Criptografia de chaves públicas

Por Ricardo Paulino

Criptografia é um método utilizado para cifrar um texto ou sentença escrita de modo a torná-lo ininteligível. Desse modo grandes empresas e/ou pessoas e processos em que estão envolvidos objetos que requerem sigilo são enviados a outras pessoas, organizações e/ou outros processos.

Um exemplo do uso de criptografia é para guardar repositórios para controle de versão na rede, com esse recurso podemos manter uma cópia dos códigos fontes de nossos programas em repositórios na rede e assim controlarmos suas versões de acordo com nossas necessidades.

O NetBeans é um software que fornece um ambiente integrado de desenvolvimento que quando tem o controle de versão configurado manté repositórios atualizados em sincronia com as gravações de seus arquivos.

Desse modo eu posso voltar a versão de um arquivo fonte quando acho que algo deu errado e mantenho um controle bem mais sistemático da minha programação. Aconcelho aos programadores que ainda não usam esse tipo de tecnologia a experimentarem esses recursos, não vão se arrepender.

Com o uso adequado das ferramentas de criptografia é possível enviar informações sigilosas através de e-mail ou através de comunicação remota, é o que faz o NetBeans, mantendo-se um nível de segurança bem aceitável para a troca de informações.

De um modo geral os dois tipos de **ambientes** onde mais é usada a criptografia:

1. Aquela utilizada para cifrar informações em arquivo ou cartas utilizando uma sequência de convenções combinadas para transformar a informação original de modo a torna-las ilegíveis por parte de quem viola a mensagem.
2. Aquela utilizada também para cifrar informações de cunho político, diplomático, militar e etc. Neste caso ocorre a modificação codificada de um texto de forma a impedir a sua compreensão por pessoas não autorizadas e que não tenham acesso à tabelas de conversão de caracteres para decifrar a mensagem.

Existem dois algorítimos básicos utilizados para se realizar a ***criptogafia***, são eles os **algoritmos de chave simétrica** e os **algoritmos de chave assimétrica**. Os ***algoritmos de chave simétrica*** são uma classe de algoritmos muito utilizados na criptografia, eles usam uma única chave para as operações de cifragem e decifragem de textos e códigos.

Enquanto as operações envolvidas na criptografia de chave simétrica se utilizam operações mais simples a ***criptografia de chave pública ou assimétrica*** utiliza duas chaves, sendo a primeira para a distribuição a todos os correspondentes de forma ***pública***, por isso a chamamos de chave pública, e a outra sendo conhecida apenas pelo seu ***dono*** permanecendo, em grande arte das vezes, no ambiente físico de sua máquina, a qual chamamos de ***chave privada***.

O algorítimo RSA é um algorítimo de chave assimétrica que usa números para cifragem da ordem de 10^{100} tornando a decifragem do código por que não possui a chave pública muito difícil. Esse é o algorítimo que possibilita a assinatura digital, ele já colocou por terra todas as tentativas de quebrá-lo .

Algoritmos de chave pública podem ser utilizados para conferir autenticidade e confidencialidade aos dados de um documento, ese é o caso da assinatura digital, vejamos estes conceitos:

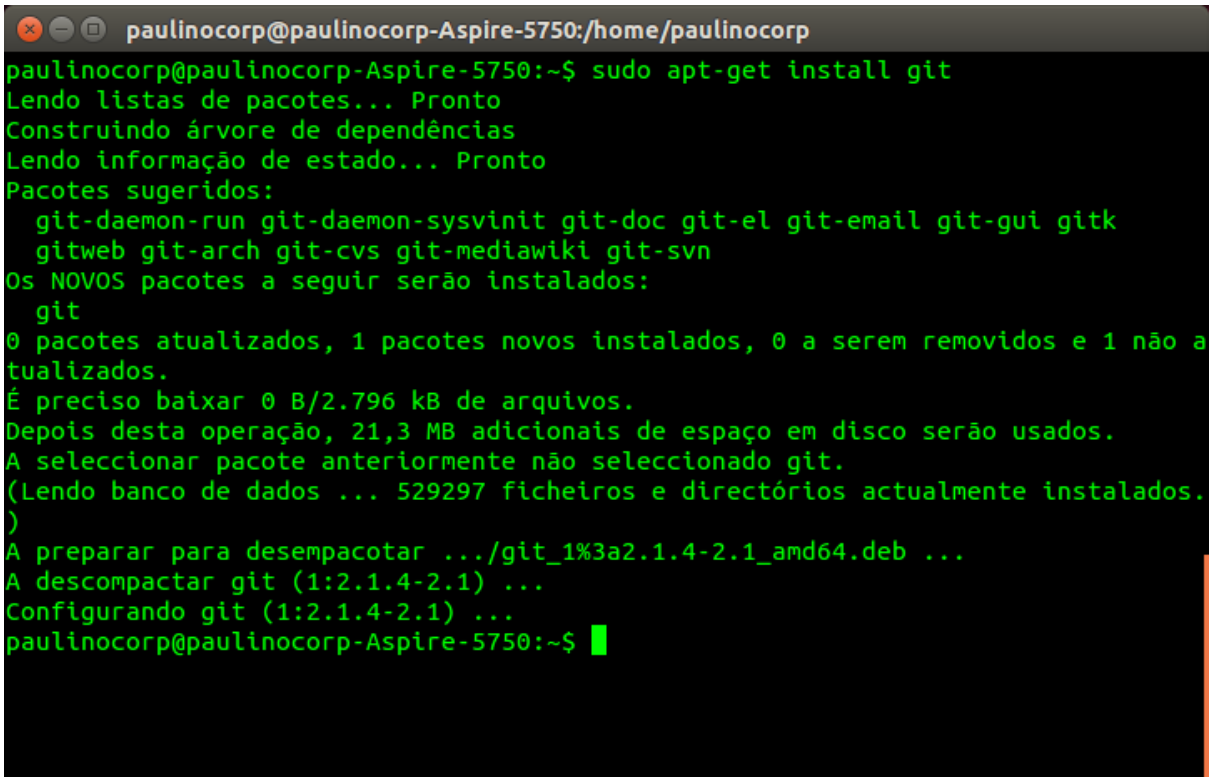
- **Confidencialidad**e: A chave pública é utilizada para cifrar o texto de uma mensagem ou código garantindo que apenas o seu dono e o portador da chave privada possam decifrá-la, evitando assim que terceiros possam ler a mensagem ou o código. Através desse mecanismo você garante ao cliente ***ssh*** utilizado para comunicações através de protocolos seguros de comunicação o acesso para alter dados desde que ele seja portador da chave privada e esta esteja na máquina onde a chave foi criada, ou seja autorizada a acessar o meio físico de dados onde está o seu fonte original. Desse modo um cliente ***ssh*** pode por exemplo alterar configurações de servidores e até mesmo instalar e desinstalar programas em servidores localizados na nuvem acessando seus dados remotamente acessados via ***putty***.
- **Autenticidade**: A chave privada, como já dissemos, é usada para decifrar a mensagem, com isso garantimos que apenas o criador daquela chave a poderia ter criado, viabilizando assim transações seguras como no caso citado acima.

