

Permissões de arquivo no Linux

Descrição do projeto

A equipe de pesquisa da minha organização precisa revisar e atualizar as permissões de arquivos em determinados diretórios dentro do diretório de projetos. As permissões atuais não estão alinhadas com o nível de acesso necessário. A verificação e atualização dessas permissões são essenciais para garantir a segurança do sistema. Para concluir essa tarefa, executei os seguintes passos:

Verificação das permissões de arquivos e diretórios

O código abaixo demonstra como utilizei comandos do Linux para verificar as permissões atualmente atribuídas a um diretório específico no sistema de arquivos.

```
researcher2@3423d4fffc6c:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Jan  5 22:16 .
drwxr-xr-x 3 researcher2 research_team 4096 Jan  5 22:47 ..
-rw--w---- 1 researcher2 research_team  46 Jan  5 22:16 .project_x.txt
drwx--x--- 2 researcher2 research_team 4096 Jan  5 22:16 drafts
-rw-rw-rw- 1 researcher2 research_team  46 Jan  5 22:16 project_k.txt
-rw-r----- 1 researcher2 research_team  46 Jan  5 22:16 project_m.txt
-rw-rw-r-- 1 researcher2 research_team  46 Jan  5 22:16 project_r.txt
-rw-rw-r-- 1 researcher2 research_team  46 Jan  5 22:16 project_t.txt
researcher2@3423d4fffc6c:~/projects$
```

A primeira linha da captura de tela mostra o comando que executei, enquanto as linhas seguintes apresentam a saída gerada. O código exibe todo o conteúdo do diretório de projetos. Utilizei o comando **ls** com a opção **-la** para listar detalhadamente os arquivos, incluindo os ocultos. A saída do comando revela a presença de um diretório chamado **drafts**, um arquivo oculto chamado **.project_x.txt**, e outros cinco arquivos de projeto. A sequência de 10 caracteres na primeira coluna corresponde às permissões atribuídas a cada arquivo ou diretório.

Descrever a sequência de permissões

A sequência de 10 caracteres pode ser analisada para determinar quem tem acesso ao arquivo e quais permissões específicas são concedidas. Abaixo está a explicação dos caracteres e seus significados:

1º caractere: Este caractere pode ser um "d" ou um hífen (-) e indica o tipo de arquivo. Se for "d", significa que é um diretório. Se for um hífen (-), indica um arquivo regular.

2º ao 4º caracteres: Esses três caracteres representam as permissões do **usuário** (proprietário do arquivo) em relação a leitura (r), gravação (w) e execução (x). Se algum desses caracteres for um hífen (-), significa que a permissão correspondente não foi concedida ao usuário.

5º ao 7º caracteres: Esses caracteres indicam as permissões do **grupo** em relação a leitura (r), gravação (w) e execução (x). Assim como no caso do usuário, um hífen (-) significa que a permissão não foi concedida ao grupo.

8º ao 10º caracteres: Esses caracteres representam as permissões para **outros** usuários, ou seja, todos os usuários do sistema que não são nem o proprietário nem fazem parte do grupo. Quando algum desses caracteres é um hífen (-), indica que a permissão correspondente não foi concedida para outros usuários.

As permissões do arquivo **project_t.txt** são **-rw-rw-r--**. Como o primeiro caractere é um hífen (-), isso indica que **project_t.txt** é um arquivo, e não um diretório. O segundo, quinto e oitavo caracteres são **r**, o que significa que o usuário, o grupo e outros têm permissões de leitura. O terceiro e o sexto caracteres são **w**, indicando que apenas o usuário e o grupo têm permissões de gravação. Nenhum dos usuários (nem o proprietário, nem o grupo, nem outros) possui permissões de execução para **project_t.txt**.

Mudar permissões de arquivos

A organização decidiu que usuários externos não devem ter permissões de gravação em nenhum de seus arquivos. Para implementar essa mudança, revisei as permissões dos arquivos que obtive anteriormente. Identifiquei que o arquivo **project_k.txt** precisava ter a permissão de gravação removida para "outros".

O código a seguir mostra como utilizei comandos do Linux para realizar essa alteração:

```
researcher2@3423d4fffc6c:~/projects$ chmod o-w project_k.txt
researcher2@3423d4fffc6c:~/projects$ chmod g-r project_m.txt
researcher2@3423d4fffc6c:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Jan  5 22:16 .
drwxr-xr-x 3 researcher2 research_team 4096 Jan  5 22:47 ..
-rw--w---- 1 researcher2 research_team  46 Jan  5 22:16 .project_x.txt
drwx--x--- 2 researcher2 research_team 4096 Jan  5 22:16 drafts
-rw-rw-r-- 1 researcher2 research_team  46 Jan  5 22:16 project_k.txt
-rw----- 1 researcher2 research_team  46 Jan  5 22:16 project_m.txt
-rw-rw-r-- 1 researcher2 research_team  46 Jan  5 22:16 project_r.txt
-rw-rw-r-- 1 researcher2 research_team  46 Jan  5 22:16 project_t.txt
researcher2@3423d4fffc6c:~/projects$
```

