



Capítulo 4: Administração de Usuários

Type	Leitura
Materials	<u>Capítulo 4.pdf</u>
Reviewed	<input type="checkbox"/>

SUPER USUÁRIO

- `Su [opções] [usuario]`
- As informações dos usuários estão dentro de `/etc/passwd` e os do grupo é em `/etc/group`
- As senhas estão dentro de `/etc/shadow` e as senhas dos grupos é no `/etc/gshadow`

ADMINISTRAÇÃO DE USUÁRIOS

1. IDs de Usuários e Grupos no Sistema

- **IDs de Grupo de Sistema:** Variam conforme a distribuição, geralmente entre `0-999` ou `100-999`.
- **IDs de Usuários Normais:** Começam geralmente a partir de `1000`.

2. Comandos Relacionados a Usuários e Grupos

- `users`: Lista os usuários atualmente logados.
- `groups`: Exibe os grupos dos usuários.

- `id`: Mostra o ID do usuário, ID do grupo principal, e os IDs dos grupos secundários aos quais o usuário pertence.

3. Arquivo `/etc/passwd`

- Contém informações sobre os usuários do sistema.

```
cat /etc/passwd
```

4. Arquivo `/etc/group`

- O arquivo `/etc/group` armazena informações sobre os grupos no sistema

```
cat /etc/group
```

Estrutura de uma Linha no `/etc/passwd`

```
root:x:0:0:root:/root:/bin/bash
```

Descrição dos Campos:

1. **Campo do Usuário:** Nome do usuário (ex.: `root`).
2. **Senha:** `x` indica que a senha está armazenada em outro arquivo (`/etc/shadow`).
3. **UID (User ID):** `0` para o usuário root. Usuários normais começam geralmente a partir de `1000`.
4. **GID (Group ID):** `0` para o grupo root. Corresponde ao ID do grupo principal do usuário.

5. **GECOS (Comentários):** Informações adicionais como nome completo, número de sala, etc. (ex.: `root`).
6. **Home Directory:** Diretório inicial do usuário (ex.: `/root`).
7. **Shell:** Shell padrão do usuário (ex.: `/bin/bash`).

Estado das Contas

- `!` ou `!` no campo de senha (no arquivo `/etc/shadow`):
 - `!` indica que a conta está bloqueada.
 - `!` indica que a conta está configurada sem senha.

```
ubuntu:$1$BJxvA4uP$Lap0ybTdVlF6cvj1PMBGF:12060:0:99999:7:::
```

- ▶ **Nome do usuário:** É o nome de login do usuário, igual ao referido no arquivo `/etc/passwd`.
- ▶ **Senha criptografada:** geralmente utiliza-se um algoritmo Hash (MD5 ou SHA-512)
- ▶ **Última mudança de senha:** este número representa o número de dias decorridos entre 1 de janeiro de 1970 e a última alteração da senha.
- ▶ **Número de dias para que a mudança de senha seja permitida:** tipicamente este número é zero, permitindo que o usuário mude sua senha quando desejar.
- ▶ **Número de dias após o qual a senha deve ser alterada:** caso a alteração da senha não seja forçada, este número será 99999.
- ▶ **Número de dias antes da expiração da senha no qual o usuário será avisado:** tipicamente o usuário é avisado com uma semana de antecedência.
- ▶ **Número de dias entre a expiração da senha e a desativação da conta:** caso não se queira desativação automática da conta, este campo é deixado em branco ou com o valor 1.
- ▶ **Dia da desativação da conta:** dias decorridos entre 1 de janeiro de 1970 e a data em que a conta será desativada. Um valor em branco (ou 1) neste campo suspende desativação automática.
- ▶ **Campo reservado:** para uso futuro.

getent

- buscar informações da senha sobre o usuário "joao"

```
getent passwd joao
```

Login

- usuário se autentique e acesse o sistema com suas credenciais (nome de usuário e senha).

```
login usuario
```

1. lastlog

- **Descrição:** Mostra o último login dos usuários cadastrados no sistema.
- **Uso:**

```
lastlog [opções]
```

- **Opções:**
 - `u [nome]` : Mostra somente detalhes sobre o usuário especificado.

