Permissões de arquivos no Linux

Descrição do projeto

A equipe de pesquisa da minha organização precisa atualizar as permissões de arquivos para certos arquivos e diretórios dentro do diretório `projects`. As permissões atuais não refletem o nível de autorização que deveria ser concedido. Verificar e atualizar essas permissões ajudará a manter o sistema seguro. Para concluir essa tarefa, realizei as seguintes etapas:

Verificar detalhes de arquivos e diretórios

O código a seguir demonstra como usei comandos Linux para determinar as permissões existentes definidas para um diretório específico no sistema de arquivos.

A primeira linha da captura de tela exibe o comando que digitei, e as outras linhas exibem a saída. O código lista todos os conteúdos do diretório `projects`. Usei o comando `ls` com a opção `-la` para exibir uma listagem detalhada do conteúdo do diretório, incluindo arquivos ocultos.

A saída do meu comando indica que há um diretório chamado `drafts`, um arquivo oculto chamado `.project_v.txt` e quatro outros arquivos de projeto. A string de 10 caracteres na primeira coluna representa as permissões definidas para cada arquivo ou diretório.

Descrever a string de permissões

A string de 10 caracteres pode ser decomposta para determinar quem está autorizado a acessar o arquivo e quais permissões específicas possui. Os caracteres e seus significados são os seguintes:

- **1º caractere**: Este caractere é um `d` ou um hífen (`-`) e indica o tipo do arquivo. Se for `d`, é um diretório. Se for `-`, é um arquivo regular.

- **2º ao 4º caracteres**: Indicam as permissões de leitura (`r`), escrita (`w`) e execução (`x`) para o **usuário**. Se algum desses caracteres for um hífen (`-`), significa que a permissão correspondente **não** foi concedida ao usuário.

- **5º ao 7º caracteres**: Indicam as permissões de leitura (`r`), escrita (`w`) e execução (`x`) para o **grupo**.

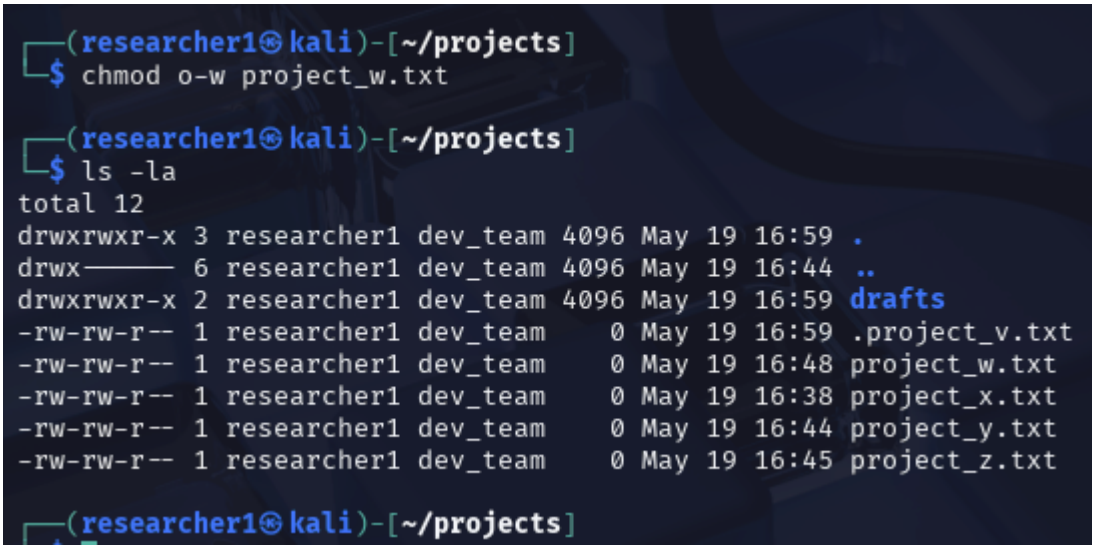
- **8º ao 10º caracteres**: Indicam as permissões de leitura (`r`), escrita (`w`) e execução (`x`) para **outros** (todos os demais usuários do sistema que não são o dono nem estão no grupo).

Exemplo:
As permissões do arquivo `project_w.txt` são `-rw-rw-r--`.
Como o primeiro caractere é um hífen (`-`), isso indica que `project_w.txt` é um arquivo, não um diretório.
O segundo, quinto e oitavo caracteres são `r`, o que indica que o **usuário**, **grupo** e **outros** têm permissão de leitura.
O terceiro e sexto caracteres são `w`, indicando que **usuário** e **grupo** têm permissão de escrita.
Ninguém tem permissão de execução para esse arquivo.

Alterar permissões de arquivos

A organização determinou que **outros usuários (other)** não devem ter acesso de escrita a nenhum de seus arquivos. Para atender a isso, consultei as permissões retornadas anteriormente e determinei que o arquivo `project_w.txt` precisava ter a permissão de escrita removida para **outros**.

