

Apunte sobre Git

Maxi

August 12, 2023

Contents

1	Configuraciones previas	1
1.1	Generando llaves SSH	1
1.2	Registrando llaves GPG	2
2	Usando Git	2
2.1	Comenzando	2
2.2	Actualizando los cambios realizados	2
2.3	Realizando cambios	3
3	Remote	3
4	Extras	3
4.1	Eliminar algo	3
4.2	Ver cambios modificados	3

1 Configuraciones previas

Antes de hacer algo, es necesario configurar Git y GitHub, respectivamente. Básicamente, el procedimiento va a ser generar llaves SSH y GPG para hacer commits y firmarlos, respectivamente.

1.1 Generando llaves SSH

Generamos una llave con algoritmo ed25519 y seguimos los pasos requeridos.

```
ssh-keygen -t ed25519 -C "your\textsubscript{email}@example.com"
```

Ahora, antes de generar la llave *privada* se necesita comenzar con el cliente ssh-agent de fondo

```
eval "$(ssh-agent -s)"
```

Ya podemos añadirla a nuestra carpeta

```
ssh-add ~/.ssh/id_rsa
```

El archivo con terminación `.pub` es la llave pública, la que se va a añadir a GitHub.

1.2 Registrando llaves GPG

Ya existe una guía para generar y administrar llaves GPG, útiles para firmar.

Si previamente se usaba otra llave GPG, la quitamos.

```
git config --global --unset gpg.format
```

Configuramos nuestra ID primaria en git (se presenta una como ejemplo)

```
git config --global user.signingkey 3AA5C34371567BD2
```

Configuramos git para que firme automáticamente todos los commits

```
git config --global commit.gpgsign true
```

2 Usando Git

2.1 Comenzando

Hacemos un `init` en la carpeta que estamos trabajando. Leer atentamente el output.

```
git init
```

Alternativamente, podemos hacer clonar un repositorio

```
git clone REPO
```

2.2 Actualizando los cambios realizados

Para obtener los últimos cambios realizados en nuestro repositorio, hacemos un `git pull`. También podemos especificar a qué rama o branch queremos realizar el pull, haciendo `git pull origin BRANCH`.

2.3 Realizando cambios

Una vez hechos los cambios en nuestra computadora, necesitamos "avisarle" a Git. Si deseamos hacerlo con todo el directorio

```
git add .
```

Ahora, necesitamos confirmar lo actualizado en un "commit", para finalmente enviarlo a nuestro repositorio. Para ello

```
git commit -m "mensaje del commit"
```

"Pusheamos" el "commit", es decir, enviamos los cambios confirmados a nuestro repositorio, en la branch que estamos trabajando

```
git push origin BRANCH
```

3 Remote

En algún momento, es necesario conectar nuestro directorio Git con GitHub. Remote pone en contacto a los repositorios. El comando `git remote` nos informa el nombre que se le adjudica, el cual es siempre por defecto "origin".

Entonces, el proceso consiste en avisarle a GitHub vía SSH que tenemos un remoto origin añadido en nuestra computadora.

```
git remote add origin SSH
```

4 Extras

4.1 Eliminar algo

Para eliminar un archivo de GitHub, hacemos

```
git rm FILE
```

Luego, es necesario añadir los cambios (`git add`), hacer un commit y pushearlos a GitHub.

4.2 Ver cambios modificados

Para ver si hay cambios no notificados a Git, usamos

```
git status
```