Agora, para fixar este conhecimento, vamos criar um par de chaves assimétricas para acesso aos diretórios do ***git***.

Vamos instalar o ***git*** no ***ubuntu***:

```
sudo apt-get install git
```

a imagem abaixo mostra o comando sendo executado no ***shell***:

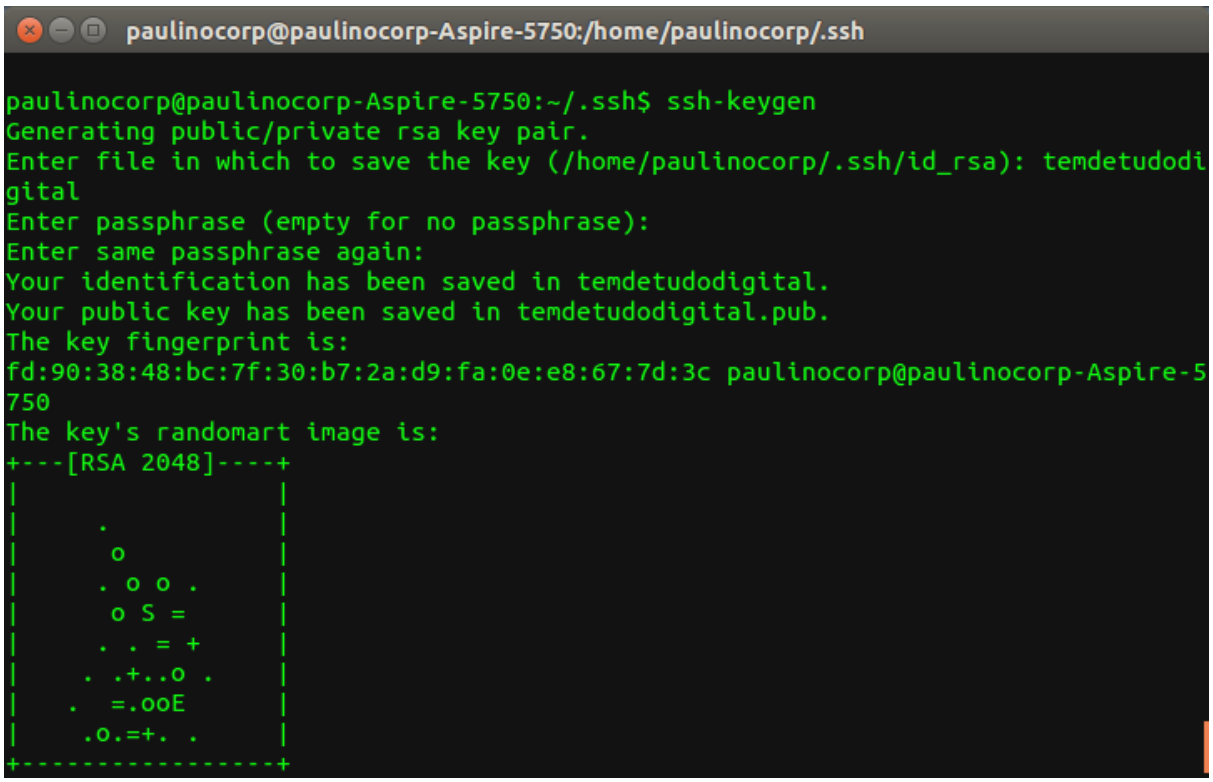


Agora para poder ter acesso seguro ao seu repositório deverá criar na sua pasta pessoal a chave pública e a chave privada que irão autenticar seu acesso com o site github.com e permitira a transferência e a manipulação dos arquivos deste repositório.

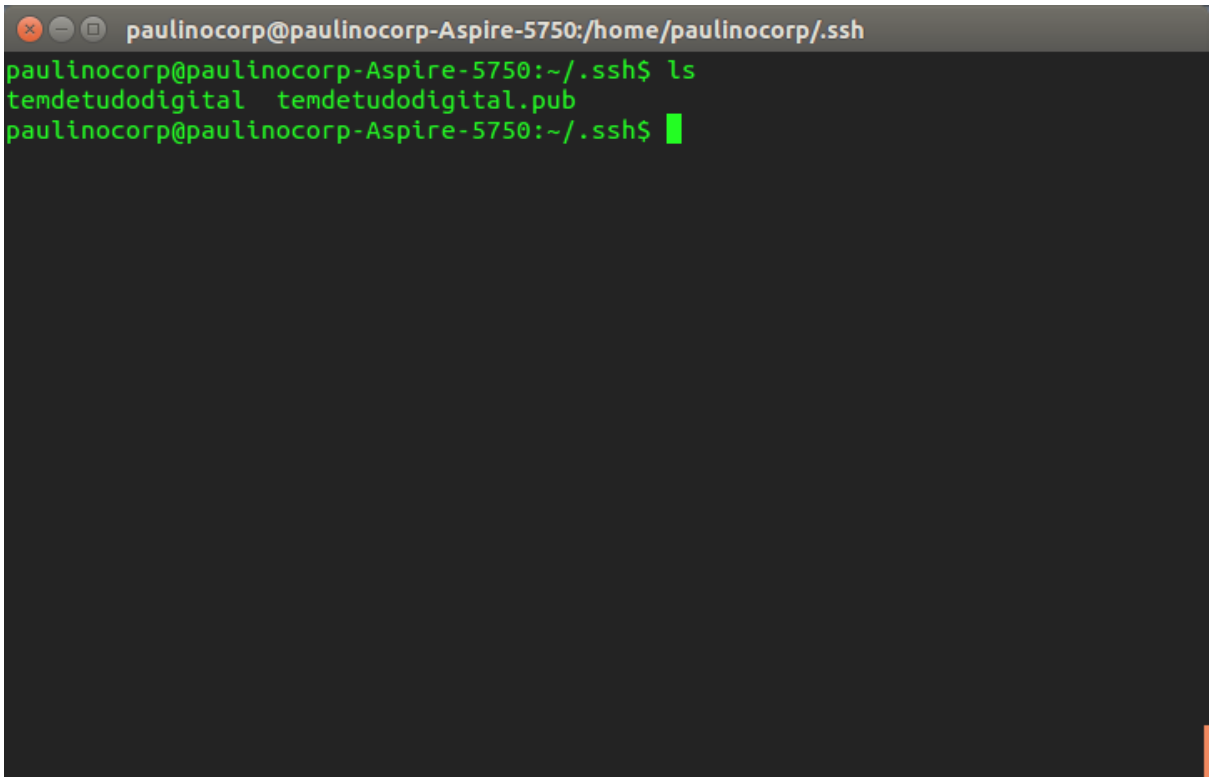
Para criar as chaves usaremos o comando a seguir:

```
ssh-keygen
```