2. last

- **Descrição:** Mostra uma listagem de entrada e saída de usuários no sistema.
- **Uso:**

```
bashCopiar código  
last [opções]
```

- **Opções:**
 - `R` : Não mostra o campo HostName.
 - `a` : Mostra o hostname na última coluna. É útil se combinada com a opção `d`.
 - `d` : Usa o DNS para resolver o IP de sistemas remotos para nomes DNS.

3. logname

- **Descrição:** Mostra o login (username) do usuário atual.
- **Uso:**

```
logname
```

4. chfn

- **Descrição:** Muda os dados inseridos durante o cadastro do novo usuário.
- **Uso:**

```
chfn [opções] [usuário]
```

- **Onde:**
 - `usuário` : Nome do usuário para o qual os dados serão alterados.
- **Opções:**
 - `f [nome]` : Adiciona/altera o nome completo do usuário.
 - `r [nome]` : Adiciona/altera o número da sala do usuário.
 - `w [tel]` : Adiciona/altera o telefone de trabalho do usuário.
 - `h [tel]` : Adiciona/altera o telefone residencial do usuário.
 - `o [outros]` : Adiciona/altera outros dados do usuário.

Adicionando Usuários

- `adduser [opções] [usuário/grupo]`

Adicionar usuário:

```
adduser: Adiciona um novo usuário
```

Adicionar grupo:

```
addgroup: Adiciona um novo grupo com as opções especifica
```

Adicionar usuário a um grupo:

`adduser -a usuário grupo:` Adiciona um usuário existente a

Comandos para verificar usuários:

Entra/muda de usuario: `su nome_do_usuario:` Entra no shell
Sair de usuario: `logout` : verifica o id do usuário
`id` : verifica o id do usuário
`cat /etc/passwd | cut -d: -f1:` ou `cat /etc/group` Exib

OPÇÕES COMUNS:

`-- hfn alfredo:` para alterar as informações de contato do

EXEMPLOS

- Para ver os usuários do sistema, digite `users`
- para logar com um usuário `su - alfredo`

#Adicionar um novo usuário:

`adduser novo_usuario`

#Adicionar um usuário existente a um grupo existente:

`adduser usuario_existente grupo_existente`

#Desativa verificações rigorosas de validade de nomes.

Exemplo:

`adduser --allow-bad-names novo_usuario`

#Define um comentário para a nova entrada gerada.

Exemplo:

```
adduser --comment "Novo usuário para departamento de vendas"
```

--gid:

Define o GID para o novo grupo ou grupo primário do novo usuário

Exemplo:

```
adduser --gid 1000 novo_usuario
```

Removendo Usuários

#Remover usuário do grupo

deluser ou userdel

Remover um usuário sem excluir o diretório home:

```
deluser --quiet joao
```

Remover um usuário e excluir o diretório home:

```
deluser --remove-home pedro
```

Remover um usuário e fazer backup dos arquivos antes

```
deluser --backup --backup-to /backup pedro
```

Remover um usuário do sistema:

```
deluser --system maria
```

Remover um grupo:

```
delgroup --verbose desenvolvedores
```

Remover um grupo apenas se estiver vazio:

```
delgroup --only-if-empty administradores
```

Remover um usuário de um grupo específico:

```
deluser --quiet ana administradores
```

Logando em outro Grupos

- para mudar o grupo principal da sessão atual para o grupo especificado

```
newgrp nome_do_grupo
```

Adicionando Grupos

- Esse diretório está em 'cat /etc/group' e tem as senhas no 'cat /etc/gshadow'
 - A divisão do diretório '/etc/group' se baseia em:
 - Ex: vboxusers:x:136:
 - 1. nome do grupo
 - 2. a senha está em outro arquivo

