



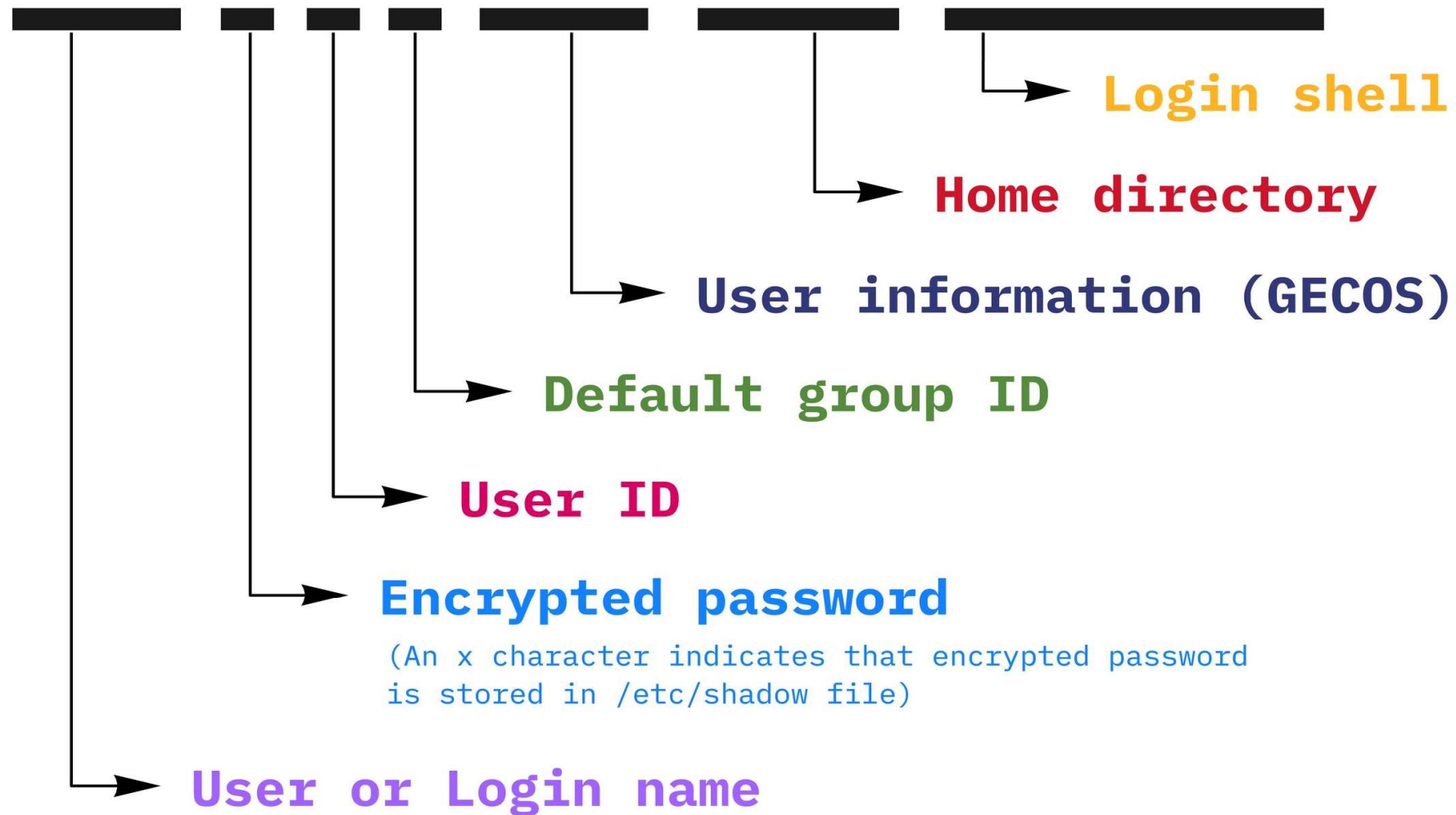
Debian server Administração de Usuários e Grupos

