

Multiplas SSH Key No Github No Mesmo Computador

Digamos que você esteja trabalhando em dois projetos diferentes, com github **diferentes** e você precisa adicionar a **SSH Key** em ambos repositórios. Ao tentar adicionar sua deploy key, você tenta colocar aquela chave que usa no outro projeto mas ao clicar em salvar, se depara com o seguinte erro:

Key is already in use



Erro de Chave em uso

Para resolver esse problema, você deveria adicionar outra ssh key sem perder a que já está em uso. Mas como?

Criando a nova SSH

Gere uma nova key, com o comando de sempre, com um porém, altere o **nome** de como ela será salva, como na imagem abaixo:

```
1 //Comando para gerar nova ssh
2 ssh-keygen -t rsa -C "seuemail@email.com"
```

```
C:\Users\vjordan\Documents\Victor\backefront.github.io\source\posts>ssh-keygen -t rsa -C "seuemail@gmail.com"
Generating public/private rsa key pair.
Enter file in which to save the key (C:\Users\vjordan/.ssh/id_rsa): C:\Users\vjordan/.ssh/id_rsa_trabalho
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in C:\Users\vjordan/.ssh/id_rsa_trabalho.
Your public key has been saved in C:\Users\vjordan/.ssh/id_rsa_trabalho.pub.
```



Nome do arquivo
a ser gerado

Gerando uma nova SSH Key, alterando o nome do arquivo

Como pode ser visto, alterei a opção default de criação de uma SSH, que é **id_rsa** para **id_rsa_trabalho**. Essa vai ser a chave que adicionarei no outro projeto.

Criando arquivo de configuração

Para usar duas ssh keys diferentes, você deve acessar a pasta onde se encontra a atual e criar um arquivo de configuração:

No Windows

```
1 nul>config
```

Linux/Mac

```
1 touch config
```

Assim que gerar esse arquivo config, você deve editá-lo, usando seu editor preferido, adicionando as seguintes configurações:

```
1 Host trabalho.github.com
2     HostName github.com
3     PreferredAuthentications publickey
4     IdentityFile ~/.ssh/id_rsa_trabalho
5
6 Host github.com
7     HostName github.com
8     PreferredAuthentications publickey
9     IdentityFile ~/.ssh/id_rsa
```

Observe que alterei o `Host` e o `IdentityFile`, fazendo referência a minha nova chave gerada e mantive a antiga public key como *default*.

Feito isso, basta adicionar essa nova chave ao seu repositório e alterar no repositório clonado o **remote**, para fazer isso, deve alterar a url para a que você configurou em `Host`, da seguinte forma:

```
1 git remote add origin git@trabalho.github.com:vjordan/projeto2
```

ou se quiser clonar:

```
1 git clone git@trabalho.github.com:vjordan/projeto2
```

Repare que depois do "@" eu coloquei **trabalho.github.com** que é o nosso HOST que criamos. Não se quisermos adicionar outros, basta alterar o valor que vai depois do @

Com isso, você terá N chaves e gerará novas sem perder as antigas, sem problema nenhum!