a imagem abaixo mostra esse comando sendo executado:



vamos dar uma olhada no arquivos gerados no diretório padrão do linux para chaves assimétricas:



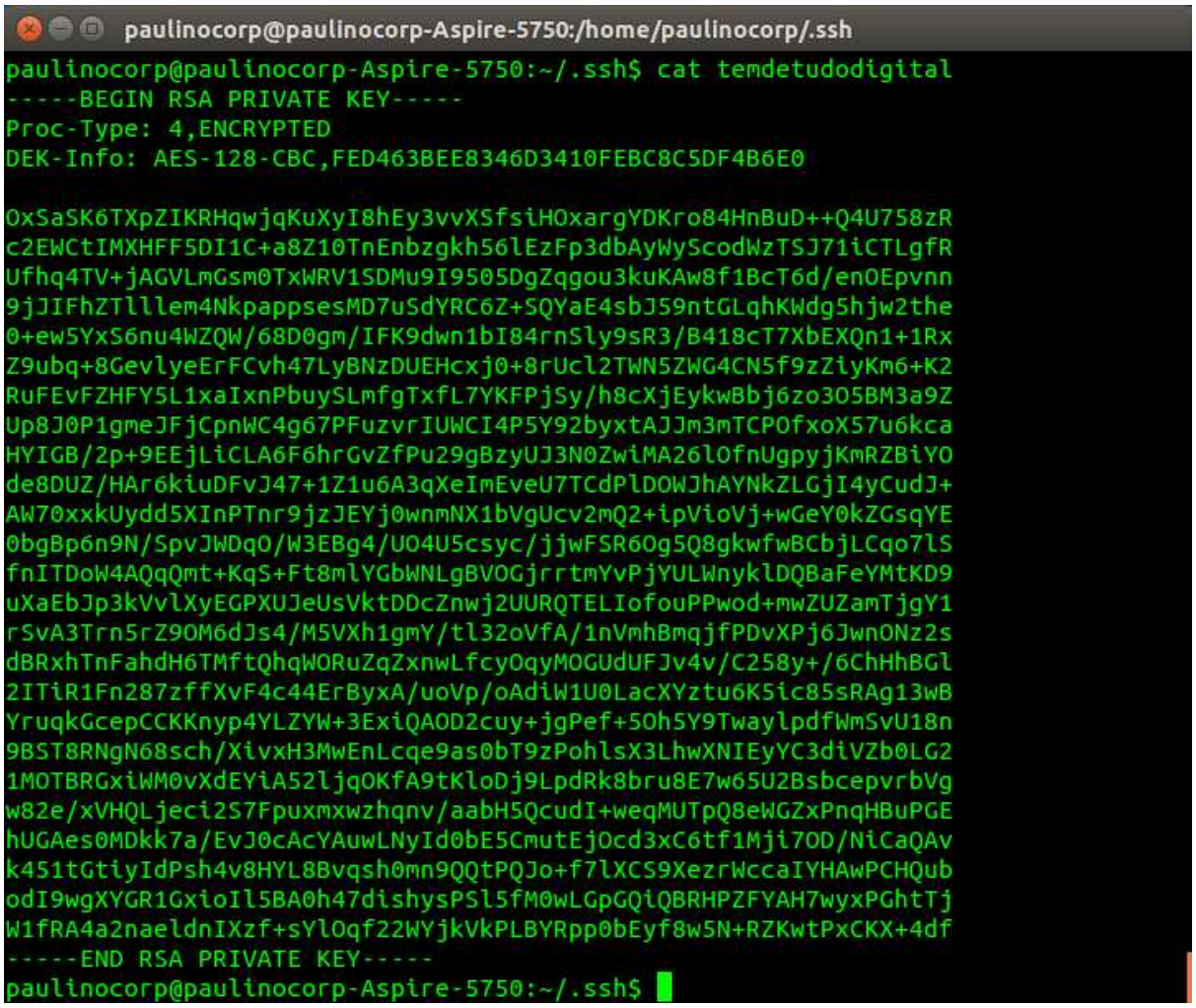
foram criados dois arquivos, um contendo a chave pública, que pode ser distribuda para que tiver o acesso permitido por você e uma chave privada.

vamos ver o conteúdo desses arquivos:

