


Personal Access Token

Hasta ahora, la forma más fácil de poner clonar un repositorio en *Github* era usando la *URL* que usa el protocolo *HTTPS*. Estas *URLs* tienen el formato *https://github.com/<nombre-usuario>/<nombre-repositorio>.git* y para clonarlo solamente necesitamos identificarnos con nuestro nombre de usuario y la contraseña. El equipo de *Github* anunció en julio del 2020 la intención de eliminar la autenticación de *Git* usando contraseñas, y en su lugar podemos identificarnos usando *Personal Access Token*, claves *SSH*, *Github App*, etc. Desde el 13 de agosto del 2021 ha dejado de estar operativo, por lo que en esta serie de vídeos veremos las diferentes alternativas que tenemos usando el sistema operativo *Windows*.

En este vídeo veremos cómo crear un *Personal Access Token* y utilizarlo para clonar un repositorio de *Github* en la plataforma *Windows*.


Manos a la obra

Lo primero será crear nuestro propio repositorio en *Github*, vamos a nuestro perfil y pulsamos el botón de crear uno nuevo:

 Lemoncode ▼

[View organization](#)

Repositories


 New

Find a repository...

Y rellenamos el formulario de creación:

Owner *

Repository name *

 Lemoncode ▾

/

Great repository names are short and memorable. Need inspiration? How about **laughing-computing-machine**?

Description (optional)

☐



Public

Anyone on the internet can see this repository. You choose who can commit.

☒



Private

You choose who can see and commit to this repository.

Initialize this repository with:

Skip this step if you're importing an existing repository.

☐ Add a README file

This is where you can write a long description for your project. [Learn more.](#)

☐ Add .gitignore

Choose which files not to track from a list of templates. [Learn more.](#)

☐ Choose a license

A license tells others what they can and can't do with your code. [Learn more.](#)

Create repository

Aquí podemos crear el repositorio público o privado, en este caso no importa.

E incluso podemos seleccionar que añada algún fichero por defecto como el README.

A continuación, vamos a crear un **Personal Access Token** el cuál contendrá los permisos necesarios, para ello, vamos a *Nuestro perfil > Settings > Developer settings > Personal access tokens* para generar uno nuevo:



Signed in as Nasdan



Set status

Your profile

Your repositories

Your codespaces

Your organizations

Your projects

Your stars

Your gists

Upgrade

Feature preview



Help

Settings

Sign out

Account settings
Profile
Account
Appearance
Account security
Billing & plans
Security log
Security & analysis
Emails
Notifications
Scheduled reminders
SSH and GPG keys
Repositories
Packages
Organizations
Saved replies
Applications

Developer settings

Settings / Developer settings

GitHub Apps

OAuth Apps

Personal access tokens

Personal access tokens

Generate new token

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

Personal access tokens function like ordinary OAuth access tokens. They can be used instead of a password for Git over HTTPS, or can be used to [authenticate to the API over Basic Authentication](#).

Vamos a darle un nombre para identificarlo, como por ejemplo **Token para clonar repositorios** y le damos una fecha de expiración:

GitHub Apps

OAuth Apps

Personal access tokens

New personal access token

Personal access tokens function like ordinary OAuth access tokens. They can be used instead of a password for Git over HTTPS, or can be used to [authenticate to the API over Basic Authentication](#).

Note
Token para clonar repositorios

What's this token for?

Expiration *
30 days The token will expire on Fri, Sep 3 2021

Se podría hacer que no expirase, pero esta opción no se recomienda por temas de seguridad, es bueno ir renovando de vez en cuando los credenciales.

Por último, nos quedaría darle los permisos necesarios para poder trabajar con el repositorio, como hacer un clone, realizar commits y push para el desarrollo de nuestra aplicación:

Select scopes

Scopes define the access for personal tokens. [Read more about OAuth scopes](#).

<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"/> security_events	Read and write security events

Y generamos el token pulsando el botón *Generate token* abajo del todo.

