**QUESTIONÁRIO DE ELABORAÇÃO DE ESTUDO PARA CERTIFICAÇÃO**

**COMO FUNCIONA A HIERARQUIA DE SISTEMAS DE ARQUIVOS DO LINUX ?**

* esta hierarquia funciona como uma árvore de cabeça para baixo sendo que o root é o nível mais alto.
* é mantido pelo FHS (File Hiearchy system), que é composto por empresas como DELL, SUN, IBM, etc …
* nesta estrutura estão sendo contemplados 13 arquivos obrigatórios e 2 arquivos opcionais (são considerados opcionais, que são home e root)
* o diretório /proc é onde ficam armazenados os arquivos do kernel, através do conteúdo dele é possível coletar informações de cpu, memória, disco, etc.
* o diretório /boot é onde ficam alocados os sistemas de inicialização do kernel, é lá onde os arquivos que serão procurados para iniciar o sistema, também contém arquivos de grub.
* o diretório /lib é onde contém as bibliotecas do sistema, funciona similar as .dll do windows.
* o diretório /var/ é onde ficam armazenadas as variáveis do sistema, arquivos de log que se movimentam com frequência são armazenados neste diretório
* /opt/ neste diretório ficam armazenados os arquivos que são de programas não nativos.
* /tmp/ são arquivos temporários, no debian automaticamente este diretório é zerado ao reiniciar o sistema.
* /bin e /sbin são os diretórios onde contém os binários, dentro destes diretórios são os binários que cada usuário tem permissão para executar, no caso do bin qualquer usuário, já no caso do sbin somente usuários com permissão de root.
* /home e /root são arquivos que são pessoas, onde ficam armazenados documentos ou dados de usuários.
* /media são onde são montado os dispositivos de media especiais, por exemplo um celular android, o conteúdo de um dvd, etc.
* /mnt/ é utilizado para montagem e compartilhamento entre servidores.
* /dev/ diretório onde ficam armazenados os dispositivos;
* /srv ficam armazenados dados de aplicações dos servidores.

**O QUE É O CONCEITOS DE DEVOPS**

* visa a colaboração e trabalho em equipe dos times de desenvolvimento e infraestrutura, trata-se de uma metodologia.
* Entre suas premissas estão a documentação de servidores, automação de ambientes , e documentação de aplicações.

**COMO COLETAR INFORMAÇÕES DO KERNEL**

* uname -a

**COMO COLETAR INFORMAÇÕES DA DISTRIBUIÇÃO**

* Debian : lsb\_release ou cat /etc/debian-release
* Centos : cat /etc/redhat-release

**COMO COLETAR INFORMAÇÕES DO CPU**

* **cat /proc/cpuinfo**

processor : 0

vendor\_id : GenuineIntel

cpu family : 6

model : 69

model name : Intel(R) Core(TM) i3-4005U CPU @ 1.70GHz

stepping : 1

microcode : 0x1f

cpu MHz : 1698.140

cache size : 3072 KB

physical id : 0

siblings : 4

core id : 0

cpu cores : 2

apicid : 0

initial apicid : 0

fpu : yes

fpu\_exception : yes

cpuid level : 13

wp : yes

flags : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov pat pse36 clflush dts acpi mmx fxsr sse sse2 ss ht tm pbe syscall nx pdpe1gb rdtscp lm constant\_tsc arch\_perfmon pebs bts rep\_good nopl xtopology nonstop\_tsc aperfmperf eagerfpu pni pclmulqdq dtes64 monitor ds\_cpl vmx est tm2 ssse3 fma cx16 xtpr pdcm pcid sse4\_1 sse4\_2 movbe popcnt tsc\_deadline\_timer aes xsave avx f16c rdrand lahf\_lm abm ida arat epb pln pts dtherm tpr\_shadow vnmi flexpriority ept vpid fsgsbase tsc\_adjust bmi1 avx2 smep bmi2 erms invpcid xsaveopt

bogomips : 3392.28

clflush size : 64

cache\_alignment : 64

address sizes : 39 bits physical, 48 bits virtual

power management:

* **uptime**

23:37:25 up 22 min, 2 users, load average: 0,43, 0,44, 0,30

| | | | | |

| | | | | |

| | | | | carga média últimos 15 min

| | | | carga média últimos 10 min

| | | carga média últimos 5 min

| | Usuários no sistema

| Tempo em que está ligado

Hora atual

**COMO COLETAR INFORMAÇÕES DA MEMÓRIA**

* cat /proc/meminfo ou free -m

**COMO LISTAR OS DISCOS DO SISTEMA**

* fdisk -l lista todos os devices que são reconhecidos em sistema, as saídas aparecem de forma sda, sdb, etc … Os dispositivos usb também são mostrados neste comando.

**ARQUIVOS E FUNCIONALIDADES**

* **/etc/passwd** - São descritos os usuários do sistema :

No Debian os id’s dos usuários são divididos da seguinte forma , 0 (ids d0 usuário administrador), <1000 (Usuários de sistema) e >1000 (id de usuários finais, ou seja, usuários criados normalmente).

