Aluno: José Rennyson Lima Dias Turma: tec0092

CHMOD

Bem, a gente não usa esse comando muitas vezes, só em casos específicos, mas assim, ele serve basicamente para editar as permissões de pastas ou ou arquivos, para que certos usuários tenham permissões específicas, tipo um arquivo que só pode ser editado por um usuário, e isso obviamente existe para dar mais segurança aos arquivos linux, sem elas, os arquivos ficariam vulneráveis a alterações e ataques mal intencionados, o que poderia resultar em grandes perdas e problemas ao sistema.

Diferenças entre CHMOD e CHOWN

o chmod serve para definir quem é o proprietario do arquivo

chown define quem tem permissão para fazer oq no arquivo(ler, editar, apagar…)

E o linux separa essas permissões em 3 grupos

**user -> proprietario do arquivo.**

**group -> grupo de usuarios quem podem acessar ou editar o arquivo.**

**other -> outros usuarios que tem acesso.**

Alterando permissões das pasta linux

Elas são basicamente “r”, “w” e “d”, que são read, write e execution, que por sua vez significam leitura, escrita e execução, se vc for editar um arquivo normal, então o primeiro simbolo sera “-”, e se for um diretorio será um “d”, e vc usa as letras assim: rwd

rwx: significa que o usuário tem permissão para leitura, edição e execução);

r-x: somente os usuários do grupo do arquivo podem ler ou executar;

r: qualquer usuário pode ler o arquivo, mas não pode editá-lo e nem executá-lo.

Alterando os proprietários de arquivos com chmod

Para alterar quem é o proprietário do arquivo é preciso usar o comando chown, que explicamos lá no início.

A sintaxe básica para esse comando é: chown [owner/group owner] [file name].

Uma opção que funciona tanto com chmod quanto chown é -R, ou recursivo. Com esse comando você pode alterar tanto as permissões quanto os proprietários em uma determinada pasta, e consequentemente, de todos os arquivos naquela pasta.

Porém, é preciso ter muito cuidado com essa opção, pois pode causar falhas e alterar as permissões de todas as suas pastas de Linux de todo o seu sistema, se for usada incorretamente.

Niveis de execução

No comando chmod 777, os diferentes níveis de permissão são divididos de 0 a 7, sendo:

0 = nenhuma permissão;

1 = apenas executar;

2 = apenas gravar;

3 = gravar e executar;

4 = apenas ler;

5 = ler e executar;

6 = ler e gravar; e

7 = ler, gravar e executar.

Logo, o chmod 777 representa o mais alto nível de permissão, em que o usuário é dono do arquivo e pode executar todas as ações acima. Porém, essa sequência pode ser muito perigosa se usada incorretamente, por exemplo em um diretório root (/).

O chmod 777 pode até parecer uma solução rápida para resolver problemas de permissão, mas se usado incorretamente pode deixar todo o sistema vulnerável e correndo perigo de ameaças externas ou alterações incorretas.

Portanto, é preciso saber usar o chmod 777 de forma cautelosa e apenas quando realmente necessário, tomando cuidado para não aplicá-lo a nenhum diretório geral do seu sistema.

Opções adicionais com os comandos chmod e chown

Existem opções adicionais como -R, que mencionamos acima para a sintaxe do comando chmod. Algumas delas são, por exemplo:

–ajuda: Exibe ajuda e sai.

-c, –alterar: informa apenas quando uma alteração é feita;

-f, – silencioso, –quieto: suprime a maior parte das mensagens de erro;

–versão: exibe informações de versão;

-v, –verbose: elabora um diagnóstico para cada arquivo processado.

Já para o chown, algumas opções adicionais de sintaxe são:

-v: reporta o proprietário dos arquivos e as mudanças de proprietário;

-c: reporta somente a mudança de proprietário;

-R: altera o proprietário de todos os arquivos e diretórios recursivamente dentro da mesma hierarquia.

é basicamente isso :)