

TAG Linux (Entrega até dia 08/04)

Primeira Parte

Instale o Arch linux e faça com que um vídeo do youtube seja executado **com som**. Faça um relatório da instalação com fotos e/ou vídeo.

link para o tutorial: [https://wiki.archlinux.org/title/Installation_guide_\(Português\)](https://wiki.archlinux.org/title/Installation_guide_(Português))

Segunda Parte:

1. Explique com suas palavras o que faz o comando utilizado para instalar o Arch Linux “arch-chroot”.

Os dados do sistema são armazenados com a estrutura de dados denominada árvore.

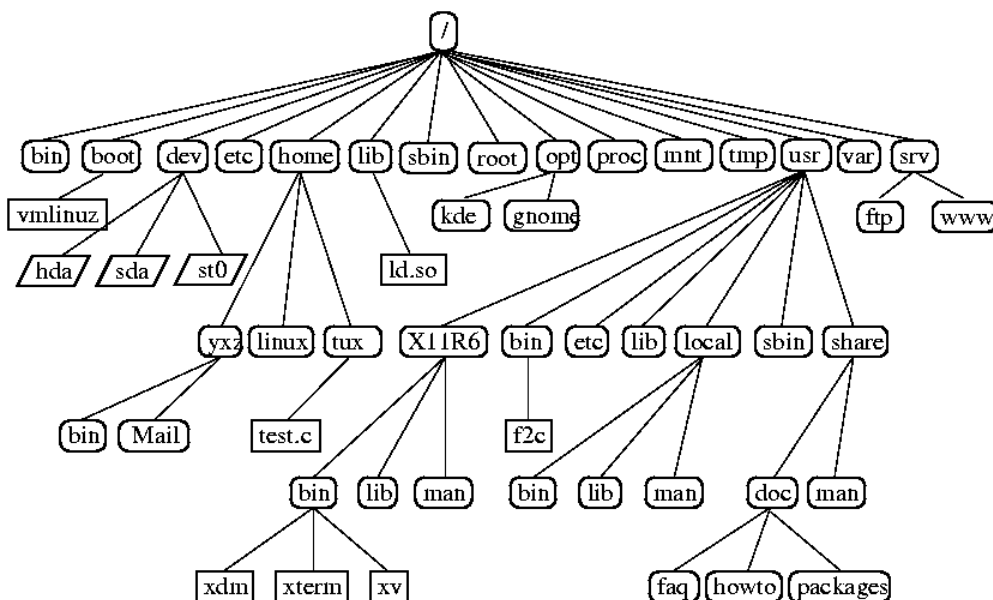
Chroot = change root (muda a raiz)

“arch-chroot”, chroot -> O comando faz entrar no chroot na /mnt é onde está a raiz.

Uma chroot é uma operação que muda o diretório raiz para o processo atual de funcionamento e seus filhos, ou seja, durante a instalação utilizamos esse código para mudarmos a raiz “/” anterior para a nova raiz “mnt”, onde serão instalados pacotes, além dos básicos, feito com o pacstrap(no relatório de instalação) e configurá-los. A “mnt” age como se fosse root do sistema sendo instalado.

Mas por que fazemos o chroot?

Para criarmos uma camada de isolamento, o que for executado no mnt, fica restrito ao mnt, criando um novo ambiente isolado para diretório root do sistema principal. Este novo ambiente é conhecido como “jaula chroot”. Comandos executados nessa jaula não tem ideia de que exista algo fora, pois ele não possui nenhum ponteiro para diretórios externos e, como definimos o mnt como raiz de uma “subcamada”, a princípio os comandos executados dentro do mnt não sabem que há estão restritos a uma “subcamada” ha falsa impressão de estarem sendo executado no sistema de arquivos raiz.



2. Como funciona o comando do terminal sudo?

R:

O comando sudo nos permite executar instruções com privilégios de segurança de outro usuário, por padrão, vem setado como superusuário. Uma vez que damos a instrução, nos é solicitado a senha pessoal para confirmarmos a identidade e executar as nossas instruções dadas, verificando um arquivo, chamado sudoers, que o administrador do sistema configura. É possível passarmos um usuário como parâmetro para o comando sudo, neste caso, vamos executar o comando sudo com os privilégios desse usuário passado como parâmetro.

3. Em um servidor web precisou-se dar acesso para que um usuário execute seus arquivos “.php”.

Podemos deixar em branco.