No CentOS os id’s são divididos da mesma forma a diferença é que ao invés de 1000 o valor fica em 500.

dbus:x:81:81:System message bus:/:/sbin/nologin

| | | | | | |

| | | | | | shell do usuário, neste caso não possui shell

| | | | | home do usuário

| | | | comentário ou descrição

| | | id do grupo

| | id do usuário

| senha (se x, significa que a senha está no /etc/shadow, um \*usuário bloqueado)

nome do usuário

* /etc/shadow - São descritos as senhas dos usuários , ela são criptografadas.
* /etc/groups - Onde ficam descritos os grupos existentes no sistema.
* /etc/vim/vimrc - arquivo para personalizar o vim no debian
* /etc/vimrc - arquivo para personalizar o vim no centos
* /etc/issue - Mensagem antes do login (debian e centos)
* /etc/issue.net - Mensagem antes do login remoto (debian e centos)
* /etc/motd - Mensagem após login (debian e centos)
* /etc/profile - define variáveis e configurações para todos os usuários
* .bash\_profile ou .profile - carrega as variáveis para os cada usuário que usam autenticação , este é o primeiro arquivo que é carregado
* .bashrc - carrega as variáveis para os usuários que não usam autenticação (por exemplo quando se está em modo grafico , ai clicamos em aplicativos > terminal (não precisa de senha para logar no bash), este é o segundo arquivo a ser carregado ao logar
* .bash\_login : executa quando o usuário efetua login
* .bash\_logout : efetua quando o usuário efetua logout
* variável TMOUT é responsável pelo tempo de inatividade do sistema, o usuário é desligado do shell caso fique inativo pelo tempo apontado.
* variável HISTSIZE é responsável pela quantidade de linhas que são armazenadas no history
* variável HISTTIMEFORMAT=’%m-%d-%y %H:%M - ’ guarda a data e a hora do comando digitado. Isso pode ser modificado em .bash\_profile (por exemplo)
* /etc/environment - Localização das variáveis de ambiente, essas variáveis são carregadas ao iniciar o SO.
* o pacote bash-completion faz com que o terminal complete com o tab
* cat /proc/sys/kernel/hostname - Altera o nome da máquina sem precisar reiniciar.

**CONFIGURAÇÃO DE TECLADO**

* Debian : dpkg-reconfigure keyboard-configuration e depois reiniciar /etc/init.d/keyboard-setup restart
* CentOS : loadkeys -d br-abnt2 (configuração para teclado brasileiro) ou loadkeys us-acentos (configuração para teclado sem acento)

**COMO FUNCIONA A ENTRADA E SAÍDA DE DADOS NO LINUX**

* O processo de entrada de dados é representado pela sigla **stdin** (standard input), o processo de entrada é feito através de um teclado por exemplo.
* O processo de saída é representado pela sigla **stdout** (standard output), é o processo de saída de um comando, pode ser representado por um monitor por exemplo.
* O processo de saída de erro é representado pelo **stderr** (standard error) , os erros também são mostrados na tela, mas neste caso somente os erros.
* As saídas de comandos (stdout) podem ser manipuladas através dos comandos abaixo.
* **>** file.txt : Adiciona o conteúdo a um arquivo
* **>>** file.txt : Adiciona a saída ao final de um arquivo
* file\_new.txt **<** file.txt : Acrescenta o conteúdo de fil.txt a file\_new.txt, trabalhando como uma entrada de dados.
* O stdin é representado por 0
* O stdout é representado por 1
* O stderr é representado por 2
* 2> file : representa o redirecionamento de um erro para um arquivo
* 2>> file : representa a adição de um erro ao final do arquivo.
* 2>&1 : Direciona toda saída de erro e toda saída para um arquivo
* &> file : direciona toda saída para um arquivo.
* &>> file : adiciona conteúdo ao final do arquivo.

**COMANDOS**

**CAT** - comando para exibir conteúdo de um arquivo.

cat /etc/arquivo.txt - exibe o conteúdo do arquivo arquivo.txt

cat < arquivo.txt - usa o arquivo arquivo.txt como entrada de texto e exibe na tela

cat << EOF > arquivo.txt - será permitido a adição de um texto , após gravar o texto basta pressionar EOF que o arquivo estará salvo.

cat - comando para visualizar arquivos na tela

cat -b passwd - numera as linhas de um arquivo que não forem vazias

cat -T passwd - Mostra as linhas de tabulação com ^I

cat -A passwd - Mostra todos os caracteres, inclusive o "enter"

cat -n passwd - numera todas as linhas inclusive linhas vazias

cat -E passwd - Mostra um sinal de $ após o final de cada linha

**TEE** - Faz um redirecionamento mostrando na tela, é semelhante ao pipe, porém ao redirecionar ele mostra a saída em tela.

tripwire --check | tee /var/log/tripwire\_log

**LN** - Cria links entre arquivos e diretórios, um **link simbólico** é semelhante a um atalho, caso o arquivo original não exista mais, o link simplesmente não funcionará, um link simbólico cria um atalho com o inode diferente do arquivo original, desta forma se for deletado ou modificado o atalho, nada acontece com o original. O **hard link** quando criado, cria uma cópia do arquivo, inclusive com o mesmo inode, desta forma qualquer alteração tanto no link quanto no arquivo original, afetarão os dois arquivos.

ln -s original /tmp/atalho

ln original /tmp/atalho\_hardlink

**UNIQ-** Comando para identificação de linhas duplicadas.

uniq -u file1 - Mostra apenas as linhas que não repetem

uniq -cd file1 - Conta e mostra apenas as linhas duplicadas.

**SORT** - Comando para ordenar a exibição de um conteúdo

sort file1 - Ordena por ordem alfabética o arquivo file1 (iniciando pelos números)

sort -r file1 - Ordenação inversa do conteúdo do arquivo

sort -h file1- ordenação humana do conteúdo (iniciando por letras e depois números)

sort -f file1 - ordenação ignorando case sensitive.

**SPLIT** - Comando para fatiar um arquivo, exemplo de uso é quando precisamos enviar um arquivo de e-mail e o anexo está muito grande.

split -b 10MB file.tar arquivo-

**DD** - Comando que clona hd, partições, cria imagens de discos e cds no linux

if = input file

of= output file

ibs=escreve os bytes por vez (padrão 512)

obs=grava os bytes por vez (padrão 512)

conv= realiza conversão a partir de uma lista:

noerror= não para o processamento em caso de erros

sync = caso um bloco não seja preenchido por completo, dados nulos são inseridos

notrunc= os dados não sao truncados ao final e são mantidos íntegros.

