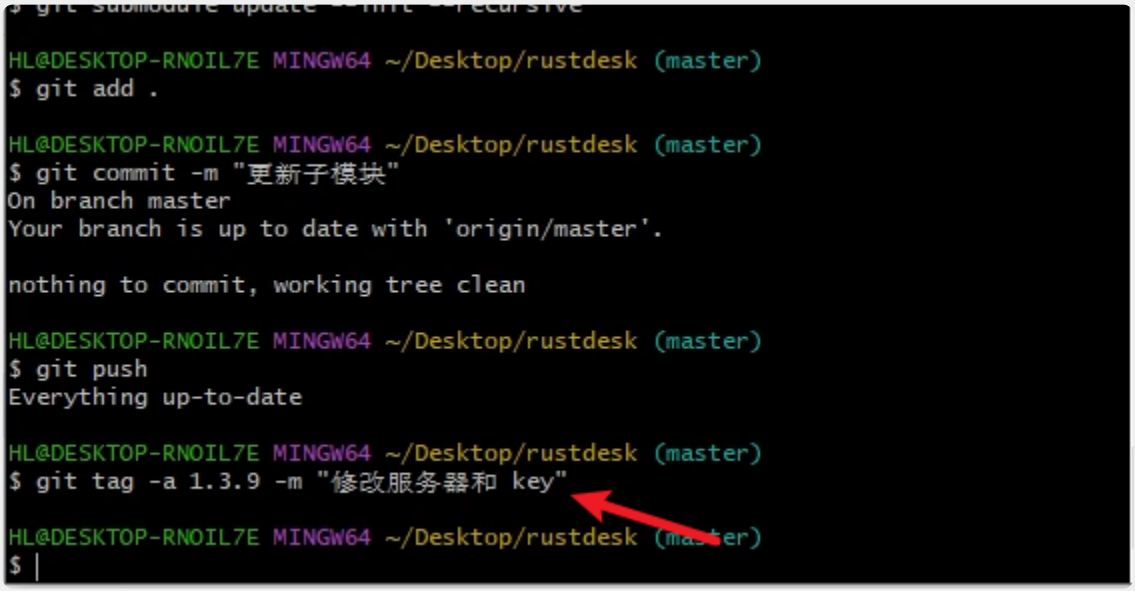


Explore our developer-friendly HTML to PDF API
Printed using PDFCrowd
HTML to PDF