3. identificação do grupo (gid)

```
#comando que faz você se pertencer a um grupo existente por uid  
groupadd nome_do_grupo
```

```
#comando que faz você se pertencer a um grupo existente por uid  
newgrp  
exemplo: newgrp - strigus
```

-f (Indicador force):

```
#Este comando cria um novo grupo chamado "mygroup", mesmo se o grupo já existir  
groupadd -f mygroup
```

Removendo Grupos

- você precisará remover os usuários do grupo ou transferi-los para outro grupo antes de tentar remover o grupo novamente.

```
delgroup ou grupdel
```

--only-if-empty: Remove um grupo apenas se estiver vazio.
Exemplo:

```
deluser --group --only-if-empty grupo
```

--remove-home: Remove o diretório home do usuário quando o usuário for removido.
Exemplo:

```
deluser --remove-home usuário
```

--verbose: Fornece informações detalhadas durante a execução.
Exemplo:

```
deluser --verbose usuário
```

Gerenciando senhas de Usuários

- No usuário normal, ele pede a senha atual pra modificar. No root não
- Quando é usuario, ele pede pra que a senha seja um pouco melhor. Já como root não precisa

```
#Para mudar a senha  
passwd nome_usuario
```

```
#Dá uma senha para um grupo  
gpasswd nome_do_grupo
```

```
#Caso o usuário esqueça a senha (-d de delete e -e de expirar  
passwd -de
```

```
#Deleta a senha para a conta.
```

```
passwd -d usuário
```

```
#Expira a senha para a conta. Agora vai pedir uma nova senha  
passwd -e usuário
```

```
#muda a senha somente se estiver expirada.  
passwd -k usuário
```

```
#tempo de conta inativa  
passwd -i 30
```

```
# A senha fica inativa após a expiração
sudo chage -I 0 alfredo

#trava a conta
passwd -l usuário

# O usuário deve aguardar 3 dias para alterar a senha após um
sudo chage -m 3 alfredo

# A senha deve ser alterada a cada 60 dias
sudo chage -M 60 alfredo

# Notificar sobre a expiração da senha com 5 dias de antecedência
sudo chage -W 5 alfredo

#Este comando destrava a conta indicada.
passwd -u usuário

#Após terminar o prazo, a senha deverá ser modificada

passwd -x MAX_DIAS

Exemplo:

passwd -x 90
```

/etc/login.defs

- O arquivo `/etc/login.defs` é um arquivo de configuração em sistemas Linux que define as políticas e parâmetros para o comportamento de logins e gerenciamento de contas de usuários. Esse arquivo é utilizado por diversos

comandos relacionados à gestão de usuários, como `useradd`, `userdel`, `usermod`, e outros.

Principais Diretrizes e Configurações do `/etc/login.defs`

1. Definição de UID e GID:

- **UID_MIN** e **UID_MAX**: Definem o intervalo de IDs de usuários para contas de usuário normais (não-sistema).
- **GID_MIN** e **GID_MAX**: Definem o intervalo de IDs de grupo para grupos normais (não-sistema).

2. Configurações de Expiração de Senha:

- **PASS_MAX_DAYS**: Número máximo de dias que uma senha pode ser usada antes de ser necessário alterá-la.
- **PASS_MIN_DAYS**: Número mínimo de dias que uma senha deve ser mantida antes que possa ser alterada.
- **PASS_WARN_AGE**: Número de dias para alertar o usuário antes da expiração da senha.

3. Configurações de Diretório Home:

- **CREATE_HOME**: Define se um diretório home deve ser criado automaticamente ao adicionar um novo usuário.
- **UMASK**: Define a máscara padrão para a criação de novos arquivos e diretórios.