O conteúdo deste arquivo corresponde à chave privada, para ver seu conteúdo usaremos o comando *cat*:

cat temdetudodigital

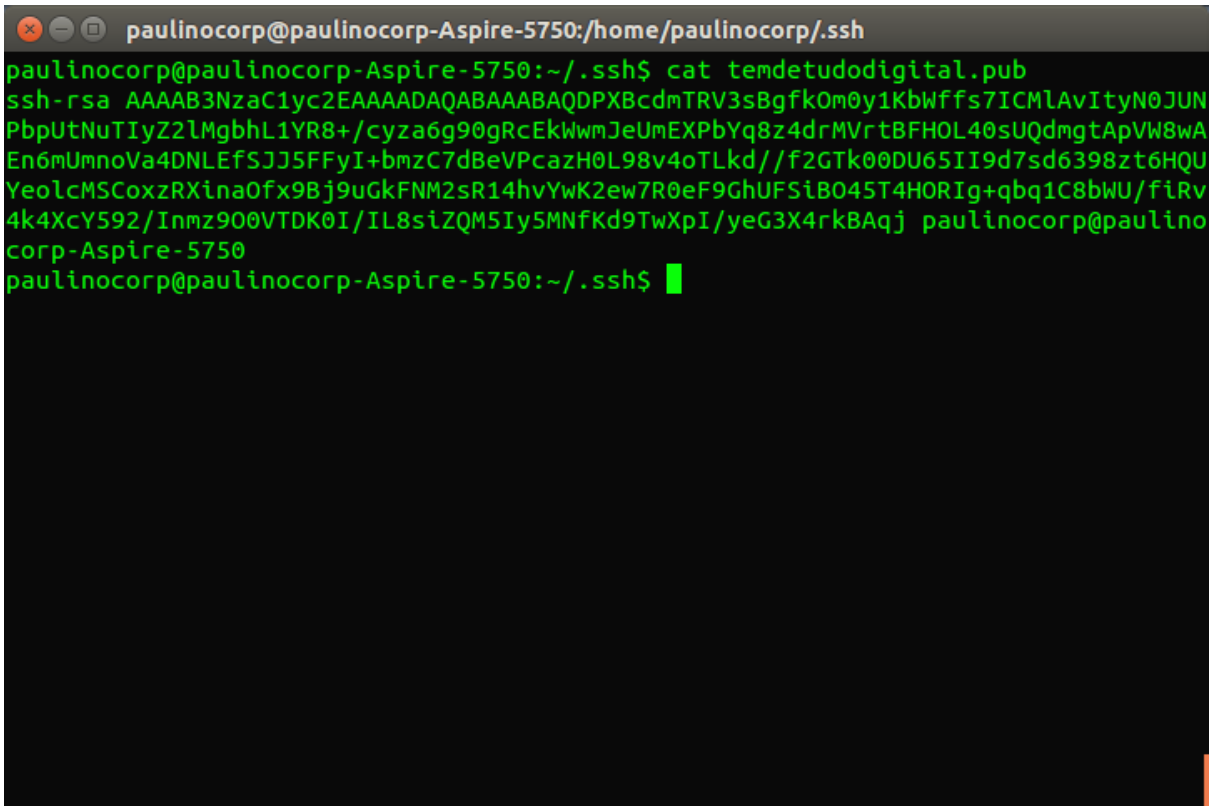
abaixo podemos ver o comando sendo executado:



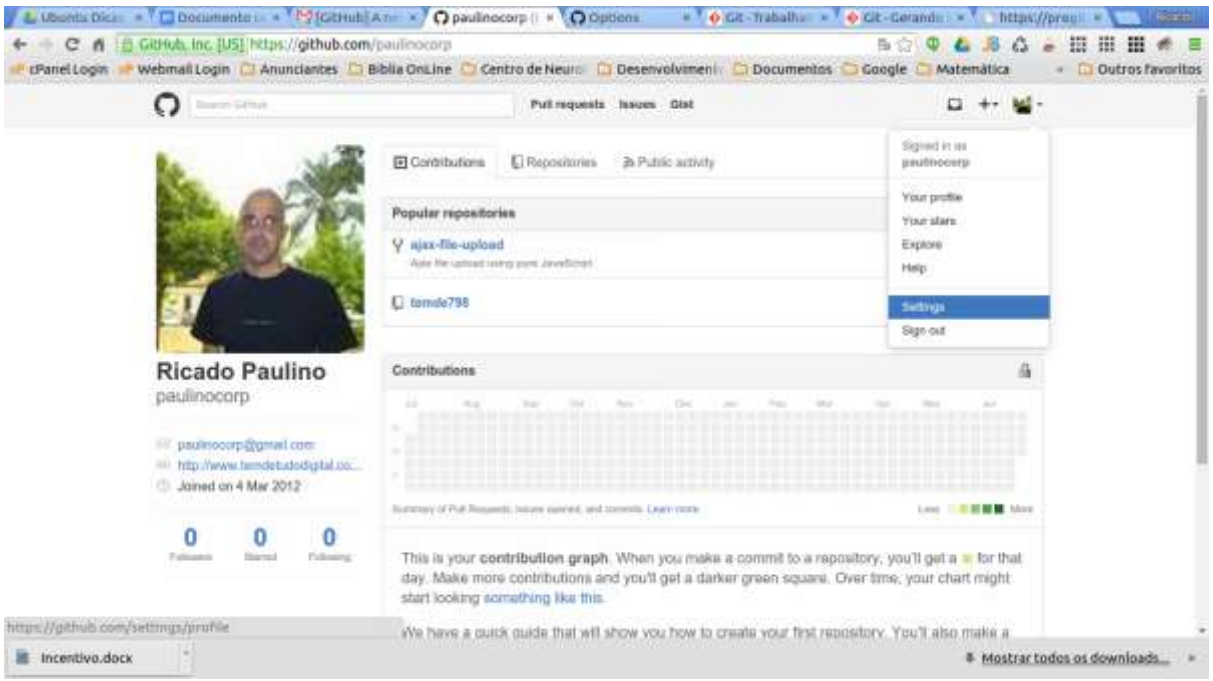
Este arquivo servirá tanto para autenticação da sua máquina no acesso remoto como para decifrar o conteúdo de seus arquivos e torná-los compreensíveis.