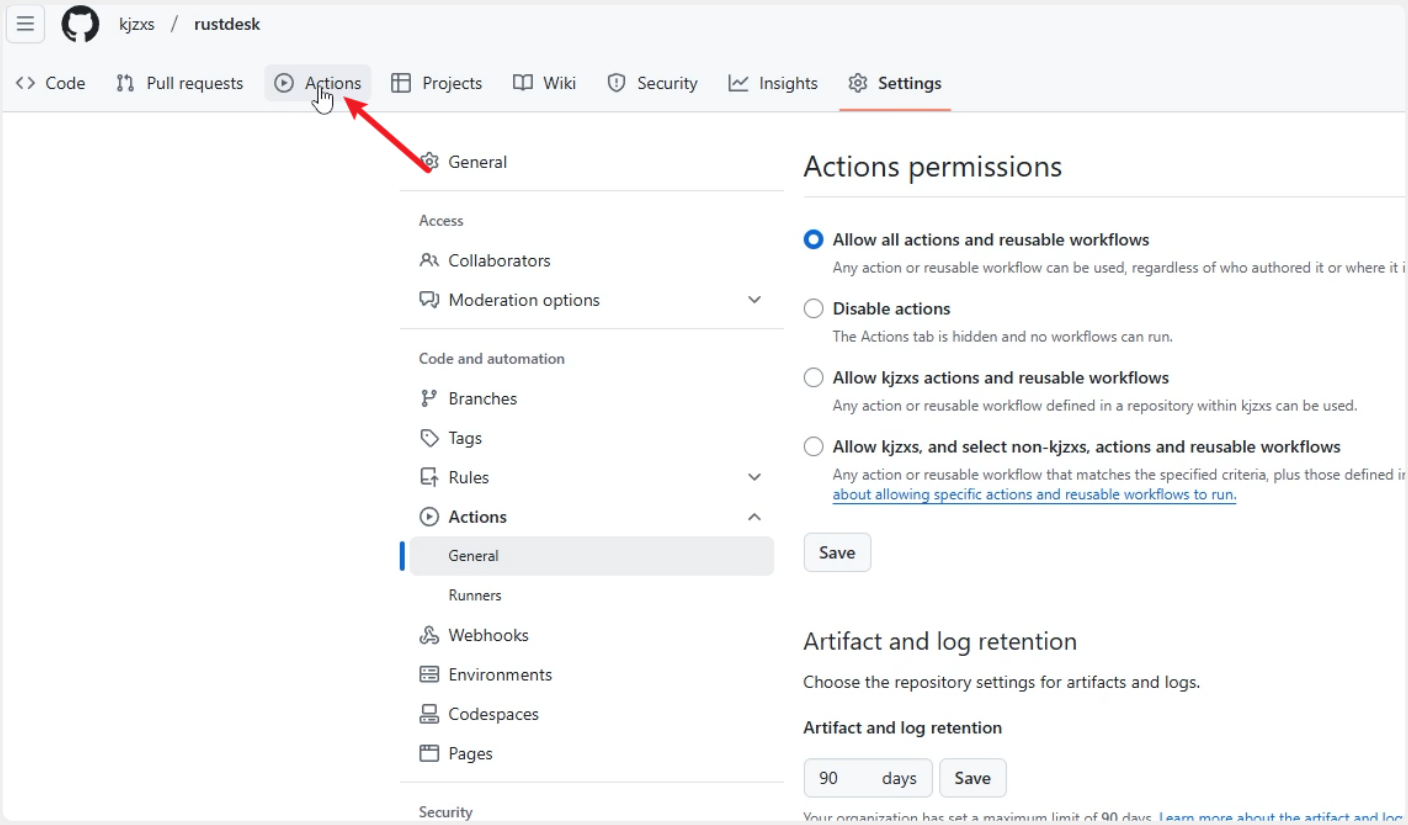
10、打开本地的git命令行，创建一个tag

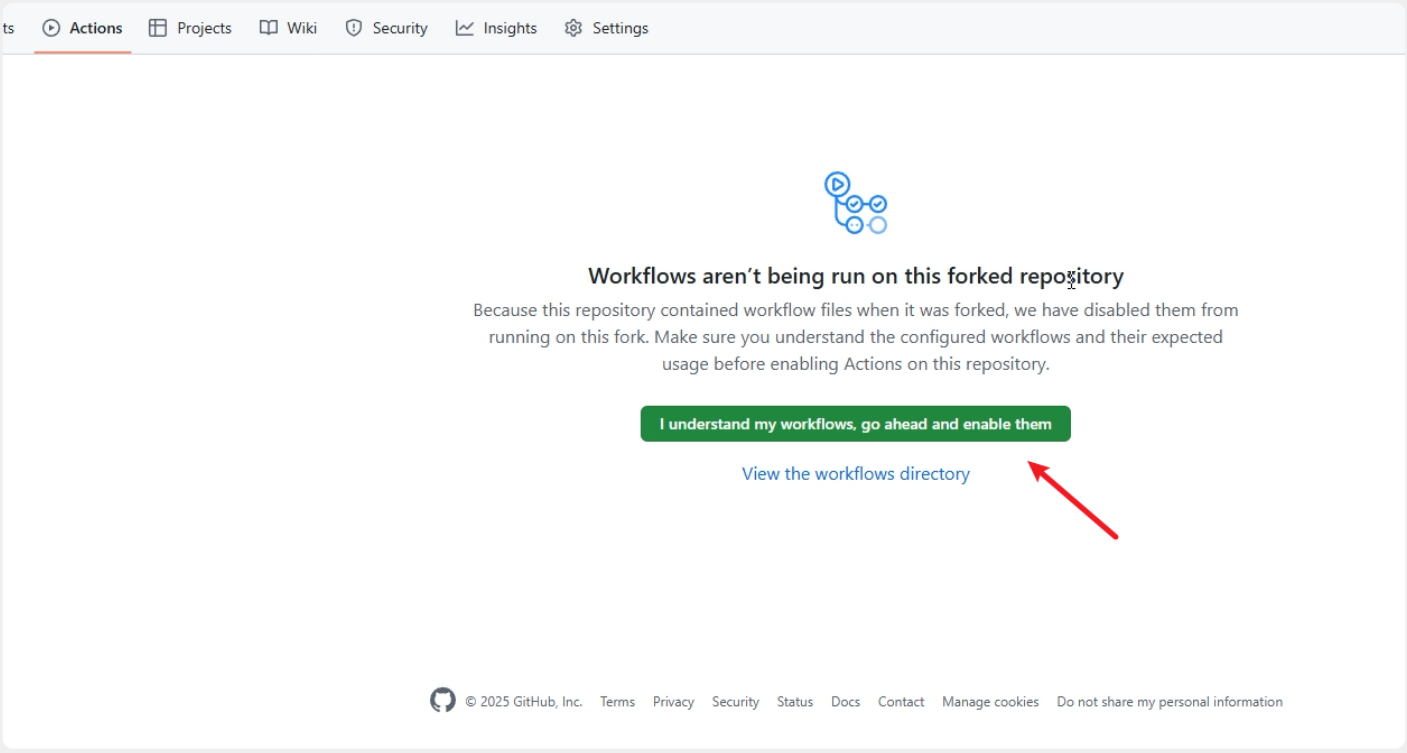
```
1. git tag -a 1.3.9 -m "修改服务器和 key"
```



