

Introd. Software Livre

Permissões de Arquivos

Usuários

- É possível alternar facilmente entre usuários no linux
- O comando su (switch user) permite trocar de usuário sem sair do usuário atual
- Outro comando importante é o comando w
- Este comando exibe os usuários atualmente logados no sistema

Permissões

- Ao listar arquivos em uma pasta pode-se ver as permissões de cada um:
- As permissões são:
 - W - Write
 - R – Read
 - X – Execute
- Para cada arquivo as permissões são dadas ao dono, ao grupo e aos outros.
- Assim as permissões aparecem na forma:
- **-wrxwrxwrx** – dono grupo outros
- O primeiro caracter pode ser um **D** o que indica que este arquivo é na verdade um diretório

Permissões

- Há outros tipos de arquivos:
- **b** – arquivo especial de blocos Ex.: /dev/sda
- **C** – arquivo especial de caracter Ex.: /dev/tty
- **p** – canal (criado para comunicação entre processos)
- **S** – socket (criado para comunicação entre máquinas na rede)

Permissões

- Mudando permissões
- `chmod` – change mode
- `chmod [modo] arquivo`
- Onde modo pode estar no formato numérico ou textual
- Textual
 - `[ugo][+ -=][rwx]`
- Numérico
 - `000 666 777`

Permissões

- Quem
 - u para dono
 - g para o grupo
 - o para outros
- Operadores
 - + Adicionar
 - - Retirar
 - = Igualar
- Ex.:
 - $u=rw$ – Dono tem permissão de leitura e escrita
 - $u+x$ – Adicionar permissão de execução para o dono

Permissões

- Modo numérico
- Mais comumente usado
 - R = 4
 - W = 2
 - X = 1
- Para combinações basta somar os valores
 - $RWX = 4 + 2 + 1 = 7$
 - $R-X = 4 + 1 = 5$
 - $RW- = 4 + 2 = 6$
- chmod pode então ser usado com os valores para u/g/o de uma vez só. Ex.:
 - `chmod 755 arquivo.sh` – dono tem privilegio 7 / grupo 5 / outros 5

Permissão	Valor
---	0
--X	1
-W-	2
-WX	3
r--	4
r-X	5
rw-	6
rwX	7

Permissões

- `chown` muda o dono e o grupo de um arquivo
- `chown [dono]:[grupo] arquivo`
- `chgrp` também pode ser usado para mudar apenas o grupo do arquivo