Gerenciar usuários e grupos

- Gerenciar usuários e grupos em uma máquina Linux é um dos principais aspectos da administração do sistema. De fato, o Linux é um sistema operacional multiusuário no qual vários usuários podem usar a mesma máquina ao mesmo tempo.
- As informações sobre usuários e grupos são armazenadas no diretório **/etc/**:
 - **/etc/passwd** - arquivo de sete campos delimitados por dois pontos contendo informações básicas sobre os usuários
 - **/etc/group** - arquivo de quatro campos delimitados por dois pontos contendo informações básicas sobre os grupos
 - **/etc/shadow** - arquivo de nove campos delimitados por dois pontos contendo senhas de usuário
 - **/etc/gshadow** - arquivo de quatro campos delimitados por dois pontos contendo senhas de grupo criptografadas

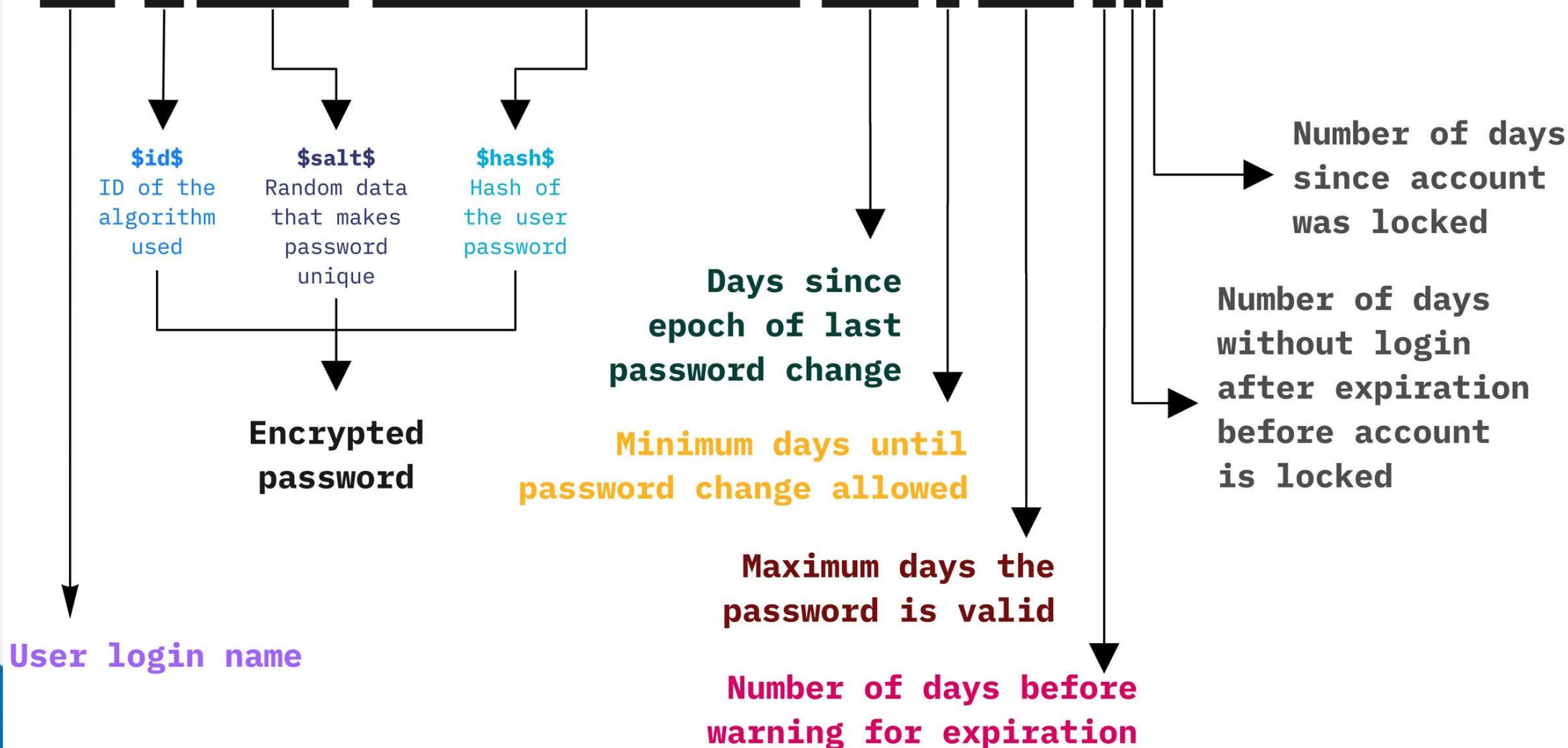
Explicando o arquivo passwd

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



Explicando o arquivo shadow

user:\$1\$.pj1GQnU\$ZWEt3P2/CpbqeiWGa/Ymd1:18556:0:99999:7:::



Contas de Usuários

- Todo usuário de um sistema Linux possui uma conta associada que, além das informações de login (como nome de usuário e senha), também define como e onde o usuário pode interagir com o sistema.
- Os privilégios e controles de acesso definem os “limites” dentro dos quais cada usuário pode operar.

Identificadores de usuários e Grupos (UIDs/GIDs)

- Os **Identificadores de Usuário (UIDs)** e **Grupos (GIDs)** são as referências básicas, numeradas, às contas.
- Os usuários e grupos são enumerados de forma independente, de modo que o mesmo **ID** pode representar tanto um usuário quanto um grupo.
- Todo usuário possui um **UID** e um **GID** primário.
- Por padrão, nos sistemas Linux, todo usuário é incluído em um grupo com o mesmo nome que seu nome de usuário e um **GID** idêntico a seu **UID**.
- Por exemplo, ao criar um novo usuário chamado **aluno**, seu grupo por padrão também será **aluno**.