Do mesmo modo vamos agora verificar o conteúdo do arquivo *temdetudodigital.pub* que contém a chave pública, para isso utilizaremos o mesmo comando mudando apenas o nome do arquivo:

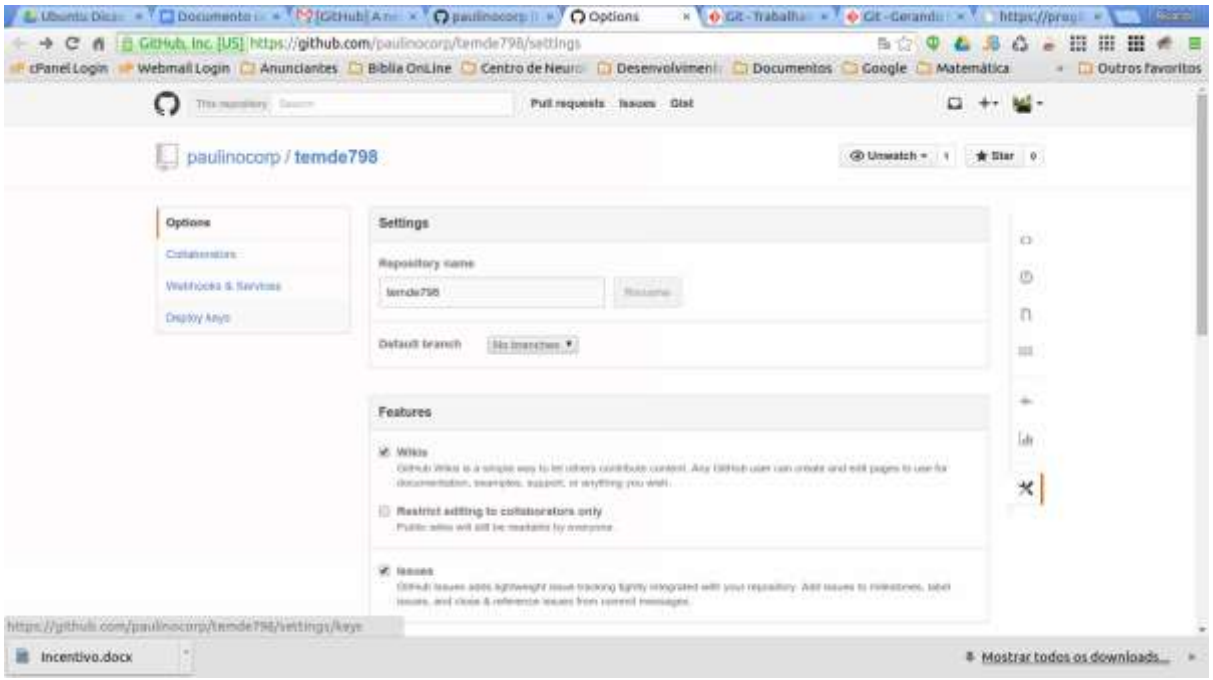
A imagem abaixo mostra o conteúdo do arquivo *temdetudodigital.pub*:



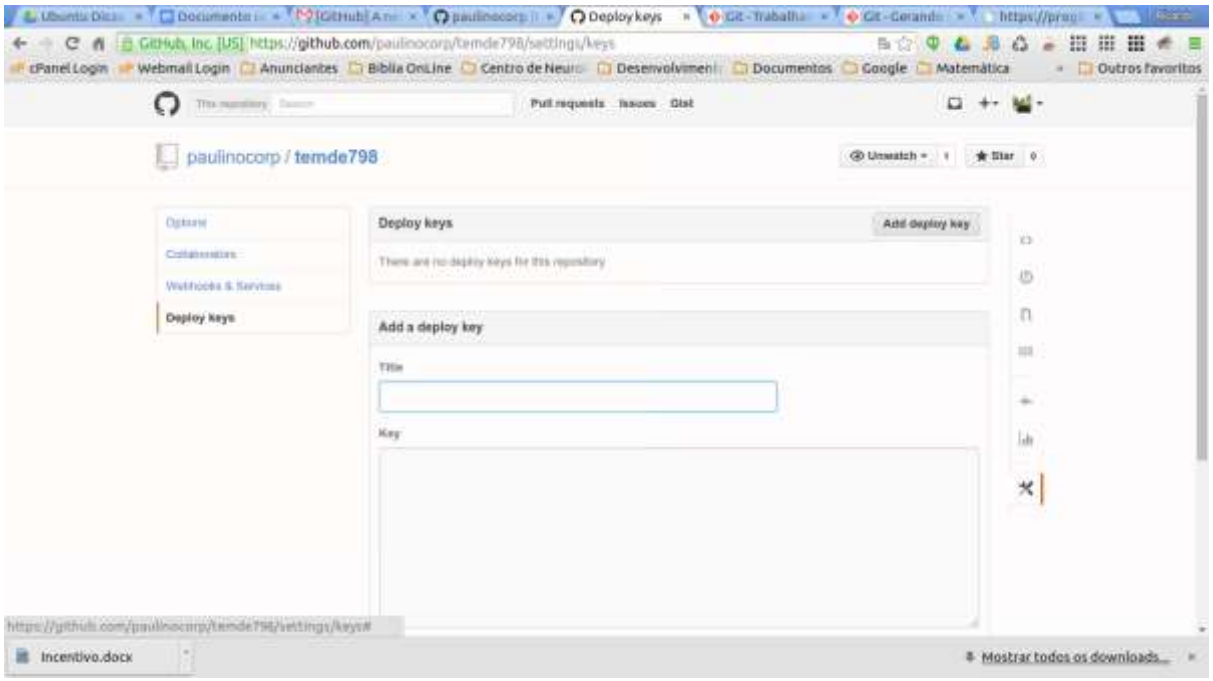
Nesta fase, geradas as chaves deveremos incluir a chave pública no site do *git*, o github.