4. Segurança de Senhas:

- **ENCRYPT_METHOD**: Especifica o método de criptografia para senhas, como SHA512, MD5, etc.
- **FAIL_DELAY**: Define o atraso em segundos após uma falha de login, como uma medida de segurança contra ataques de força bruta.

5. Políticas de Shell e Login:

- **LOGIN_RETRIES**: Define o número de tentativas de login permitidas antes de uma ação ser tomada.
- **LOGIN_TIMEOUT**: Tempo em segundos que o sistema espera para um login ser completado antes de ser cancelado.

- **MAIL_DIR:** Diretório onde as caixas de correio dos usuários estão localizadas.

6. Configurações Adicionais:

- **USERGROUPS_ENAB:** Define se cada usuário terá seu próprio grupo com o mesmo nome.
- **CHFN_RESTRICT:** Restringe quais campos de informações do usuário podem ser alterados pelo comando `chfn`.

Intervalo de UIDs para usuários normais

UID_MIN	1000
UID_MAX	60000

Intervalo de GIDs para grupos normais

GID_MIN	1000
GID_MAX	60000

Políticas de senha

PASS_MAX_DAYS	90
PASS_MIN_DAYS	0
PASS_WARN_AGE	7

Método de criptografia de senhas

ENCRYPT_METHOD	SHA512
----------------	--------

Configurações de segurança

LOGIN_RETRIES	5
LOGIN_TIMEOUT	60
FAIL_DELAY	4

chage é utilizado em sistemas Linux para alterar e visualizar as informações de expiração de senha de um usuário. Ele permite definir políticas como a validade da senha, o tempo de aviso antes da expiração e a data da última alteração. Isso é útil para forçar os usuários a alterarem suas senhas regularmente, contribuindo para a segurança do sistema.

Comando **chage**

```
chage [opções] [usuário]
```

Principais Opções do **chage**

- **l, --list**: Mostra as informações de expiração da senha do usuário.

```
chage -l joao # Exibe as configurações de expiração de
senha do usuário 'joao'
```

- **m, --mindays [dias]**: Define o número mínimo de dias que a senha deve ser usada antes de ser alterada.

```
chage -m 7 joao # Define que a senha de 'joao' deve ser
usada por pelo menos 7 dias antes de poder ser alterada.
```

- **M, --maxdays [dias]**: Define o número máximo de dias que a senha pode ser usada antes de expirar.

```
bashCopiar código
chage -M 90 joao # Define que a senha de 'joao' expira
após 90 dias.
```

- **W, --warndays [dias]:** Define o número de dias de aviso antes que a senha expire.

```
bashCopiar código
chage -W 7 joao # O sistema avisará 'joao' 7 dias antes
de a senha expirar.
```

- **I, --inactive [dias]:** Define o número de dias após a expiração da senha durante os quais a conta permanece ativa antes de ser desativada.

```
bashCopiar código
chage -I 30 joao # A conta de 'joao' será desativada 30
dias após a expiração da senha, se não for alterada.
```

- **E, --expiredate [data]:** Define a data de expiração da conta do usuário.

```
bashCopiar código
chage -E 2024-12-31 joao # Define que a conta de 'joao'
expira em 31 de dezembro de 2024.
```

- **d, --lastday [data]:** Define a data da última alteração da senha.

```
bashCopiar código
chage -d 2024-01-01 joao # Define que a última alteraçã
o de senha de 'joao' foi em 1º de janeiro de 2024.
```

Exemplo de Uso

Para listar as configurações de expiração de senha do usuário `joao`, você usaria:

```
bashCopiar código
chage -l joao
```