Comando `id` - identificando usuário

- Lista informações atuais de um usuário na linha de comando
- `id`

```
aluno@debian:~$ id
uid=1000(aluno) gid=1000(aluno) grupos=1000(aluno),24(cdrom),25(floppy),29(audio),30(dip),44(video),46(plugdev),100(users),106(netdev),112(bluetooth),114(lpadmin),117(scanner)
```
- 1000 = *UID* de usuário, seguido pelo nome de usuário (nome comum ou nome de login) entre parênteses.
- 1000 = *GID* o *ID* primário de grupo, seguido pelo nome do grupo (nome comum) entre parênteses.
- Uma lista de GIDs adicionais (nomes de grupo) a que o usuário também pertence.

Comando **id**

- Mostra informações como o **UID** (User ID), **GID** (Group ID) e os grupos suplementares de um usuário.
- Sintaxe básica do comando:
 - **-u**: Exibe o UID (User ID) do usuário.
 - **-g**: Exibe o GID (Group ID) do usuário.
 - **-G**: Exibe os grupos suplementares aos quais o usuário pertence.

Comando **last** – exibe usuário logado

- Exibi informações sobre os logins anteriores dos usuários no sistema, inclui o nome de usuário, data e hora do login e outros detalhes relevantes.
- **last**

```
aluno@debian:~$ last
```

root	pts/1	192.168.0.5	Sun Sep 24 19:01 - 20:28	(01:27)
aluno	tty2	tty2	Sat Sep 23 16:35	gone - no logout
reboot	system boot	6.1.0-12-amd64	Sat Sep 23 16:35	still running
reboot	system boot	6.1.0-11-amd64	Sat Sep 23 16:34 - 16:35	(00:01)
aluno	tty2	tty2	Sat Sep 23 16:20 - down	(00:13)

Comando **last**

- **-n**: Limita o número de entradas exibidas. Por exemplo, `last -n 5` exibirá os últimos 5 logins.
- **-f**: Especifica o arquivo de log a ser usado. O arquivo de log padrão é `/var/log/wtmp`, mas você pode especificar outro arquivo, se necessário.
- **-i**: Mostra os endereços IP associados aos logins.
- **-t**: Exibe o tempo gasto em cada sessão de login.
- **-x**: Mostra o sistema no qual o usuário estava logado.