Explore our developer-friendly HTML to PDF API
Printed using PDFCrowd
HTML to PDF

11、来到github，开启自动工作流



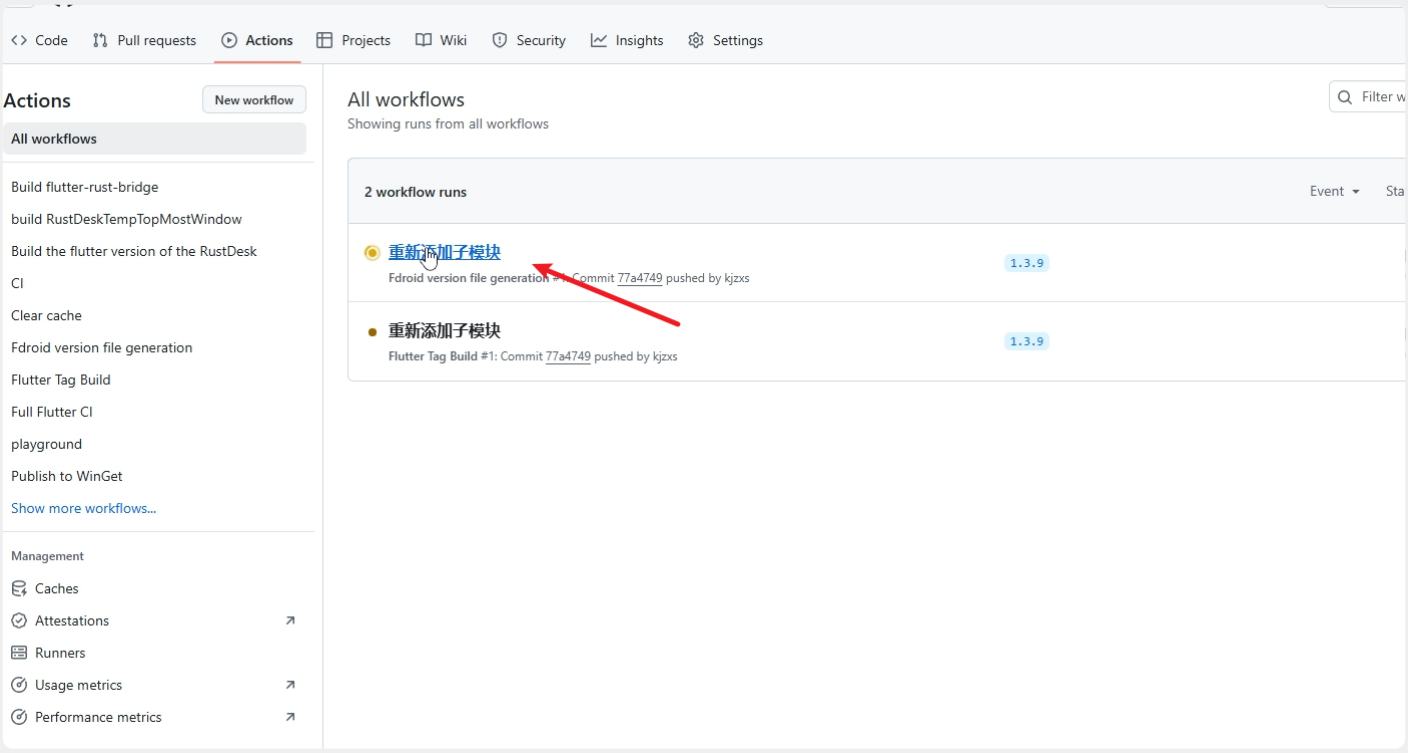


12、来到本地git命令行，提交创建tags的请求，此操作会激活自动工作流，编译工作由此开始运行

Explore our developer-friendly HTML to PDF API
Printed using PDFCrowd
HTML to PDF