Adicionando Usuários a determinados grupos

```
#Adiciona o usuario a um grupo
sudo usermod -G nerds alfredo
```

```
#Faz com que o usuário faça parte do grupo para um determinado
su - nome_do_grupo      # para entrar no grupo (com base na vi
newgrp - nome_do_grupo  # vai pedir a senha para entrar no gp
```

```
Faz com que o usuário faça parte do grupo
gpasswd -a nome_do_usuario nome_do_grupo #para entrar no grup
```

```
Adiciona um 'lider' ao grupo
gpasswd -A nome_do_usuario_lider nome_do_grupo
```

```
Adiciona um 'lider' ao grupo e como 'membro' do grupo
gpasswd -M nome_do_usuario_lider nome_do_usuario_membro nome_
```

Modificando os Usuários

- Para modificar, usa-se o `usermod`

```
#adicionando o usuario a um grupo
sudo usermod -a -G sudo joao
```


#Define o novo valor do ID do usuário.

Exemplo: `usermod -u 1001 username`

`-g, --gid`: Define o novo valor do ID do grupo

Exemplo: `usermod -u 5555 -g 0`

`-c`: Define um comentário para o grupo

Exemplo: `usermod -u 5555 -g 0 -c "giropops strigus"`

`-s, --shell SHELL`: Altera o shell de login do usuário. É para alterar o shell, basta dar um `(chsh lua)`

Exemplo: `usermod -s /bin/bash girus`

`-l`: altera o nome de login de um usuário sem afetar outras

Exemplo: `usermod -l newname oldname`

`-m`, moveria o diretório home do usuário para `/novo/diretorio`.
`usermod -m -d /novo/diretorio usuário`

Exemplo: `usermod -m -d /home/strigus_novo strigus`

`-b, --badnames`: Permitir nomes não convencionais para as con

`sudo usermod -b username`

`-e, --expiredate EXPIRE_DATE`: Definir a data de expiração da

`sudo usermod -e 2024-12-31 username`

`-f, --inactive INACTIVE`: Definir o número de dias após a expi

```
sudo usermod -f 30 username
```

-g, --gid GROUP: Definir um novo grupo primário para a conta

```
bash
```

```
sudo usermod -g newgroup username
```

-G, --groups GROUPS: Definir uma nova lista de grupos supleme

```
bash
```

```
sudo usermod -G group1,group2 username
```

-a, --append: Adicionar o usuário a grupos suplementares sem

```
sudo usermod -a -G newgroup username
```

-l, --login NEW_LOGIN: Definir um novo nome de login para a c

```
sudo usermod -l newlogin username
```

-L, --lock: Bloquear a conta de usuário.

```
sudo usermod -L username
```

-m, --move-home: Mover o conteúdo do diretório home para um n

```
sudo usermod -m -d /new/home/dir username
```

-o, --non-unique: Permitir o uso de IDs de usuário duplicados

```
sudo usermod -o -u 1001 username
```

-p, --password PASSWORD: Definir uma nova senha criptografada

bash

```
sudo usermod -p '$6$random_salt$encrypted_password' u
```

-s, --shell SHELL: Definir um novo shell de login para a conta

bash

```
sudo usermod -s /bin/bash username
```

-u, --uid UID: Definir um novo UID (identificador de usuário)

```
sudo usermod -u 1002 username
```

-U, --unlock: Destruir a conta de usuário.

```
sudo usermod -U username
```

-v, --add-subuids FIRST-LAST: Adicionar um intervalo de UID s

```
sudo usermod -v 10000-20000 username
```

-V, --del-subuids FIRST-LAST: Remover um intervalo de UID sub

```
sudo usermod -V 10000-20000 username
```

-w, --add-subgids FIRST-LAST: Adicionar um intervalo de GID s

```
sudo usermod -w 10000-20000 username
```

-W, --del-subgids FIRST-LAST: Remover um intervalo de GID sub

```
sudo usermod -W 10000-20000 username
```

-Z, --selinux-user SEUSER: Definir um novo mapeamento de usuá

```
sudo usermod -Z system_u username
```

- `chfn` é usado para alterar as informações de comentário do

```
`chfn [opções] [nome_do_usuario]`
```

