

## Criando um ambiente de teste:

r - Ler  
w - Editar  
x - Executar

- Como root, crie um diretório chamado **documento**, dentro de /home.
  - Qual a permissão desse diretório? → `rw-r--r--`
- Dentro desse diretório, crie um arquivo chamado **nome.txt**, utilizando o **NANO**, dentro desse arquivo digite seu nome completo.
  - Qual a permissão desse arquivo? → `rw-r--r--`
- Ainda como root, crie um usuário chamado ALUNO.
- Faça o login com esse usuário e tente acessar essa pasta, usando o comando **cd**. Depois tente visualizar o conteúdo desse arquivo utilizando o **cat** e o **NANO**.
  - Conseguiu entrar na pasta? Conseguiu visualizar o conteúdo do arquivo das 2 maneiras?  
→ Não
- Agora, faça logout e, como root, mude a permissão desse diretório, apenas para que o usuário criador tenha permissão para ele. Antes disso, vamos aprender como isso pode ser feito.
- Depois que você alterou a permissão do arquivo nome.txt apenas para o dono, o usuário aluno consegue visualizar o conteúdo usando o cat? → Não

## Alterar o permissionamento de um arquivo:

- Mudando com sistema OCTAL

Para se alterar o permissionamento de um arquivo, deve ser utilizado o comando **chmod** de acordo com a sintaxe a seguir:

```
chmod <perm_octal> <nome_do_arquivo_diretório>
```

Supondo que na sua máquina exista um arquivo **file04.txt** e que se queira que aplicar a permissão **4663**. O comando seria:

```
chmod 4663 file04.txt
```

- **Mudando com sistema por grupo de usuários:**

Para alterar as permissões de um arquivo ou diretório, também podemos usar o sistema U/G/O/A (em minúsculo), que definem os níveis para **USUÁRIO** (criador/dono), **GRUPO** (usuários que pertencem ao grupo), **OUTROS** (demais usuários) e **ALL** (todos). Em conjunto com essas 4 letras, podemos usar os operadores =, + e -, para igualar, adicionar ou remover algumas permissões.

Para mostrar como essa combinação é feita, vamos supor que você deseje adicionar permissão de gravação no arquivo teste.txt para um usuário dono. Então o comando a ser digitado será

```
chmod u+w teste.txt
```

O "u" indica que a permissão será dada a um usuário dono. O sinal de adição (+) indica que está sendo adicionada a permissão e "w" indica que a permissão que está sendo dada é de gravação.

Caso você queira dar permissões de leitura e execução ao seu grupo, o comando será:

```
chmod g+rw teste.txt
```

vamos supor que o arquivo o usuário criador tem todas as permissões (rwx) e queremos que os demais usuários tenham também essa permissão, podemos usar o seguinte comando com o operador =:

```
chmod o=u teste.txt
```

O chmod tem uma opção que permite que as permissões sejam atribuídas recursivamente. Isto é, ela muda a permissão de todos os arquivos em todo o diretório e subdiretórios.

Vamos então falar para o chmod mudar a permissão de maneira recursiva (-R) em um diretório:

```
chmod -R o=rwx <diretório>/
```

Algumas distribuições LINUX aceitam fazer mais de um comando u/g/o/a por vez (não é o nosso caso). Por exemplo, falar para o chmod remover a permissão de execução do usuário dono do arquivo (u-x), acrescentar as opções de escrita e execução para o grupo dono (g+wx) e deixar os demais usuários apenas com a permissão de execução (o=x):

```
chmod u-x,g+wx,o=x teste.txt
```

Lembrando que no nosso caso (ubuntu) isso não funciona, mas fica como informação. Se você precisa mudar a permissão dos 3 grupos ao mesmo tempo, use o sistema OCTAL.

## Voltando para a prática

- Agora, voltaremos para o root, crie dentro de **documentos** outro arquivo chamado data.txt que tenha a data atual do sistema.
- Ainda como root, tire a permissão usado o **u/g/o/a** para que os demais usuários não tenham acesso a nada no diretório **documentos**.
- Faça login novamente com o usuário aluno e tente entrar na pasta **documentos** usando o cd.
- Agora tente usar o cat diretamente no arquivo data.txt, consegue visualizar?
- Agora, volte como root e deixe apenas a permissão x para os demais usuários na pasta **documento**, agora use o sistema OCTAL.
- Tente repetir as instruções anteriores, de entrar na pasta e visualizar o conteúdo de data.txt com o cat.
  - Você percebeu que consegue entrar na pasta e visualizar o conteúdo do arquivo nome.txt? Mas tente digitar o comando **ls -l** com o usuário aluno dentro de **documentos**, o que acontece?
  - A permissão X permite você entrar e visualizar o que você tem direito dentro da pasta, mas não consegue visualizar o conteúdo daquela pasta.
- Volte como root, e mude a permissão recursivamente da pasta documentos para que os demais usuários possam ter acesso apenas a leitura (r) nessa pasta.
- Volte como o usuário aluno para a pasta, dê novamente o comando **ls -l** e se conseguir visualizar, tente remover o arquivo **nome.txt**.
- Ainda como o usuário aluno tente executar o seguinte comando abaixo:  
date >> /home/documentos/nome.txt --- *Deu certo?*

## Alterar a propriedade (donos) de um arquivo:

Para se alterar a propriedade de um arquivo, deve ser utilizado o comando chown de acordo com a sintaxe a seguir:

```
chown <nomedousuário> <nomedoarquivo>
```

Supondo que na sua máquina exista um usuário alunogtech e um arquivo de nome file01.txt. Para se alterar a propriedade deste arquivo para o usuário, deve ser utilizado o comando abaixo:

```
chown alunogtech file01.txt
```

Supondo que na sua máquina exista um grupo gtech e um arquivo de nome file02.txt. Para se alterar a propriedade deste arquivo para o grupo (sem alterar o usuário proprietário), deve ser utilizado o comando abaixo:

```
chown :gtech file02.txt
```

Supondo que na sua máquina exista um usuário professorgtech e um grupo professores e um arquivo de nome file03.txt. Para se alterar a propriedade de usuário e grupo deste arquivo, deve ser utilizado o comando abaixo:

```
chown professorgtech:professores file03.txt
```