4. Detalhe 3 Shell's e suas respectivas diferenças.

- 1) C Shell → Inclui características úteis de programação como aritmética embutida e sintaxe de expressão tipo C.
- 2) Bourne Shell → Não possui recursos para uso interativo como a capacidade de lembrar comandos anteriores. Também carece de manipulação aritmética e lógica de expressão embutida. É a shell padrão para o SO Solaris.
- 3) Korn Shell → É um superconjunto da shell Bourne(2), por isso suporta tudo na concha Bourne e tem características interativas. Inclui características como aritmética embutida e arrays tipo C, funções e instalações de manipulação de cordas. É mais rápido que a shell C. É compatível com o roteiro escrito para a shell C.

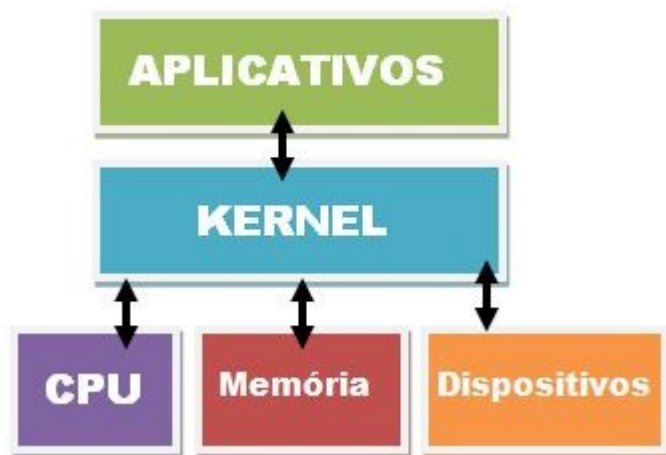
5. Qual a diferenças entre um serviço no Linux e um processo?

Processo → um processo é qualquer instância ativa (em execução) de um programa. um programa é qualquer arquivo executável, mantido em armazenamento em sua máquina. Toda vez que você executa um programa, você criou um processo.

Serviço → é um componente de aplicação que pode realizar operações de longa duração em segundo plano. Ele não fornece uma interface de usuário. Por exemplo, um serviço pode lidar com transações de rede, tocar música, executar arquivos ou interagir com um provedor de conteúdo, tudo a partir sem gerar mudanças na interface visual, passando despercebido pelo usuário.

6. Por que é errado chamar o Linux de Sistema Operacional?

Linux é “apenas” o kernel de um sistema operacional chamado gnu/linux, o kernel esta contido no sistema operacional, mas não diz respeito ao sistema operacional como um todo. Segue abaixo a ilustração do funcionamento de um sistema operacional.



7. Para que servem as entradas na árvore de diretórios do Linux:

a. `/dev/stdout`

`/dev` é onde temos os dispositivos “stdou”(standart out). Com isso, temos no diretório “`/dev/stdout`” temos os dispositivos padrões de saída, impressora e display da tela, por exemplo.

b. `/tmp/.X11-unix`

X11 é um protocolo usado pra interfaces gráficas no linux, onde o computador funciona como um servidor e os programas falam com o servidor o que exibir na tela. Este arquivo é como socket, serve para se comunicar com o servidor.

c. `/proc/cpuinfo`

Nesse diretório, temos um arquivo de texto, somente para leitura, que contem informações sobre as CPUs(unidade central de processamento) em um computador. Havendo mais de uma CPU, esse arquivo de texto separa a informação referente a cada uma dessas CPU por meio de uma linha em branco.

d. `/etc/shadow`

Esse diretório, serve como um banco de dados em arquivos de texto, lá estão guardados os nomes e as senhas dos usuários. Para alterar esse arquivo precisamos de privilégios root, mas pode ser lido por qualquer usuário. Mas como funciona a segurança se todos podem ler o nome e senha do usuário? A senha de cada usuário é armazenada de forma criptografada dentro do arquivo `/etc/passwd`. Estas credenciais são descriptografadas usando uma função de hash unidirecional para que não possam ser descriptografadas

e. `/var/spool`

Nesse diretório, guardamos os dados que serão processados posteriormente, por um programa, usuário ou administrador, muitas vezes os dados são apagados após terem sido processados.

f. `/etc/fstab`

Nesse diretório configurar-se uma regra em que sistemas de arquivos específicos são detectados e, em seguida, montados automaticamente na ordem desejada pelo usuário toda vez que o sistema é

inicializado. Permite que o usuário evite erros de ordem de carregamento que podem consumir tempo e energia valiosos.

8. Qual a diferença entre o /tmp/.X11-unix para as outras entradas?

X11 é um protocolo usado pra interfaces gráficas no linux, onde o computador funciona como um servidor e os programas falam com o servidor o que exibir na tela. Este arquivo é como socket, serve para se comunicar com o servidor.