O código a seguir demonstra como usei comandos Linux para fazer isso:

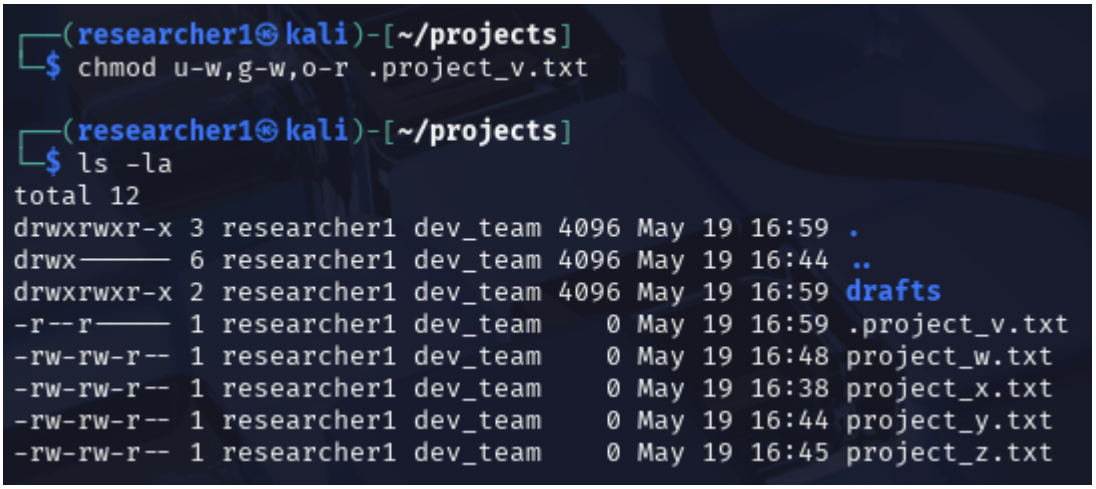


As duas primeiras linhas da captura mostram os comandos digitados, e as outras linhas mostram a saída do segundo comando. O comando `chmod` altera as permissões de arquivos e diretórios. O primeiro argumento indica quais permissões devem ser alteradas, e o segundo argumento especifica o arquivo ou diretório. Neste exemplo, removi a permissão de escrita para "outros" no arquivo `project_w.txt`. Depois disso, usei `ls -la` para revisar as alterações feitas.

Alterar permissões em um arquivo oculto

A equipe de pesquisa da minha organização recentemente arquivou o `project_v.txt`. Eles não querem que ninguém tenha acesso de escrita a esse projeto, mas o **usuário** e o **grupo** devem ter acesso de leitura.

O código a seguir mostra como usei comandos Linux para alterar essas permissões:



As duas primeiras linhas da captura de tela mostram os comandos digitados, e as outras linhas mostram a saída do segundo comando.

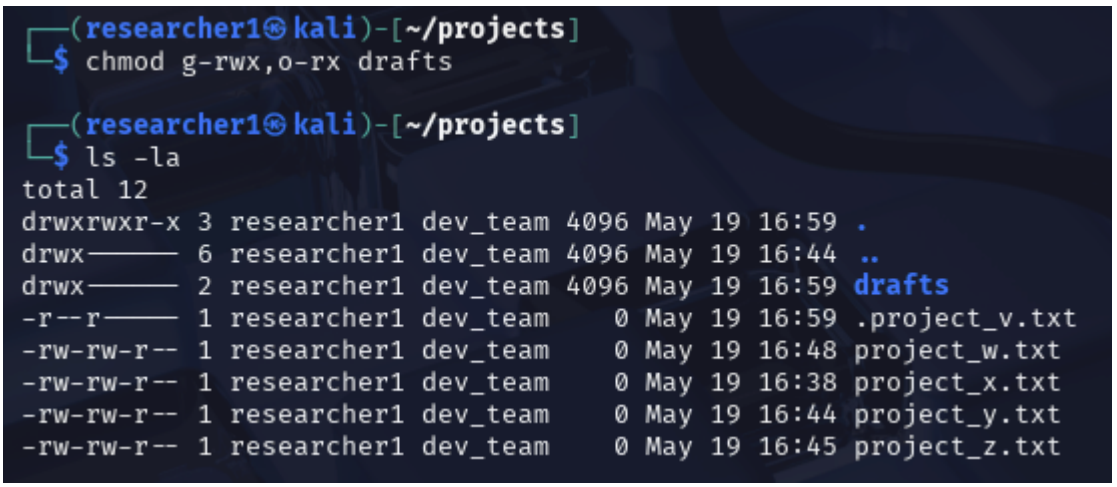
Sei que `.project_v.txt` é um arquivo oculto porque seu nome começa com ponto (`.`).

Neste exemplo, removi a permissão de escrita do usuário com `u-w`. Depois, removi a permissão de escrita do grupo com `g-w` e removi a permissão de leitura aos outros com `o-r`.

Alterar permissões de diretório

Minha organização deseja que apenas o usuário `researcher1` tenha acesso ao diretório `drafts` e ao seu conteúdo. Isso significa que ninguém, além de `researcher1`, deve ter permissão de execução.

O código a seguir mostra como usei comandos Linux para alterar essas permissões:



A saída exibe a listagem de permissões de vários arquivos e diretórios. A linha 1 indica o diretório atual (`projects`), e a linha 2 indica o diretório pai (`home`). A linha 3 é o diretório `drafts`, com permissões restritas. Nela, pode-se ver que apenas o usuário `researcher1` tem permissão de execução.

Anteriormente, o grupo e outros também tinham essa permissão, então usei o comando `chmod` para removê-la. O usuário `researcher1` já tinha permissão de execução, portanto, ela não precisou ser adicionada novamente.

Resumo

Alterei várias permissões para corresponder ao nível de autorização desejado pela minha organização para arquivos e diretórios dentro do diretório `projects`. O primeiro passo foi usar o comando `ls -la` para verificar as permissões. Essas informações guiaram minhas decisões nos passos seguintes. Em seguida, usei o comando `chmod` diversas vezes para alterar as permissões dos arquivos e diretórios.