**Criar uma imagem**

dd if=/dev/sr0**(cd montado)** of=/sdb1/imagem.iso **(diretório montado em sdb1 e o arquivo que sera criado com o nome de imagem.iso)** bs=2048 **(tamanho do bloco de 2048)** conv=noerror,sync

dd if=/dev/sr0 of=/sdb1/imagem.iso bs=2048 conv=noerror,sync

**Clonar uma partição**

dd if=/dev/sdb1of=/dev/sdc1 bs=4096 conv=noerror

**Clonar um disco**

dd if=/dev/sdb of=/dev/sdc bs=1M conv=noerror

**Backup do MBR incluindo tabela de partição**

dd if=/dev/sda**(O sistema operacional fica instalado na unidade a)** of=/dev/sdc1/MBR.img**(salvará dentro de sdc1)** bs=512**(a partição MBR possui 512 blocos)** count=1 **(ele vai copiar somente uma partição de 512 blocos (bs=512))**

dd if=/dev/sda of=/sdc1/MBR.img bs=512 count=1

**ALIAS** : são apelidos dados a comandos, para se criar um alias a sintaxe é:

alias contar=’wc -l /etc/passwd’ … isso faz com que ao digitar contar no bash ele conte as linhas de /etc/passwd.

**SOURCE** : recarrega as configurações de um arquivo.

**SET** : Lista variáveis locais no Debian.

**ENV**: lista variáveis globais

**UNSET** : remove o valor de uma variável … sintaxe unset variável

**LOCALE**: Lista variáveis relacionadas a localização.

**nl** - comando que lista linhas de um arquivo, similar ao cat -b

nl -ba /etc/lilo.conf - numera todas as linhas inclusive linhas em branco

nl -bt /etc/passwd - numera todas as linhas que não são vazias

**wc** - contador de palavras, letras e linhas

wc -m - conta a quantidade de caracteres de um arquivo

wc -b - conta a quantidade de bytes, ou seja, espaço em branco, etc

wc -l - conta a quantidade de linhas de um aquivo

**head** - Comando que lista o inicio de um arquivo

head -n 10 file - MOstra as 10 primeiras linhas de um arquivo

head -c12 file - mostar os 12 primeiros caracteres do arquivo teste

**tail** - Mostra o final de um arquivo

tail -200 teste - mostra as ultimas duzentas linhas de arquivo teste

tail +200 teste - MOstra a partir da linha duzentos do arquivo teste

tail -f teste - mostra o conteudo do arquivo em tempo real

tail -c12 teste - conta os doze primeiros caracteres do final de um arquivo.

tail -fs2 teste - mostra em tempo real o conteudo do arquivo fazendo uma checagem de 2 em 2 segundos.

**echo** - imprime um conteudo na tela

echo -e "teste\n" - habilita o reconhecimento do backspace e imprime teste e pula uma linha

echo -E "teste\n" - desabilita o reconhecimento do backspace e, neste caso ele não irá pular uma linha, mas imprimira o \n

echo -e "teste\tteste" - imprime teste e imprime novamente após uma tabulação horizontal

echo -e "teste\vteste" - imprime teste e imprime novamente após uma tabulação vertical (escadinha)

**rm** - Remove um conteudo

rm -i arquivo - apresenta um prompt para remover um arquivo

rm -I arquivo - apresenta um prompt em caso de remoção recursiva ou em mais de 3 arquivos

rm -f arquivo - força a remoção de um arquivo

rm -rf diretorio - forçã a remoção de um diretório e seu conteudo recursivamente.

rm -d diretoirio - remove um conteudo vazio

**ls** - listagem de arquivos e diretórios

ls -a - imprime o conteúdo do diretório atual incluindo arquivos ocultos.

ls -l - mostra o conteúdo em listas

ls -i - apresenta o endereço do inod do arquivo

ls -h - apresenta o tamanho do arquivo em saída humana

ls -t - lista com os arquivos mais novos em primeiro lugar

ls -r - inverte a ordem de ordenação

**file** - mostra o tipo do arquivo

file teste.txt - mostra o tipo de arquivo teste.txt

**mv** - renomear ou mover um arquivo

mv -i arquivo arquivo1 - pergunta antes de sobrescrever

mv -f arquivo arquivo1 - não pergunta antes de sobrescrever

mv -n arquivo arquivo1 - não sobrescreve nenhum conteúdo

mv -u arquivo arquivo1 - sobrescreve somente se o arquivo de origem for mais novo ou não conter no local de destino.

**dd** - Faz cópia bit a bit

=============================================================

if = input file

of= output file

ibs=escreve os bytes por vez (padrão 512)

obs=grava os bytes por vez (padrão 512)

conv= realiza conversão a partir de uma lista:

noerror= não para o processamento em caso de erros sync = caso um bloco não seja preenchido por completo, dados nulos são inseridos notrunc= os dados não sao truncados ao final e são mantidos íntegros.