Ahora, hay que copiar el token generado y guardarlo a buen recaudo, ya que esta será la única vez que podamos verlo. Si por algún motivo lo hemos perdido, simplemente tendríamos que generar uno nuevo repitiendo el proceso:

GitHub Apps

OAuth Apps


Personal access tokens

Personal access tokens

Generate new token Revoke all

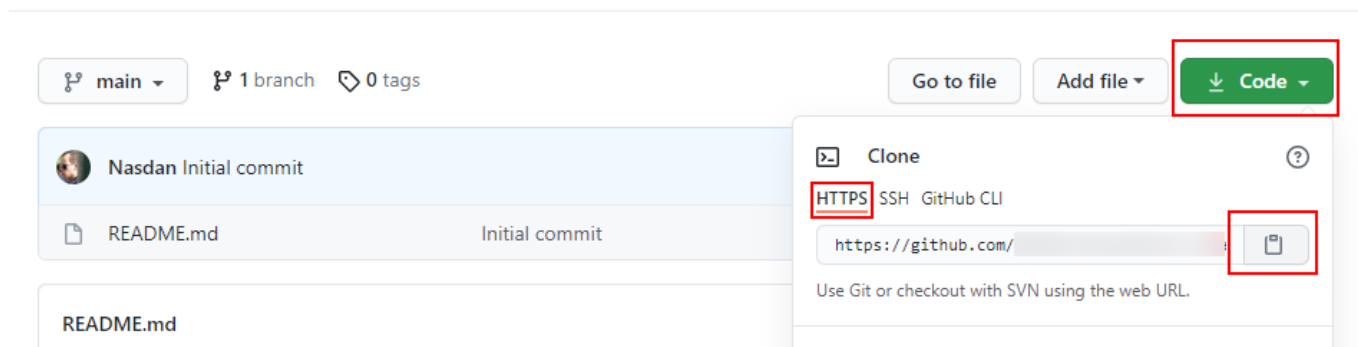
Tokens you have generated that can be used to access the [GitHub API](#).

