

Aula - Git: Configurando SSH

Aqui aprenderemos o **que é uma chave *SSH*** e também **como criar uma chave *SSH*** do zero no terminal e depois adicioná-la em nosso usuário do *GitHub*.



M1 - Sprint 1 - Aula 14 - Git e Github Criando chave SSH

Kenzie

06:08

A aula a seguir, não é com base no vídeo. As orientações não mudam, somente o exemplo.

Tarefas

1. Criar chave *SSH*
 2. Adicionar chave *SSH* no *GitHub*
-

Passo a passo

O que é uma chave SSH?

Uma chave *SSH* é, resumidamente, uma espécie de “identificador” que atua em autenticações de usuário. As chaves *SSH* são uma forma eficiente de identificar usuários conhecidos, substituindo a autenticação com login e senha. A grande vantagem em relação ao método tradicional, é a segurança: uma chave *SSH* é gerada com um poderoso algoritmo de encriptação, e pode ser comparada com uma senha de **617** dígitos.

1. Criar chave SSH

Abra o terminal/Git Bash do seu sistema operacional e execute o comando a seguir para geramos a chave *SSH* que posteriormente adicionaremos no nosso usuário do GitHub:

Copiar para área de transferência

```
ssh-keygen -t ed25519 -C "seu_email@dominio.com"
```

O que é ed25519?

O ed25519 é o tipo da chave que você está gerando. Você também pode usar a chave tipo *RSA*, mas não é mais recomendada por ser menos segura.

Após executar o comando, você receberá em seu terminal a saída a seguir que tem duas linhas.

Na **primeira linha** o terminal avisa que está gerando as suas chaves, privada e pública com ed25519. Já na **segunda linha** está te perguntando onde você quer salvar as suas chaves *SSH*, entre parênteses e ainda na segunda linha o terminal tem uma sugestão do caminho e nome do arquivo onde será guardada a chave gerada.

Recomendo você não alterar esse caminho, portanto basta pressionar a tecla **ENTER** do seu teclado para confirmar e manter o arquivo salvo no caminho que foi sugerido. Isso vai te poupar alguns configurações que vem caso você mude esse caminho quando for usar a chave *SSH*.

Copiar para área de transferência


```
Generating public/private ed25519 key pair.  
Enter file in which to save the key (/home
```

Após ter apertado **ENTER** o terminal fará mais uma pergunta (a seguir você pode ver como a segunda pergunta aparecerá no seu terminal): **Digite uma senha**

```
(deixe vazio para nenhuma senha): .
```

Copiar para área de transferência

```
Enter passphrase (empty for no passphrase):
```

A seguir a terceira pergunta, nela você tem que escrever a mesma senha que usou no passo anterior, se seguiu a recomendação é só pressionar  e ser feliz.

Sério, sem senha?

Se essa chave *SSH* for usada em códigos privados, em um repositório de uma empresa, por favor use senha!

Copiar para área de transferência

Enter same passphrase again:

Show! Terminamos o processo para criar a chave *SSH*. Se você recebeu uma saída similar a seguir no seu terminal significa que tudo deu certo pra você também.

Copiar para área de transferência

Your identification has been saved in id_ed25519

Your public key has been saved in id_ed25519

The key fingerprint is:

SHA256:8RH02oqduKLudWWrtL3TU0GYnyGmgHVK4I

The key's randomart image is:

+--[ED25519 256]--+

```
|      .EBBo.  o  |
|      +.*.=o = o  |
|      *.=o * + o  |
|      oo B . +  |
|      S = o   .  |
|      . . * *   .  |
|      . . o * . . .  |
|      . . . . .o  |
|      . . . . .  |
```

+-----[SHA256]-----+

Por que a chave SSH tem dois arquivos?

Temos dois arquivos um representa o que chamamos de **chave privada** e o outro de **chave pública**.

Chave privada fica na sua máquina e mantenha ela em absoluto sigilo, se uma pessoa tiver acesso a chave privada ela poderá alterar seus códigos do *GitHub* sem qualquer limitação.

Chave pública pode ser compartilhada sem ter consequências negativas. Essa é a chave que adicionaremos no *GitHub*.

Quando você tenta enviar um código para o *GitHub* usando autenticação com as chaves *SSH*, o *GitHub* testa verificando se você tem posse da chave privada, uma vez que isso deu certo o código é enviado.

2. Adicionar chave SSH no GitHub

Primeiro precisamos do conteúdo que está dentro da chave pública que acabamos de gerar. Portanto, ainda dentro do terminal execute o comando a seguir

```
clip < ~/.ssh/id_ed25519.pub
```

que salvará a chave pública na área de transferência, em seguida vamos colar a informação dentro do nosso Github.

Links!

A base para os próximos passos está na [documentação](#)

- Abra o github com seu perfil.
- Agora clique em profile(Seu Perfil).
- Nas opções que foram abertas clique em settings(Configurações).
- Agora, no menu lateral esquerdo clique em SSH and GPG keys.
- Clique em New SSH key.
- Adicione um título para sua chave, exemplo: "Desktop_trabalho".
- Cole a sua chave com o atalho do teclado ou usando o menu do botão direito do mouse.
- Agora basta clicar em Add SSH key

Configurações feitas com sucesso. Na documentação, você encontra o passo a passo completo, com imagens de referência.

Agora toda vez que precisar enviar ou pegar códigos no *GitHub* poderá usar a autenticação via *SSH* no

lugar da tradicional *HTTPS*.