==============================================================

Criar uma imagem

dd if=/dev/sr0(cd montado) of=/sdb1/imagem.iso (diretório montado em sdb1 e o arquivo que sera criado com o nome de imagem.iso) bs=2048 (tamanho do bloco de 2048) conv=noerror,sync

dd if=/dev/sr0 of=/sdb1/imagem.iso bs=2048 conv=noerror,sync

Clonar uma partição

dd if=/dev/sdb1of=/dev/sdc1 bs=4096 conv=noerror

Clonar um disco

dd if=/dev/sdb of=/dev/sdc bs=1M conv=noerror

Backup do MBR incluindo tabela de partição

dd if=/dev/sda(O sistema operacional fica instalado na unidade a) of=/dev/sdc1/MBR.img(salvará dentro de sdc1) bs=512(a partição MBR possui 512 blocos) count=1 (ele vai copiar somente uma partição de 512 blocos (bs=512))

dd if=/dev/sda of=/sdc1/MBR.img bs=512 count=1

**split** - Divide o arquivo em partes

split -b 30M arquivo arq. - Dividirá o arquivo em partes de 30 M e a saída será arq.\*

split -b 12k arquivo arquivo-split- divide o arquivo em 12k com o nome de saída de arquivo-split\*

**tac** - imprime arquivo em ordem ao contrária

tac -b teste - imprime o arquivo teste com o conteúdo em uma única linha caso tenha espaço

tac arquivo - imprime o conteúdo de arquivo em ordem invertida.

**sort** - comando que faz ordenação de conteúdo.

sort -f arquivo - ordena o arquivo ignorando maiúsculas e minúsculas

sort -r arquivo - ordenação reversa do arquivo.

**uniq** - remove as linhas duplicadas.

uniq -cd - conta as linhas repetidas e imprime somente as linhas repetidas.

uniq -cu - faz a contagem das linhas repetidas , porém printa somente o que não é repetida.

**grep** - comando para pesquisa de texto dentro de arquivo.

grep -i “teste” arquivo - Pesquisa pela palavra teste dentro de arquivo ignorando case sensitive desativado

grep -n “teste” arquivo - Pesquisa pela palavra teste dentro de arquivo numerando as linhas

grep -v “teste” arquivo - Pesquisa pela palavra teste dentro de arquivo invertendo a busca.

grep "teste" -A2 test - Pesquisa por teste dentro de test , e quando achar imprime duas linhas depois

grep "teste" -B2 test - Pesquisa por teste dentro de test , e quando achar imprime duas linhas antes

**egrep** - é um grep melhorado.

egrep “teste|sty” < file - procura teste ou sty dentro de file

egrep -e "^es|\$ste" < test - procura o que inicia com es e termina com ste e imprime

egrep ^[ste] < test - Pesquisa o que contém na inicialização os caracteres s, t ou e e imprime.

echo Sunday | sed 's/day/night/'

sed 's/one/ONE/' <file

sed 's/\/usr\/local\/bin/\/common\/bin/' <old >new

sed 's\_/usr/local/bin\_/common/bin\_' <old >new  
sed 's:/usr/local/bin:/common/bin:' <old >new

sed 's|/usr/local/bin|/common/bin|' <old >new  
sed 's/abc/(abc)/' <old >new - Procurando conteúdo de uma string

awk - Ferramenta para manipulação de texto

awk -f myscript.awk myfile.in - o script externo .awk é executado tratando o arquivo .in.

awk '{ print }' /etc/passwd - Imprime cada linha do arquivo passwd (similar ao cat)

awk '{ print $0 }' /etc/passwd - Similar ao código anterior

awk '{ print "" }' /etc/passwd - Imprime linhas em branca, a cada linha do arquivo será vazia

awk '{ print "hiya" }' /etc/passwd - Imprime Hiya e, cada linha do arquivo.

awk -F":" '{ print $1 }' /etc/passwd - Coleta o primeiro campo antes do delimitador “:”

awk -F":" '{ print $1 $3 }' /etc/passwd - Imprime o primeiro e o terceiro campo do arquivo.

awk -F":" '{ print $1 " " $3 }' /etc/passwd - Imprime um espaço vazio após o campo 1 e 3 do arquivo.

awk -F":" '{ print "username: " $1 "\t\tuid:" $3 }' /etc/passwd - A saída seria essa:

username: halt uid:7  
username: operator uid:11

cut - Manipulador de texto

echo 'aaa:bbb:ccc' | cut -f1 -d':' - O Resultado seria aaa, pois iria pegar o primeiro campo antes do delimitador “:”

echo 'aaa:bbb:ccc' | cut -f2 -d':' - O Resultado seria bbb, pois iria pegar o primeiro campo antes do delimitador “:”

echo 'tux-tole' | cut -c1-3 - Rsultado seria tux, pois este comando serve para coletar tudo dentro de um intervalo.

echo 'Windows ou Linux' | cut -c12- - O resultado seria LInux, pois o comando pega e imprime tudo a partir do caracter 12.

echo 'Windows ou Linux' | cut -c12-15 - O resultado seria Linu, pois o comando pega e imprime tudo a partir do caracter 12 até o 15.

**EDITOR DE TEXTO**

vim - Editor de texto vi, só que melhorado.

Os comandos :set number ou :set nu mostram as linhas do terminal numeradas.

O comando :set nohlsearch remove o destacado das palavras (ao realizar uma pesquisa).

O comando gg vai para a primeira linha

O comando G vai para o final do arquivo.

O comando u desfaz alguma alteração.

O comando yy copia e p cola , ex y2y p (cópia duas linhas e cola)

O comando dd deleta a linha inteira (d2d, deleta duas linhas)

Os comandos shitf + c ou cc recortam

O comando ctrl+r refaz uma alteração.

O comando set ic ignora o case sensitive nas pesquisas.

O comando vim -o file1.txt file2.txt - abrir dois arquivos ao mesmo tempo horizontalmente

O comando vim -O file1 file2 - abrir dois arquivos ao mesmo tempo na vertical.

O comando :! ls /etc/passwd - De dentro de um arquivo é possível executar um comando no bash, digitando (dois pontos, exclamação, espaço e o comando

O comando :r ls /etc/passwd - Executa um comando sem sair do vim

:set nohlsearch - elmina o destaque em uma pesquisa de palavras no vim

:set syntax on - Ativa a coloração e destaque de conteudo no vim

:set syntax off - Desativa a coloração e destaque no vim

:set number - ativa a numeração de linhas

:set nonumber ou :set nonu - desativa a numeração

:help - mostra ajuda

:set vsplit - divide a janela em vertical

:split - divide a tela em horizontal

ctrl+w - alterna entre janelas após o uso do vsplit.