9. O que são links no Linux? E qual(is) a(s) diferença(s) entre um hard link e um soft link?

Links são atalhos, apontam para outro arquivo ou diretório em qualquer sistema de arquivos, podendo haver dois tipos, os soft links e os hard links.

Soft links → ele passa a referenciar o arquivo original. Isso significa que ao ler ou editar o link simbólico, você estará lendo e editando o arquivo original. Porém, caso você exclua o link simbólico, saiba que o arquivo original continuará intacto. O contrário já não acontece: se o arquivo original for excluído ou renomeado, o link simbólico continua existindo, mas apontando para lugar nenhum

hard links → só podem referenciar arquivos, e nunca diretórios; não podem referenciar arquivos em outros volumes ou partições; hardlinks continuam funcionando mesmo que o arquivo original tenha sido apagado; e hardlinks fazem referência ao inode do arquivo original. Ao contrário dos softlinks, que referenciam nomes de arquivos e diretórios, os hardlinks referenciam uma posição física da partição.

Inode → pode ser entendido como uma tabela de ponteiros de arquivos

10. Por que ao se ligar um dispositivo periférico a um sistema computacional com Linux ele deve ser montado para poder ser utilizado?

A montagem não é apenas dar acesso aos arquivos, é o momento em que dizemos ao sistema de arquivos que o dispositivo possui, se for apenas de leitura ou acesso de leitura/gravação, além de ser o momento em que definimos onde podemos acessar esses arquivos.

11. Qual a função da variável de ambiente PATH?

A variável PATH é uma variável que contém uma lista ordenada de caminhos onde o sistema operacional irá procurar por executáveis ao executar um comando. Usar estes caminhos implicar em não precisar especificar um caminho absoluto ao executar um comando.

12. Quais as vantagens de se colocar o diretório /home em uma outra partição separada do "/" ?

A vantagem é que, diante da necessidade de formatarmos o computador, podemos formatar a partição aonde está o sistema operacional e a partição do /home continuará ilesa. Podemos reinstalando qualquer outra distro, basta configurar a /home para a partição que colocamos.

13. O que são os sistemas de init? Quais as diferenças entre o sysvinit e o systemd?

“init” inicializes é um processo daemon, que começa assim que o computador é iniciado e continuar a executar até o desligamento. De fato, o init é o primeiro processo que começa quando um computador é iniciado, tornando-se o pai de todos os outros processos em execução direta ou indiretamente e, portanto, normalmente é atribuído “pid = 1”.

systemd → é o pai de todos os outros processos direta ou indiretamente, e é o primeiro processo que começa na inicialização, portanto, tipicamente atribuído um “pid = 1”. Systemd foi projetado para superar as deficiências de inicialização. É em si é um processo em segundo plano que é projetado para iniciar processos em paralelo, reduzindo assim o tempo de inicialização e sobrecarga computacional. Tem muitos outros recursos, em comparação com o init.

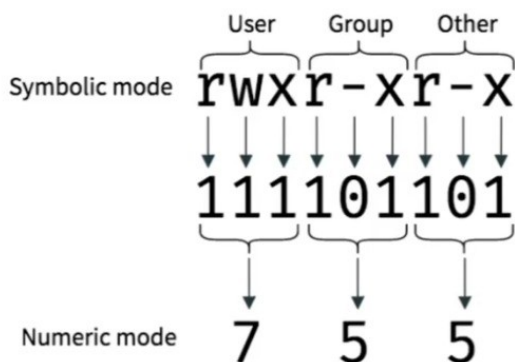
Sysvinit → O init inicia um processo em série, ou seja, uma tarefa só começa depois que a execução da última tarefa foi bem sucedida e ela foi carregada na memória. Isso muitas vezes resultou em tempo de inicialização mais longo.

De forma bem resumida, o systemd e o sysvinit são “organizadores de processos”, mas o systemd obtém vantagem por executar instruções em paralelo

14. Pra que servem as flags especiais de permissão do Linux setuid, setgid e sticky bit?

As flags especiais de permissão, settam as regras de permissões para acessar determinado arquivo e o que pode fazer ou não com aquele arquivo. Baseadas nas regras abaixo:

- 1) user → O seguinte conjunto de três caracteres (rwx) é para as permissões do proprietário.
- 2) group → O segundo conjunto de três caracteres (rwx) é para as permissões do Grupo.
- 3) other → O terceiro conjunto de três caracteres (rwx) é para as permissões de Todos os Usuários.



setuid: bit que faz uma execução com os privilégios do proprietário do arquivo

setgid: bit que faz uma execução com os privilégios do grupo do arquivo

sticky bit: bit que permite que somente o proprietário ou a raiz possa excluir arquivos e subdiretórios