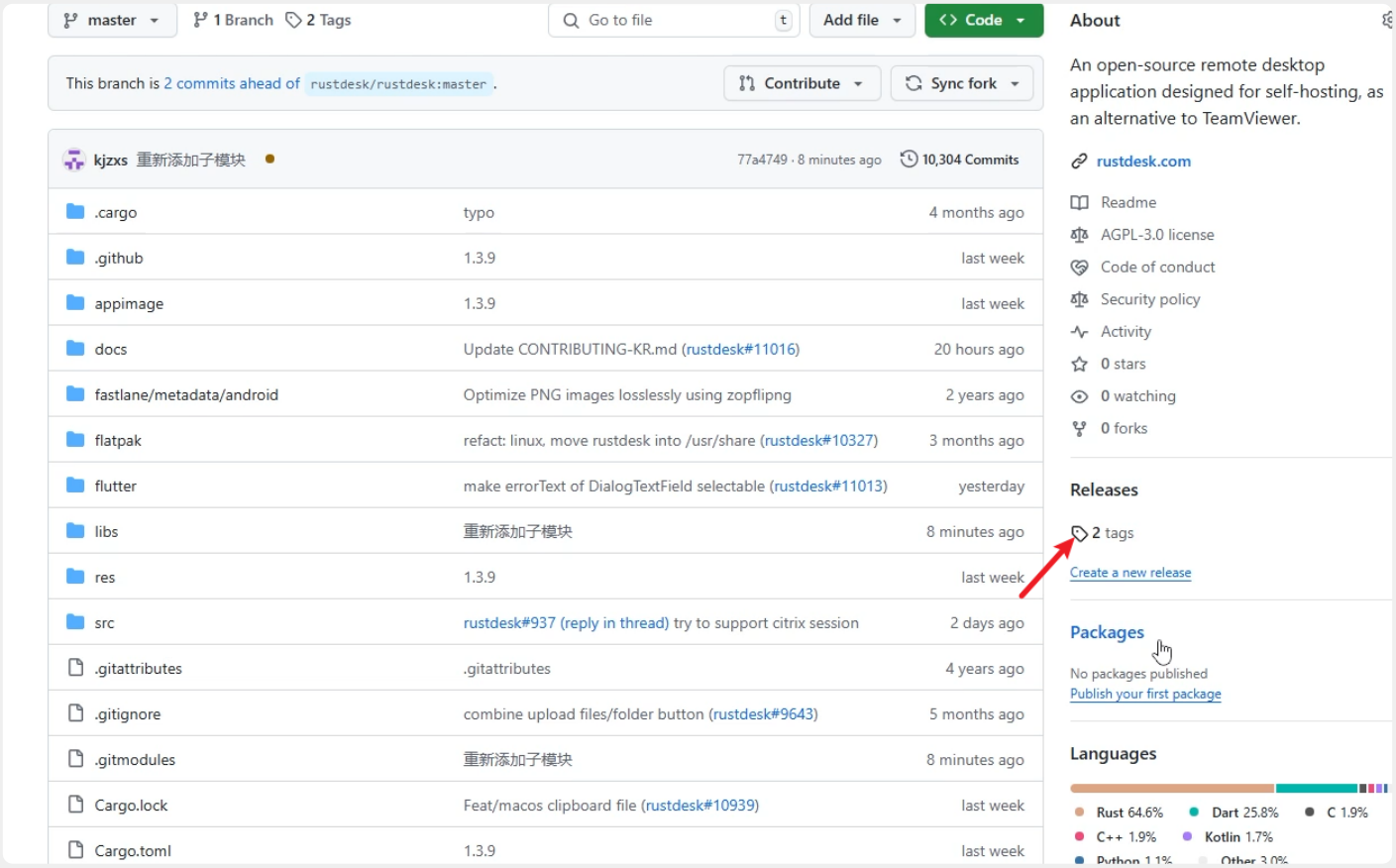
```
1. git push --tags
```

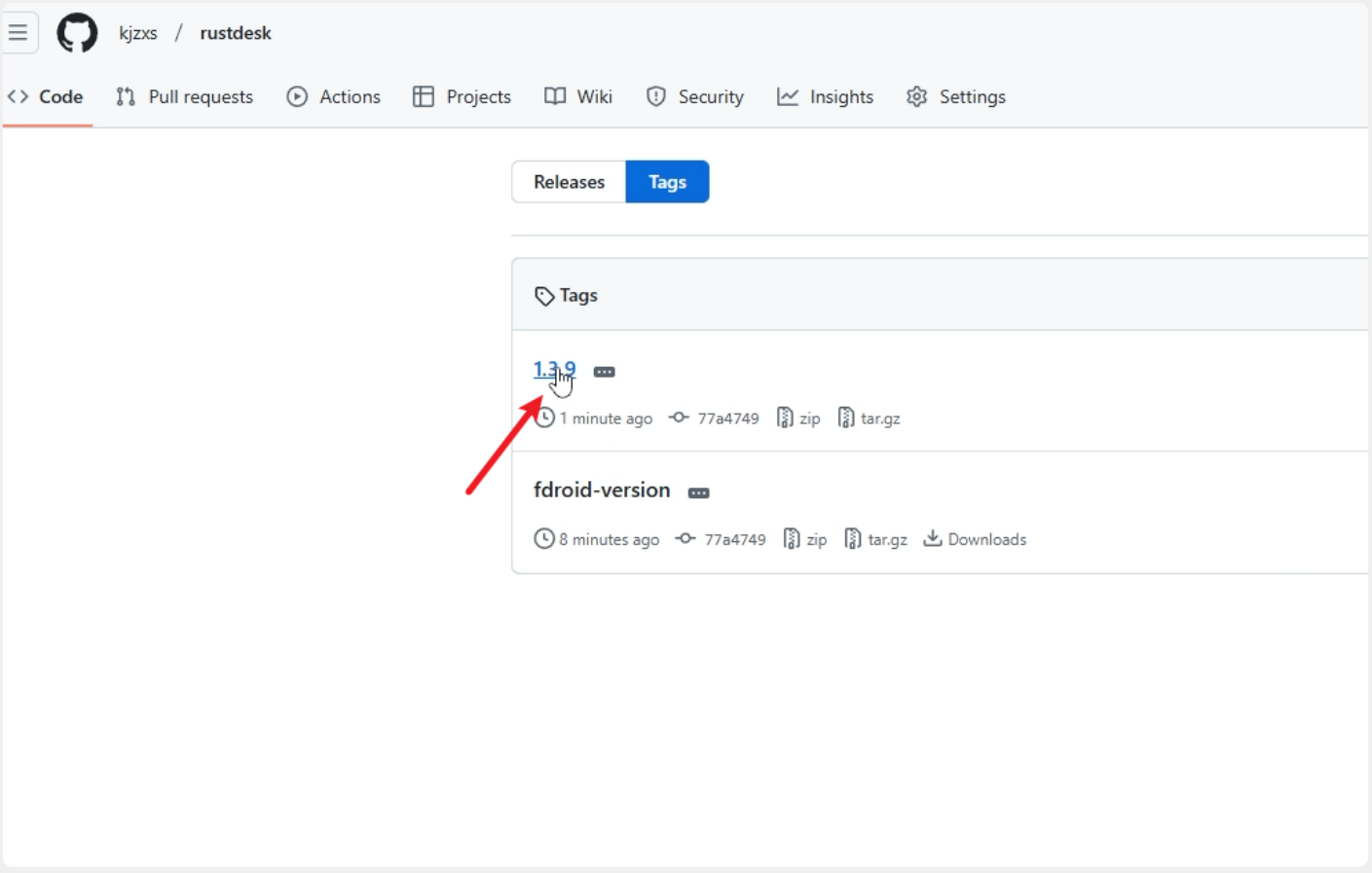
13、工作流会自动开始，耐心等待一个半小时，客户端就编译好了



14.编译完成后，可在tags下查看

Explore our developer-friendly HTML to PDF API
Printed using PDFCrowd
HTML to PDF





Explore our developer-friendly HTML to PDF API
Printed using PDFCrowd
HTML to PDF

rustdesk客户… RustDesk rustdesk中继…

版权归属： Administrator
本文链接： https://www.kxzxs.com/archives/rustdesk-latest-client-compilation
许可协议： 本文使用《署名-非商业性使用-相同方式共享 4.0 国际 (CC BY-NC-SA 4.0)》协议授权

上一篇 下一篇

评论区

Explore our developer-friendly HTML to PDF API Printed using PDFCrowd HTML to PDF