:e /etc/passwd - Abre um novo arquivo.

:r! /cat /etc/passwd - Executa um comando e a saída ele insere dentro o arquivo aberto atualmente.

:x - sai salvando

shift + zz - sai salvando.

**PACOTES ÚTEIS**

* figlet - pacote responsável por gerar conteúdo com letras escritas em pipe

**QUAL SÍMBOLO REPRESENTA PERMISSÃO TOTAL**

O símbolo # representa permissão total, já o $ representa permissão parcial ou não administrativa.

**O QUE É A VARIÁVEL PS1**

* A variável PS1 é usada para alterar como as informações do terminal são apresentadas, exemplo :

vim /etc/profile +10

export PS1=”nome\_da\_mina\_maquina - [\w]# ”

\w indica que mostrará o diretório atual completo

\W Diretório corrente

\w Caminho completo do diretório corrente

\u Nome do usuário

\t Hora do sistema

\d Data

\h Host da máquina

**Join** - comando para juntar arquivos

join -t: arquivo1.txt arquivo2.txt - Junta dois arquivos e imprime na tela separando os pelo : , ou é possível manda-lo para um arquivo separado;

**COMANDOS DE REDE DEBIAN**

ifconfig - mostra as interfaces no debian

ifconfig eth1 down/up - apenas habilita e desabilita a placa, ou seja, não altera sua configuração.

ifdown e ifup - desabilita e habilita a placa modificando a configuração, fazendo um reset, fazendo com que a placa inicie de acordo com a configuração definida dentro de /etc/network/interfaces.

mtr- comando traça a rota completa , mostrando todos os pontos que eu pacote passa ate chegar ao destino.

host - informações de domínio

dig- informações de domínio mais completo.

[www.ipok.com.br](http://www.ipok.com.br)

traceroute - comando que traça uma rota do pacote da origem ao destino, mas ele mostra apenas até o roteador.

ifconfig - mostra o ip do servidor no debian

netstat - comando para listar serviços e estado de portas

/etc/resolv.conf - Arquivo para armazenamento de configuração de endereços dns

mii-tool eth0 - comando do pacote mii-tool usado para identificar se um caso está conectado fisicamente

route add default gw 192.168.200.1 - Adicionando uma rota padrão.

route del default - Deleta uma regra default.

**COMANDOS DE REDE CENTOS**

ip a - mostra as interfaces no centos

ip a - Mostra o ip do servidor

ip addr show - mostra o ip do servidor

ip address - Mostra o ip do servidor

ip route show - mostra o gateway padrão no centos

route -n - mostra o gateway padrão no debian

ss -ntpl - Similar ao comando netstat para o centos 7

ethtool eth0 - pacote ethtool deve ser instalado , a partir dai é possível utilizar o comando ethtool para verificar se o link está fisicamente conectado.

ip addr add 192.168.200.20/24 dev eth0 - Configurando um ip via linha de comando

ip link set eth0 down - desative a placa de rede

ip link set eth0 up - ative a placa de rede

ip add a 172.17.3.2/16 dev eth0 label eth0:0 - Cria um ip virtual e uma placa virtual.

ip route add default via 192.168.200.1 - Adicionando uma rota default

ip route del default via 192.168.200.1 - Deleta uma rota padrão.

**DEBIAN** **CENTOS**

/etc/hostname /etc/sysconfig/network ou /etc/hostname

/etc/network/interfaces /etc/sysconfig/network-scripts/ifcfg-eth0

/etc/network/interfaces /etc/sysconfig/network-scripts/ifcfg-eth0

/etc/resolv.conf /etc/resolv.conf

/etc/hosts /etc/hosts

**O QUE É O SHELL**

Shell é uma ferramenta de acesso ao sistema básico.

Sistema operacional > Shell > Kernel > Hardware

**VARIÁVEIS IMPORTANTES**

* HOME → Define o diretório home do usuário logado;
* PATH → Define os diretórios usados para encontrar os comandos;
* SHELL → Define o shell que está sendo utilizado;
* PWD → Define em qual diretório você está no momento;
* USER→ Define o usuário que está logado;
* EDITOR → Define o editor de textos padrão para aplicações que invocam editores de texto automaticamente.
* LANG → Especifica o idioma\_PAIS local. Podem ser especificados mais de um idioma na mesma variável separando-os com :, desta forma caso o primeiro não esteja disponível para o programa o segundo será verificado e assim por diante; ➢ LC\_MESSAGES → Especifica o idioma que serão mostradas as mensagens dos programas. Seu formato é o mesmo de LANG; ➢ LC\_ALL → Configura todas as variáveis de localização de uma só vez. Seu formato é o mesmo de LANG.

Exemplo:

export LANG=pt\_BR

export LC\_ALL=pt\_BR

export LC\_MESSAGES=pt\_BR

**GERENCIAMENTO DE PACOTES DEBIAN**

which top : lista o caminho de execução do top (/usr/bin/top)

dpkg -S /usr/bin/top : mostra qual pacote preciso instalar para ter o top na máquina (neste caso o resultado seria procps)

dpkg -l : lista os pacotes instalados

dpkg -I( i maiúsculo) htop.deb : traz informações do pacote htop ainda não instalado

dpkg -I htop.deb | grep Depends : coleta as informações de dependências do pacote

dpkg -L htop : Lista onde os arquivos do htop estão armazenados

dpkg -i ou --install htop.deb : Instala um pacote

dpkg -r htop : remove um pacote parcialmente . ou seja, mantém os arquivos ainda no sistema

dpkg --purge : remove o programa e os seus arquivos, fazendo uma remoção completa.

dpkg -C (não precisa de argumento): Faz auditoria nos pacotes para ver se nada foi alterado.

dpkg -­s bash - verifica o status do pacote bash.