Quem está logado no sistema?

- Os comandos `who` lista o nome terminal e data do usuário ativo.

```
aluno@debian:~$ who
aluno      tty2          2023-09-26 18:38 (tty2)
```

- O comando `w` lista além do nome, terminal e data e dados relacionados ao programas executados.

```
aluno@debian:~$ w
18:40:45 up 3 min,  1 user,  load average: 0,60, 0,74, 0,33
USUARIO  TTY      DE              LOGIN@  OCIOSO JCPU   PCPU  O  QUE
aluno    tty2     tty2            18:38   3:05   0.06s  0.05s /usr/libexec/gn
```

Criando usuário novos

- O comando **useradd** é uma das maneiras de se criar um novo usuário, é necessário nível administrativo para tal.
 - **-m**: Cria um novo usuário e cria o diretório home para o novo usuário, geralmente em /home/NOME_DE_USUARIO.
 - **-g**: Especifica o GID (Group ID) principal do novo usuário.
 - **-G**: Especifica grupos suplementares aos quais o novo usuário pertencerá.
 - **-s**: Define o shell padrão do novo usuário. Por exemplo, **-s /bin/bash** define o shell como Bash.
 - **-d**: Especifica o diretório home personalizado para o novo usuário.
 - **-p**: Define a senha criptografada do novo usuário. Normalmente, é mais seguro usar o comando **passwd** após criar o usuário para definir a senha.
 - **-e**: Define a data de expiração da conta do usuário, formato AAAA-MM-DD.
 - **-c**: Define o campo comentários (nome, usuário, telefone e etc...)

Criando usuário novos

- Criar um usuário com diretório padrão, comentário João da Cruz e o nome da conta de login cruz.
- `# useradd -m -c "João da Cruz" cruz`
- `# Su cruz`
- Nota que ao trocar de usuário para cruz, o shell está diferente pois não foi definido o shell como bash.

```
$ pwd  
/home/cruz
```

- Comando para alterar o shell
- `chsh --shell /bin/bash <nome_usuario>`

Criando usuário novos

- Criar um usuário com diretório padrão, definindo o bash como padrão shell e o nome da conta de login maria.
- # useradd -m -s /bin/bash maria
- # Su maria

```
maria@debian:/$
```

Criando senhas para usuários

- O comando **passwd** no Linux é usado para definir ou alterar a senha de um usuário. Permite que os administradores ou próprios usuários configurem/modifiquem suas senhas de acesso ao sistema.
- **-l**: Bloqueia (desativa) a senha da conta do usuário, tornando-a inutilizável para fazer login.
- **-u**: Desbloqueia (ativa) a senha da conta do usuário, permitindo que ela seja usada para fazer login novamente.
- **-d**: Remove a senha da conta do usuário, tornando-a acessível sem senha (não é recomendado).
- **-S**: Exibe o status da senha da conta, mostrando informações sobre se a senha está definida ou desabilitada.
- **-e**: Força o usuário a alterar a senha na próxima vez que fizer login.
- **-x**: Define a data de expiração da senha, após a qual o usuário será forçado a alterar a senha.
- **-i**: Define o número de dias após o qual a senha pode ser alterada novamente.