As duas primeiras linhas da captura de tela mostram os comandos que executei, enquanto as linhas seguintes apresentam a saída do segundo comando. O comando **chmod** é utilizado para modificar as permissões de arquivos e diretórios. O primeiro argumento especifica as permissões a serem alteradas, enquanto o segundo argumento indica o arquivo ou diretório afetado. Neste exemplo, removi a permissão de gravação para "outros" no arquivo **project_k.txt**. Em seguida, usei o comando **ls -la** para verificar as alterações realizadas.

Alterar permissões de arquivo em um arquivo oculto

A equipe de pesquisa da minha organização arquivou recentemente o **project_x.txt** e deseja garantir que ninguém tenha permissões de gravação sobre esse arquivo. No entanto, o usuário e o grupo devem manter permissões de leitura.

O código a seguir mostra como utilizei comandos do Linux para modificar as permissões desse arquivo:

```
researcher2@3423d4fffc6c:~/projects$ chmod u-w,g-w,g+r .project_x.txt
researcher2@3423d4fffc6c:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Jan  5 22:16 .
drwxr-xr-x 3 researcher2 research_team 4096 Jan  5 22:47 ..
-r--r----- 1 researcher2 research_team  46 Jan  5 22:16 .project_x.txt
drwx--x--- 2 researcher2 research_team 4096 Jan  5 22:16 drafts
-rw-rw-r-- 1 researcher2 research_team  46 Jan  5 22:16 project_k.txt
-rw----- 1 researcher2 research_team  46 Jan  5 22:16 project_m.txt
-rw-rw-r-- 1 researcher2 research_team  46 Jan  5 22:16 project_r.txt
-rw-rw-r-- 1 researcher2 research_team  46 Jan  5 22:16 project_t.txt
```

As duas primeiras linhas da captura de tela mostram os comandos que executei, enquanto as linhas seguintes apresentam a saída do segundo comando. Eu sei que **.project_x.txt** é um arquivo oculto porque seu nome começa com um ponto (.). Neste exemplo, removi as permissões de gravação tanto para o usuário quanto para o grupo, e adicionei permissões de leitura para o grupo. Removi as permissões de gravação do usuário com **u-w**, depois fiz o mesmo para o grupo com **g-w**, e finalmente, concedi permissões de leitura ao grupo com **g+r**.

Alterar permissões de diretório

Minha organização determinou que apenas o usuário **researcher2** deve ter acesso ao diretório **rascunhos** e seu conteúdo. Isso implica que ninguém, além de **researcher2**, deve ter permissões de execução sobre o diretório.

O código a seguir ilustra como usei comandos do Linux para modificar as permissões:

```
researcher2@3423d4fffc6c:~/projects$ chmod g-x drafts
researcher2@3423d4fffc6c:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Jan  5 22:16 .
drwxr-xr-x 3 researcher2 research_team 4096 Jan  5 22:47 ..
-r--r----- 1 researcher2 research_team  46 Jan  5 22:16 .project_x.txt
drwx----- 2 researcher2 research_team 4096 Jan  5 22:16 drafts
-rw-rw-r-- 1 researcher2 research_team  46 Jan  5 22:16 project_k.txt
-rw----- 1 researcher2 research_team  46 Jan  5 22:16 project_m.txt
-rw-rw-r-- 1 researcher2 research_team  46 Jan  5 22:16 project_r.txt
-rw-rw-r-- 1 researcher2 research_team  46 Jan  5 22:16 project_t.txt
researcher2@3423d4fffc6c:~/projects$
```

A saída exibe uma lista de permissões para vários arquivos e diretórios. A primeira linha mostra o diretório atual (**projects**), enquanto a segunda linha exibe o diretório pai (**home**). A terceira linha corresponde a um arquivo regular chamado **.project_x.txt**. A quarta linha se refere ao

diretório **drafts**, que possui permissões restritas. Nessa linha, é possível observar que apenas o usuário **researcher2** tem permissões de execução. Como o grupo já possuía permissões de execução, utilizei o comando **chmod** para removê-las. O usuário **researcher2** já tinha permissões de execução, então não foi necessário concedê-las novamente.

Resumo

Modifiquei diversas permissões para garantir que correspondessem ao nível de acesso desejado pela minha organização para arquivos e diretórios dentro do diretório de projetos. O primeiro passo foi utilizar o comando **ls -la** para revisar as permissões atuais do diretório, o que orientou minhas decisões nas etapas subsequentes. Em seguida, usei o comando **chmod** repetidamente para ajustar as permissões em arquivos e diretórios conforme necessário.