**GERENCIADOR DE PACOTE ALTO NÍVEL DEBIAN**

O APT(advanced packaging tool) e YUM (yellowdog update modified) tem como principal função realizar a verificação e instalação das dependências, preparando para que o dpkg ou rpm entre em ação, ou seja, após terminarem de preparar as dependências eles enviar o pacote para o gerenciador de baixo nível realizar a instalação.

As maneiras possíveis dos gerenciadores de alto nível trabalharem é através de repositórios, seja em cd rom, repositório local e o mais comum repositório remoto.

apt-get update - Atualiza a lista de repositórios

apt-get upgrade - atualiza os pacotes.

apt-cache search “log color” : pesquisa por algum pacote que tenha log color relacionado.

apt-cache show htop : mostra informações detalhadas sobre um pacote.

apt-get install htop : instalar um pacote

apt-get remove htop : remove o pacote htop mas mantém os arquivos de configuração

apt-get remove --purge : remove o pacote e também seus arquivos de configuração.

apt-get clean : limpa o diretório /var/cache/apt/archives

apt-get dist-upgrade: atualiza a distribuição, por exemplo, caso esteja usando a versão wheeze e queira usar a versão jessie, basta substituir wheeze por jessie dentro de /etc/apt/source.list e atualizar o repositório e atualizar a distribuição (apt-get update && apt-get dist-upgrade).

apt-get autoremove: remove os pacotes que não possuem mais nenhuma ligação, pacotes inutilizaveis

apt-get -f install : Identifica pacotes que estão com problemas e tenta instalá-los

apt-get -f remove : Identifica pacotes com problema e tenta removê-los

apt-get -d htop : faz apenas o download do pacote htop, mas não instala.

**CICLO DE DESENVOLVIMENTO DO DEBIAN**

O Debian possui três distribuições disponíveis:

unstable(sid) : está em fase de teste não homologado

testing(jessie): em fase de homologação , sendo preparado para a próxima versão.

stable (wheeze): versão estável, já homologada e sem bugs.

main : contém pacotes que estão completamente de acordo com o debian free software guide line

contrib : conjunto de programas de código aberto que não podem funcionar sem o elemento não livre

non-free : contém programas que não estão completamente de acordo com os pré-requisitos do debian software free guide-line, mas podem ser distribuídos normalmente.

**REPOSITÓRIO DEBIAN**

/etc/apt/source.list - Arquivo de configuração de um repositório do debian.

deb http://ftp.debian.org/debian wheezy main contrib non­free

deb: tipos de pacotes

http://… : endereço remoto do repositório

wheeze: distribuição

main, contrib, non-free: sessões dos pacotes.

/var/cache/apt/archives : diretório de arquivos baixados via apt-get .

**GERENCIAMENTO DE PACOTES RED HAT**

which top : Localizo o diretório do binário do comando top

rpm -qf /usr/bin/top : localiza o pacote responsável pela instalação do comando top (procps-ng)

rpm -qpi htop.rpm : Concede informações de um pacote ainda não instalado.

rpm -qpR htop.rpm : Lista as dependências de um pacote a ser instalado.

rpm -ivh htop.rpm : Faz a instalação de um pacote listando os detalhes e com barras de progresso.

rpm -e htop : remove o pacote htop

rpm -ql htop: lista os arquivos de htop

rpm -Va (sem argumento): Faz auditoria nos pacotes para ver se nada foi alterado.

O gerenciador de pacote de alto nível do centos é o yellow dog modified (yum)

yum search htop : procura o pacote “linux logo”

yum repolist : lista os repositórios que estão sendo usados.

yum update --skip-broken: atualiza o repositório e ignora os pacotes quebrados.

yum info htop : apresenta informações sobre o pacote htop

yum install htop : instala o pacote htop

yum remove htop : remove o pacote htop

yum clean all : limpa o diretório de cache do yum (var/cache/yum)

yum upgrade --skip-broken: atualiza os pacotes ignorando os pacotes quebrados.

yum install htop -y --downloadonly : faz apenas o download do pacote, faz o download para /var/cache/yum/…

**REPOSITÓRIO CENTOS**

/etc/yum/repos.d/ - diretórios de repositório do centos, e extensão dos arquivos de repositórios são .repo

name : o nome descritivo para o repositório

baseurl : endereço dos arquivos

gpgcheck: se fará a verificação ou não (1 = sim e 0 = não)

gpgkey : qual chave criptográfica utilizada para a verificação dos pacotes.

OBS: As configurações de repositórios também podem ser feitas em /etc/yum.conf

/var/cache/yum : diretório onde contém os arquivos baixados através do yum.

**O QUE É SYSFS**

É um sistema de arquivos que organiza e apresenta informações sobre os dispositivos e barramentos conectados ao sistema. A partir do kernel 2.6 .

/sys - Dispositivos conectados naquele momento (ex, pen drive)

/proc - Diretório virtual, ou seja . é mantido somente enquanto a máquina é ligada, de uso de kernel, contém informações sobre os processos.

/dev - Os dispositivos conectados( ex, placa de vídeo)

**O QUE É DEVFS**

É o gerenciador de dispositivos mais antigo, Ele começa a popular arquivos no boot, de acordo com o que o linux suporta, ou seja, cria arquivos mesmo que não estão em uso dentro de /boot/ . É um sistema de arquivos estático .

**O QUE É UDEV**

Sistema de arquivo o /dev mais atual … Cria apenas os arquivos quando ele reconhece que o dispositivo é conectado. É um sistema de arquivos dinâmico.