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

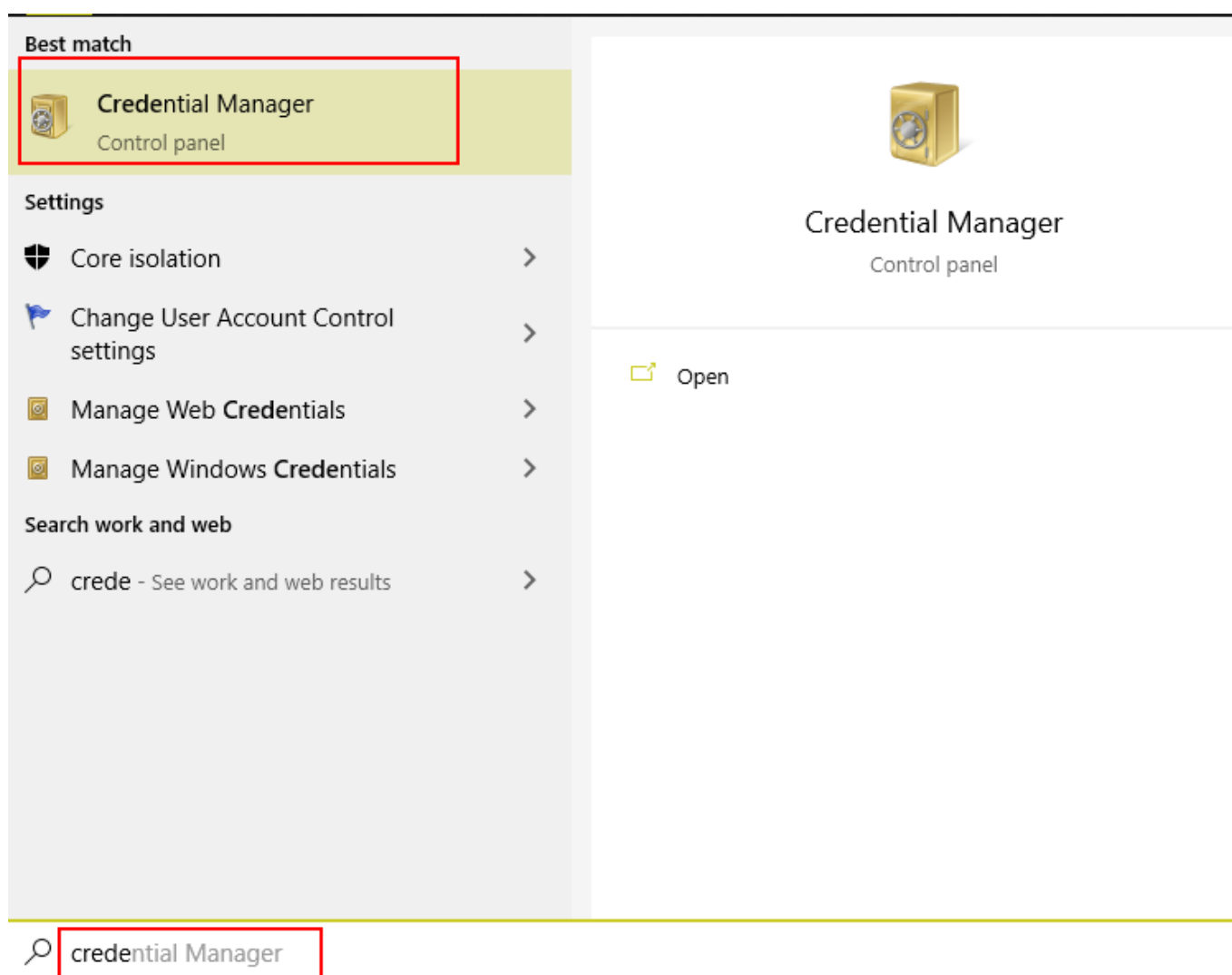
✓ ghp_  Delete

Personal access tokens function like ordinary OAuth access tokens. They can be used instead of a password for Git over HTTPS, or can be used to [authenticate to the API over Basic Authentication](#).

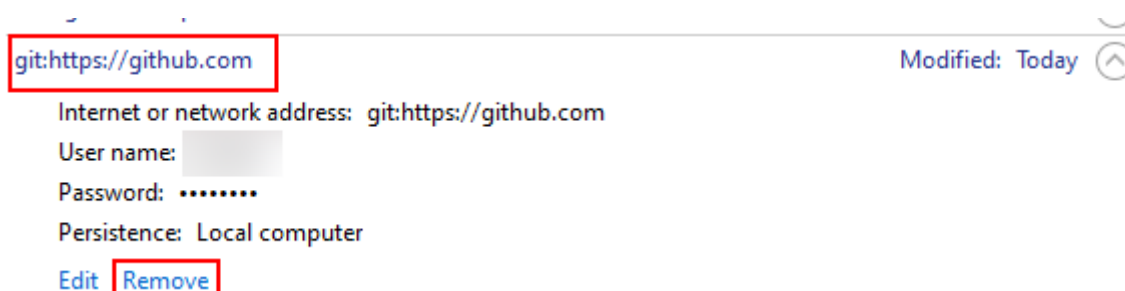
Ya estamos listos para volver a la pantalla principal del repositorio que hemos creado anteriormente, y copiar la URL usando el protocolo *HTTPS*



Vamos a probarlo! Podemos abrir cualquier terminal en nuestra máquina local donde podamos ejecutar los comandos de *git* y antes de nada nos aseguramos que borramos todos los credenciales que tenemos configurados, en *Windows* tendríamos que utilizar el *Administrador de credenciales*:



Y eliminamos las credenciales para *Github*:



Ahora en el terminal, clonamos el repositorio usando nuestro usuario y el *Personal Access Token* como contraseña:

```
git clone https://github.com/<nombre-usuario>/<nombre-repositorio>.git .  
> Username for 'https://github.com': <nuestro-usuario>  
> Password for 'https://...@github.com': <personal-access-token>
```

Recordar si estas en parallels para pegar usar CTRL+SHIFT+Click ratón (ver en keyboards vale como paste para git bash)

Así, tenemos disponibles estas credenciales mientras el token sea válido (recuerda que cuando expire tendrás que renovarlo desde la propia web de *GitHub*).