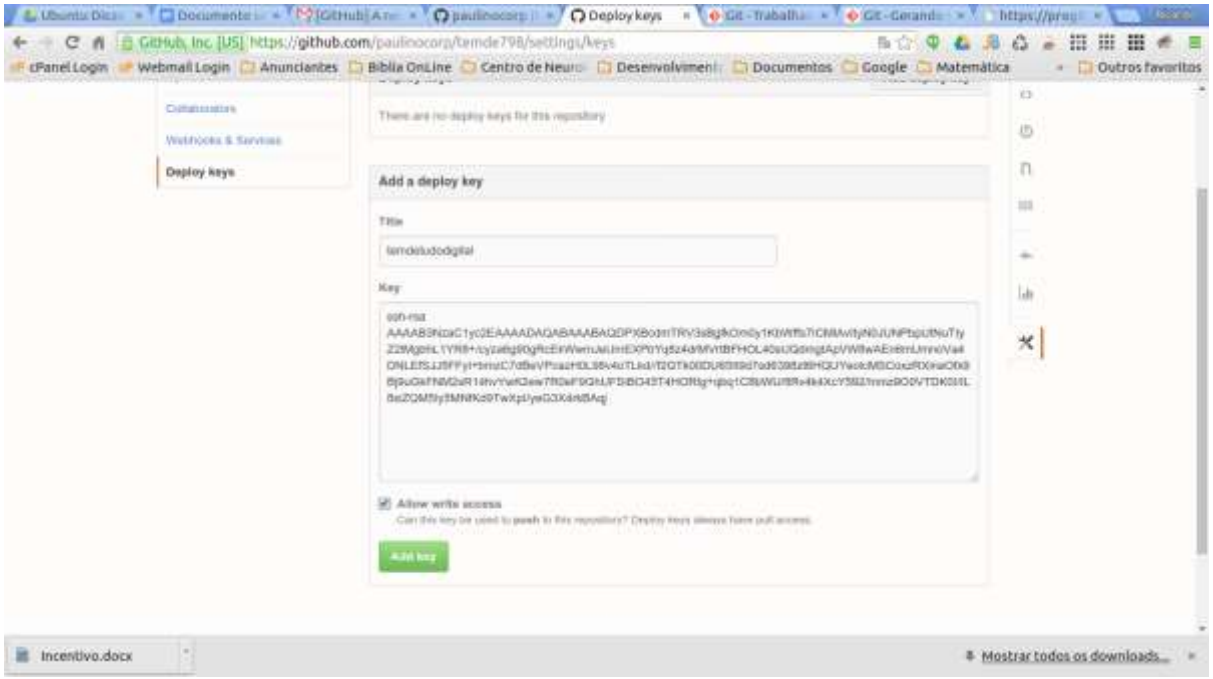
Entrando em *settins encontraremos a opção* ***deploy keys***:



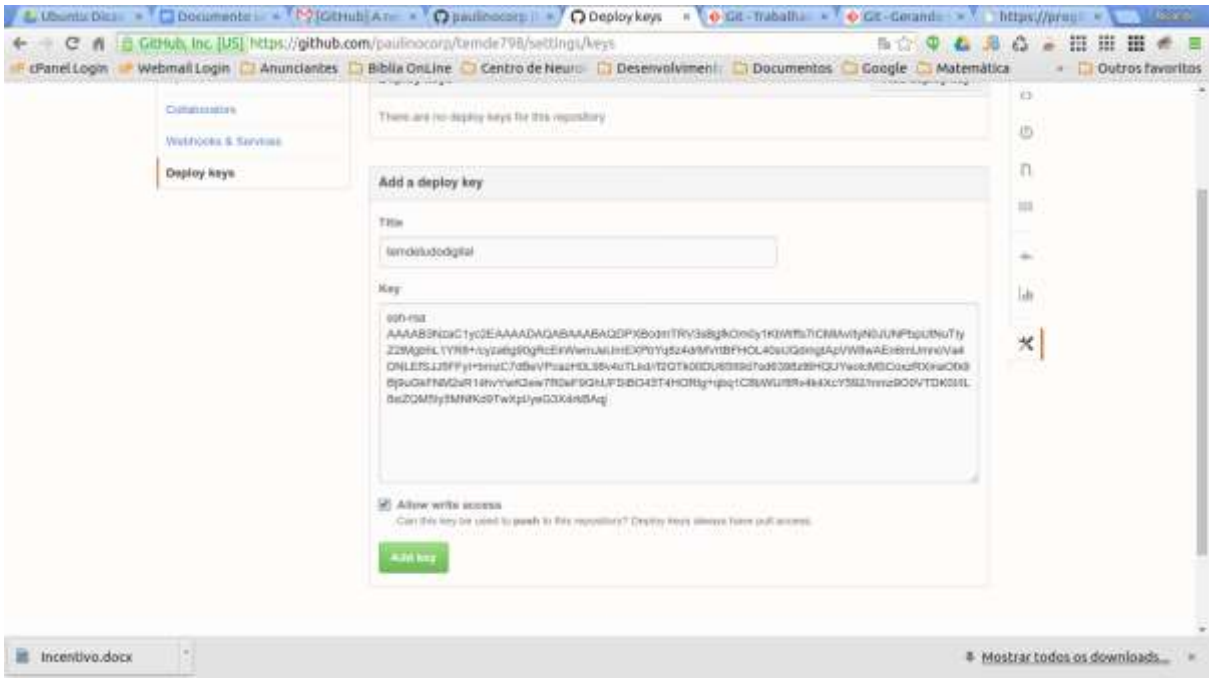
Ao clicarmos com o *mouse* nesta opção aparecerá a tela a seguir:



Devemos clicar no botão ***Add deploy key*** e depois no campo ***title*** colocar o título para o repositório e no campo ***key*** o conteúdo da chave pública.



Veremos na próxima página apresentada pelo site ***github*** que a nossa chave primária foi incluída com sucesso no site.