**INSTALAÇÃO DE PROGRAMAS VIA CÓDIGO FONTE.**

A instalação de programas via código fonte fonte segue basicamente uma receita de bolo, os passos são :

1 - faz download do pacote

2- ler o arquivo readme /install

3 ./configure = lê as bibliotecas, dependências, prepara o sistema para a instalação, faz um check-list, verifica compilador.

4- make = gera o código fonte em binário

5- make install = instalar a biblioteca, binários e documentação

6- make clean = faz a limpeza em caso de erros.

Pros

customização (performance e tunning)

última versão

processo independente de distro

Contras

Falta de padronização e de controle

Sujeira que ficam no servidor após a instalação.

Perda de performance em caso de adição de opções desnecessárias.

Processo de atualização manual.

Makefile : arquivo que mostra os passos a serem executados nas sessões dos programas via make. Por exemplo, ao abrir um arquivo makefile, teremos a sessão chamada install (onde conterá as instruções caso você digite make install ) , uninstall (com as instruções caso você digite make uninstall ), clean (caso você digite make clean) , e assim por diante.

O makefile fica disponível apenas quando o ./configure é chamado, antes disso fica disponível um arquivo chamado makefile.in

./configure --help : Lista as opções do configure

**BIBLIOTECAS**

São um conjunto de funções que pode ser adicionado a um determinado programa, uma biblioteca é como se fosse um incremento a um programa, para realizar uma determinada tarefa, o programa chama uma biblioteca para executar tal tarefa.

Existem dois tipos de bibliotecas:

bibliotecas compartilhadas: não tem a biblioteca incorporada, mas eu tenho ela no sistema, ela que não vem no programa, mas eu tenho no debian ou centos, ou seja, no programa é feita apenas uma referência de uma biblioteca que deve ser procurada no sistema operacional. Com isso vários softwares podem usar a mesma biblioteca ao mesmo tempo não sendo necessário abrir outras instâncias.

Pontos positivos são: economia de memória, pois como serão carregadas apenas uma bibliotecas para vários programas não será preciso abrir mais de uma instância na biblioteca, o código fica mais enxuto, pois basta fazer a referência a biblioteca já existente ao invés de escrevê-la.

Ponto negativo: Ele perde um pouco mais de tempo pesquisando a biblioteca referenciada.

bibliotecas estáticas: são bibliotecas que já vem incorporadas ao programa.

biblioteca estática: são incorporadas no software, ou seja, não dividem como nenhum outro programa.

Pontos positivos: são usadas de forma independente

Pontos negativos: Se dois programas quiserem usar a mesma biblioteca, serão abertas duas instâncias, se houvesse 1000 programas usando a mesma biblioteca seriam abertas 1000 instâncias, isso causaria um gasto de memória muito maior.

**Diretórios das bibliotecas**

/lib -

/usr/lib/ -

/usr/local/lib -

/etc/ld.so.conf : especifica onde estão configurados as bibliotecas do sistema

/etc/ls.so.cache: arquivo contendo todas as bibliotecas do sistema.

ldconfig - comando utilizado para reconstruir o arquivo so.cache em caso de exclusão ou para recarregar bibliotecas, e também recarregar configurações.

**ACESSO REMOTO SSH**

/etc/ssh\_config : arquivo de configuração para o lado cliente, ou seja, para as conexões de saída, por exemplo, se eu quero acessar uma outra máquina via ssh o arquivo que será lido é o ssh\_config;

/etc/sshd\_config : Arquivo de configuração para o servidor ssh, neste arquivos estão especificadas todas as regras de entrada, por exemplo, caso eu queira permitir uma conexão ssh com este servidor o arquivo que deverá ser alterado é o sshd\_config.

TCPWrappers: Uma biblioteca que provê uma camada adicional de segurança, ele se encontra na camada de acesso, bloqueando o acesso sem utilização de firewall. O bloqueio via tcpwrapper é automático nao sendo necessário reiniciar nenhum serviço.

/etc/hosts.allow - Arquivo de configuração do tcpwrapper que permite a conexão

/etc/hosts.deny - Arquivo de configuração do tcpwrapper que nega a conexão

sintaxe do arquivo:

sshd: 192.168.200. :spawn /bin/echo “$(date) Conexão recusada -SSH-%a” >> /var/log/tcpwrapper.log

**GERENCIAMENTO DE PROCESSOS**

pstree : lista os processos em forma de árvore.

pidof htop: lista os números de processos (pid) de htop, ele só retorna resultado se o nome do programa bater exatamente como o descrito (ex, se eu digitasse pidof hto ele não retornaria nada).

pgrep htop: lista os números de processos (pid) de htop (é similar ao grep, ele faz uma pesquisa pelo conteúdo e retorna, ex… se eu digitasse pgrep hto ele retornaria todo processo que contém hto )

Estados de um processo:

D - Processo morto

R - sendo executado (running)

S - Dormido ininterruptamente(aguardando um processo terminar)

T - Parado por um sinal de controle

Z - Processo zumbi, finalizado mas não encerrado pelo processo pai.

lsof -u root : Listar todos os processos que estão sendo executados pelo root

lsof : lista todos os processos

lsof -i : Lista as conexões ativas, similar ao netstat

lsof /bin/bash : lista quem está usando o diretório ou arquivo.

pkill htop : encerra todos os processos com o nome de htop (semelhante ao killall), mas nesse caso é possível colocar somente uma parte do nome (exemplo pkill hto, encerraria todos os processos que contenham hto )

kill $(pidof crond) ou kill ‘pidof crond’ : executa o comando kill após processar a saida do comando pidof crond , o ‘$’ representa um subshell .