Comando `passwd`

- Definindo senha para usuário
- `# passwd <nomeusuário>`

- Exigindo que o usuário altere a senha

- `#passwd -e cruz`

```
root@debian:~# passwd -e cruz
```

```
passwd: senha modificada.
```

```
root@debian:~# su cruz
```

```
Mude sua senha imediatamente (aplicado pela raiz).
```

```
Mudando senha para cruz.
```

```
Atual senha:
```

```
Nova senha:
```

```
Redigite a nova senha:
```

Alterar conta de usuário

- O comando **usermod** é usado para modificar as informações de um usuário existente no sistema, como nome, grupos, diretório home e configurações de conta.
 - **-l NOVO_NOME**: Renomeia a conta de usuário
 - **-c "NOVO_COMENTÁRIO"**: Define um novo comentário/descrição.
 - **-e DATA_EXPIRAÇÃO**: Define a data de expiração da conta do usuário.
 - **-g NOVO_GID**: Define o novo GID (Group ID) principal do usuário.
 - **-aG GRUPO**: Adiciona o usuário a um grupo suplementar (usa -a para adicionar).
 - **-d NOVO_DIRETÓRIO**: Define um novo diretório home para o usuário.
 - **-s NOVO_SHELL**: Define um novo shell padrão para o usuário.
 - **-u NOVO_UID**: Define um novo UID (User ID) para o usuário.

Comando usermod

- Alterar o nome do usuário cruz para joao_cruz
- `# usermod -l joao_cruz cruz`
- Definir expiração da conta do joao_cruz para 31/12/2023.
- `# usermod -e 2023-12-31 joao_cruz`



Deletando usuários

- O comando **deluser** é usado para excluir uma conta de usuário no sistema Linux. Necessita de privilégios para execução.
 - **--remove-home**: Remove o diretório home do usuário e seu conteúdo.
 - **--remove-all-files**: Remove todos os arquivos associados ao usuário, incluindo os que não estão no diretório home.
 - **--backup**: Faz um backup dos arquivos do usuário antes de excluí-los.
 - **--backup-to**: Define o diretório onde o backup dos arquivos do usuário será armazenado.
 - **--remove-shell**: Remove a entrada do shell da conta do usuário no arquivo `/etc/passwd`.
 - **--remove-all-groups**: Remove o usuário de todos os grupos aos quais pertence.

Comando deluser

- Deletando usuário joao_cruz

- #deluser joao_cruz

```
root@debian:~# deluser joao_cruz
Removendo crontab ...
Removendo usuário 'joao_cruz' ...
userdel: user joao_cruz is currently used by process 3060
deluser: '/usr/sbin/userdel joao_cruz' retornou código de erro 8. Saindo.
```

- Verificando os processos

- # ps -aux

- Matando todos os processos do usuário joao_cruz

- # killall -u joao_cruz

```
root@debian:~# killall -u joao_cruz
```

```
Sessão terminada, matando o shell...root@debian:~# ...morto.
Terminated
```

```
root@debian:~# deluser joao_cruz
Removendo crontab ...
Removendo usuário 'joao_cruz' ...
Concluído.
```

Criando grupos

- O comando `groupadd` no Linux é usado para criar um novo grupo de usuários no sistema.
- Grupos são usados para organizar e gerenciar permissões de acesso a arquivos e recursos.
- Criando grupo
- `# groupadd <nomegrupo>`

Comando `groupadd`

- Criar um grupo chamado alunos
 - `#groupadd alunos`
- Verificar grupos existentes
 - `#cat /etc/group`
 - `#getent group`
- Excluindo um grupo
 - `#groupdel alunos`

Gerenciamento de Usuários e Grupos

- Inserir usuário no grupo
- # `gpsswd -a <usuário> <nomegrupo>`
- Deleta usuário do grupo
- # `gpsswd -d <usuário> <nomegrupo>`
- Mudar usuário de grupo
- # `usermod -G <nomeusuario>`

Comando `getent`

- É usado para consultar informações de bancos de dados, incluindo bancos de dados de usuários e grupos, que normalmente são armazenados em arquivos como `/etc/passwd` e `/etc/group`, bem como outros bancos de dados suportados.
- `# getent <nome_base_dados> [key ...]`
- `# getent group`
- `# getent gpasswd`