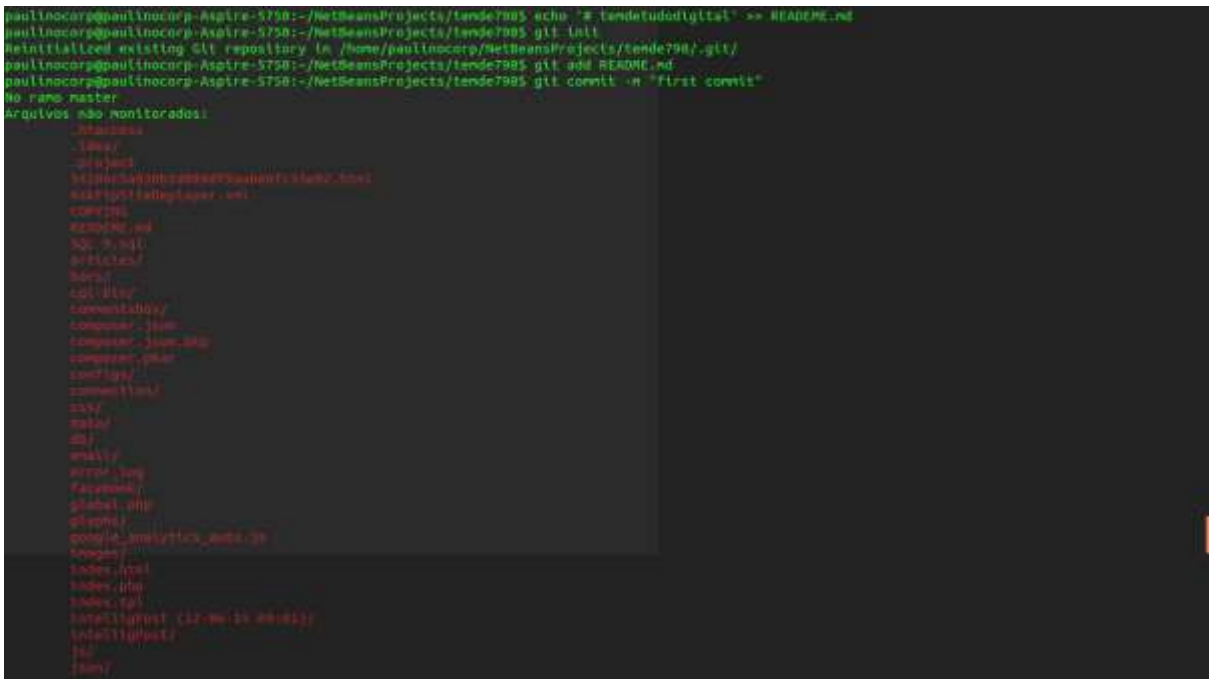


Isto tudo serve para garantir o acesso ao nosso repositório ponde enviálo, modificalo, importá-lo, fazer *clonagem* bem como purgá-lo.

Vamos agora exportar os nossos arquivos de projeto para o repositório:

```
echo "# temdetudodigital" >> README.md
git init
git add README.md
git commit -m "first commit"
```

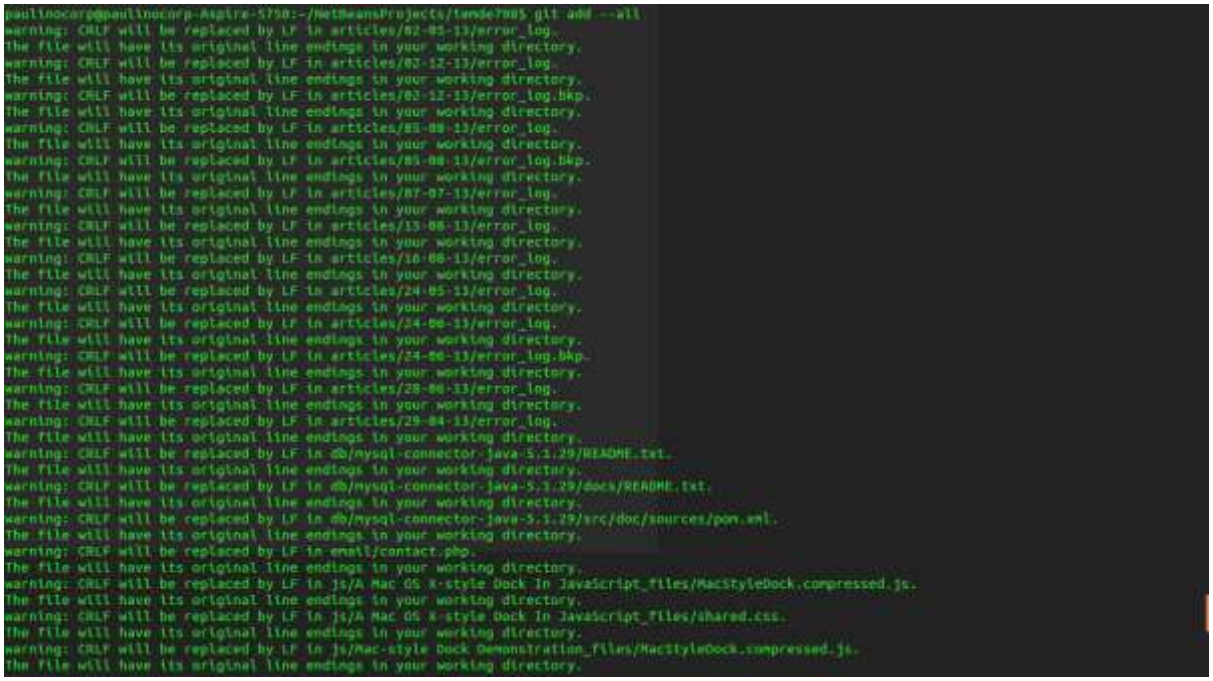
Podemos ver a execução desses comandos na imagem abaixo:



Como o ***git*** avisou que esses arquivos ainda não estava sendo monitorado nos servidores do ***github*** nos os adicionaremos com o comando:

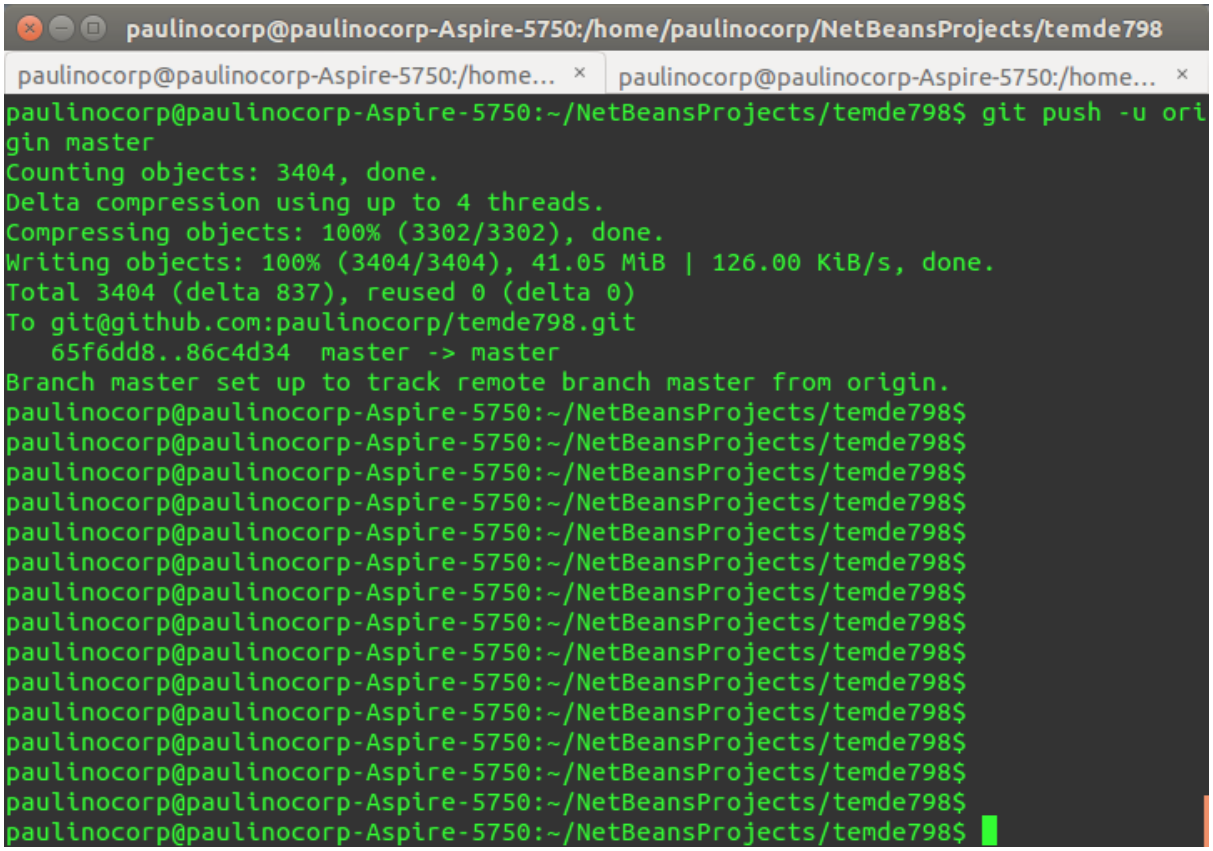
```
git add --all
```

A imagem abaixo mostra os arquivos sendo importados:



Um outro meio de exportar, verificar e atualizar o repositório é através dos seguintes comandos:

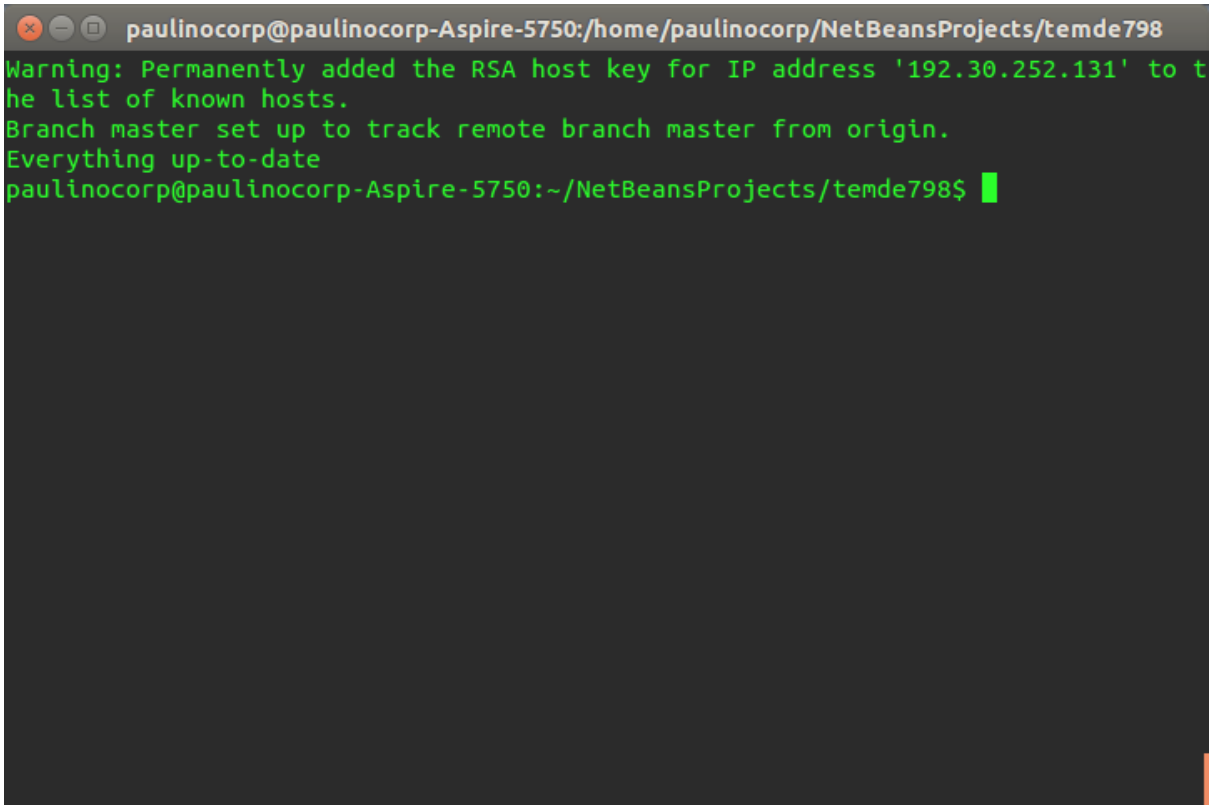
```
git remote add origin git@github.com:paulinocorp/temde798.git
git push -u origin master
```



Atualizamos o repositório com o seguinte comando executado no diretório raíz de nosso projeto conforme a imagem acima:

```
git push -u origin master
```

Se executarmos o comando novamente veremos que o ***git*** retornará a mensagem de que os arquivos estão atualizados.



O ***git*** informa que os dados estão atualizados.

Caros amigos leitores, espero que com este simples artigo tenha atingido o objetivo de lhes trazer algo de útil.