Por exemplo, para alterar o nome completo de um usuário chama
chfn -f "John Doe" johndoe

Modificando os Grupos

- Para modificar, usa-se o `groupmod`

```
gpasswd nome_do_grupo
```

Dá uma senha para um grupo

-g, --gid GID: Alterar o ID do grupo para um novo valor.

```
sudo groupmod -g 1001 groupname
```

-h, --help: Exibir mensagem de ajuda e sair.

```
groupmod -h
```

-n, --new-name NEW_GROUP: Alterar o nome do grupo para um novo nome.

```
sudo groupmod -n newgroupname groupname
```

-o, --non-unique: Permitir o uso de IDs de grupo duplicados.

```
sudo groupmod -o -g 1001 groupname
```

-p, --password PASSWORD: Alterar a senha do grupo para uma nova senha.

```
sudo groupmod -p '$6$random_salt$encrypted_password' groupname
```

-R, --root CHROOT_DIR: Especificar um diretório raiz para a execução.

```
sudo groupmod -R /newroot groupname
```

-P, --prefix PREFIX_DIR: Especificar um diretório prefixo onde o grupo é executado.

```
sudo groupmod -P /newprefix groupname
```

gpasswd

- Utilizado para administrar grupos em sistemas Linux. Ele permite adicionar ou remover usuários de grupos, definir senhas para grupos e configurar o administrador de um grupo. Isso é especialmente útil para gerenciar permissões de acesso em um ambiente multiusuário.

Sintaxe do Comando **gpasswd**

```
gpasswd [opções] [grupo]
```

```
# Adiciona o usuário 'joao' ao grupo 'devs'.
```

```
gpasswd -a joao devs
```

```
#Definir Senha para um Grupo
```

```
sudo gpasswd nerds
```

```
#Adicionar um Usuário como Administrador do Grupo
```

```
sudo gpasswd -A alfredo nerds
```

```
#Comando para Remover um Grupo
```

```
delgroup nome_do_grupo
```

```
#Remove 'ana' do grupo 'devs'.
```

```
gpasswd -d ana devs
```

Principais Opções do **gpasswd**

- **a, --add [usuário]:** Adiciona um usuário a um grupo.

```
gpasswd -a joao devs # Adiciona o usuário 'joao' ao grupo 'devs'.
```

- **d, --delete [usuário]:** Remove um usuário de um grupo.

```
gpasswd -d joao devs # Remove o usuário 'joao' do grupo 'devs'.
```

- **A, --administrators [usuários]:** Define administradores do grupo.

```
gpasswd -A maria,paulo devs # Define 'maria' e 'paulo'
como administradores do grupo 'devs'.
```

- **M, --members [usuários]:** Define os membros do grupo, substituindo a lista atual.

```
gpasswd -M joao,ana,paulo devs # Define 'joao', 'ana' e
'paulo' como os membros do grupo 'devs'.
```

- **[grupo]** (sem opções): Solicita a definição de uma senha para o grupo. Essa senha pode ser usada para acesso temporário ao grupo através do comando `newgrp`.

```
gpasswd devs # Define uma senha para o grupo 'devs'.
```

Exemplos de Uso

1. Adicionar um usuário a um grupo:

```
gpasswd -a joao sudo # Adiciona 'joao' ao grupo 'sudo'.
```

2. Remover um usuário de um grupo:

```
gpasswd -d ana devs # Remove 'ana' do grupo 'devs'.
```

3. Definir administradores de um grupo:

```
gpasswd -A maria devs # Define 'maria' como administradora do grupo 'devs'.
```

4. Definir membros de um grupo (substitui membros atuais):

```
gpasswd -M joao,ana,paulo devs # Define 'joao', 'ana', e 'paulo' como os únicos membros do grupo 'devs'.
```

5. Definir uma senha para o grupo:

```
gpasswd devs # Define uma senha para o grupo 'devs'.
```