fuser -k /dev/pty/0 - encerra a sessão do usuário logado no pty

alt+f2 e digita xkill : mostra uma “caveira “ na tela e no elemento gráfico que o usuário clicar ele será encerrado.

nohup ping 8.8.8.8 & : faz com que o comando ping seja executado, mas libera o terminal, ou seja, mesmo que eu saia da sessão e retorne o comando continuará sendo executado, é possível acompanhar a saída do comando no arquivo nohup.out que é gerado automaticamente.

renice: altera a prioridade de um programa que já está sendo executado

nice: inicia um processo com uma prioridade diferente

**PROCESSO DE INICIALIZAÇÃO DE UM SISTEMA**

1- Ao ligar o pc a BIOS faz a leitura do hardware (todos os componentes)

2- Após a leitura, o primeiro estágio de bootloader é iniciado , que é o MBR, o MBR possui 512 bytes, e ao ser carregador ele inicia o segundo estágio do bootloader que é a chamada do GRUB/LILO

3- O GRUB ou LILO são chamados e a partir dar as configuraçoes sao lidas e então o kernel é inicializado;

4- Após o kernel ser inicializado, o processo de init é iniciado havendo então a interação com o usuário.

**TIPOS DE INICiALIZAÇÃO**

systemV : Usado pelo Debian ,trabalha com níveis de execução, esses níveis são conhecidos como runlevels

0 : halt (desliga a máquina)

1 ou s ou S : modo onde somente o root possui acesso, monousuário sem acesso a rede e interface gráfica.

2,3,4,5 : Trabalha com multiusuário, não faz distinção de modo gráfico ou texto, pois é possível personalizar no grub … o modo padrão do debian é o 2.

6: nível de reboot

Systemd : Utilizado pelo centos , trabalha com níveis de execução

0 : desligamento do sistema (halt)

1 : modo monousuário, somente o root nesse nível

2 : modo multiusuário sem o compartilhamento de arquivo nfs

3 : modo multiusuário com rede e modo texto

4 : não usado, pode ser personalizado pelo usuário

5 : multiusuário completo com parte gráfica

6 : modo de reinicialização do sistema

No Debian quando colocamos um arquivo na inicialização (/etc/init.d) é criado um apontamento dentro de /etc/rc2.d/ automaticamente.

**INICIAR UM PROGRAMA AUTOMÁTICO NO DEBIAN**

insserv -rv rsyslog : remove o rsyslog da inicialização

insserv -dv rsyslog : adiciona o rsyslog na inicialização (serve para qualquer programa, mesmo um script feito manualmente)

rcconf: programa para configurar inicialização de programas automáticas no debian

sysv-rc-conf: Pacote para inicialização de programas automáticos no debian

/etc/inittab : Arquivo de configuração de runlevel no debian (systemv)

**RUNLEVELS NO SYSTEMD**

systemctl get-default : comando que lista o runlevel atual padrão do sistema

/lib/systemd/system/: diretório que contém as unidades ou serviços e runlevels.

OBS: para adicionar um runlevel como padrão basta criar um link simbólico apontando para o default.target. (ln -sf /lib/systemd/system/multi-user.target /etc/systemd/system/default.target) .

systemctl set-default graphical.target : adiciona o modo gráfico como runlevel padrão.

/etc/systemd/system/multi-user.target.wants/ : contém os programas que iniciam automaticamente.

**TRABALHANDO COM HARDWARE E DISPOSITIVOS**

Sistema de arquivos

devfs : São sistemas de arquivos antigos, que ao iniciar o SO ele carrega dentro do /dev todos os arquivos de dispositivos mesmo que eles são sejam usados ou estejam em uso, com isso há uma grande quantidade de arquivos gravados dentro de /dev.

udev - Sistema de arquivo mais moderno que carrega para o /dev apenas o que é requisitado, ou seja, somente o que está em uso ou será usado.

apt-get install lshw - INstala o pacote para apresentar os hardwares da máquina, permite também a visualização em uma página no navegador

fdisk : Configurar partições de disco

cfdisk : Configurar partições de disco (mais amigável que o fdisk)

yum install parted : Instala o pacote do parted para habilitar o comando partprobe.

partprobe /deb/sdb : Faz uma nova leitura da tabela de partições.

blkid : comando que lista as as partições junto com o UID e filesystem

cat /proc/partitions - Lista as partições

df -Th : mostra os discos montados apresentando também o filesystem.

ext2 : sistema de arquivos utilizado antigamente

ext3,ext4 : são sistemas de arquivos mais atuais e utilizam a tecnologia journal ( a tecnologia journal cria um mapa do que está sendo gravado em disco, como se tirasse uma fotografia do arquivo antes de gravar, isso facilita na hora da recuperação de dados.

xfs : melhor para banco ou arquivos grandes , também utiliza a tecnologia journal

iso: apenas leitura, sistema de arquivos padrão para CD-ROM

swap: Espaço reservado para troca de dados com a memória RAM, em alguns casos ele é considerado um sistema de arquivos.

vfat : sistemas de arquivos windows, conhecido como fat32.

ls -l /dev/disk/by-uuid : lista os discos mostrando a identificação única (UUID)

tune2fs -l grep UUID : coleta o UUID do disco.

**/etc/fstab**

arquivo responsável pela montagem automática dos dispositivos, as opções numéricas no final da linha do arquivo representam 0(backup desativado) 0 (verificação de disco desativada) , 1 (backup ativado) 1 (verificação de disco ativada, para verificação da partição / sempre deve ser usado a opção 1) , a opção 2 ( verificação de disco deve ser usada em partições diferentes da / );

Sist\_arquivos Ponto\_de\_Montagem Tipo Opções dump ordem  
 /dev/sda1 / ext3 defaults 0 1  
 /dev/sda2 /boot ext3 defaults 0 2  
 /dev/sda3 /dos msdos defaults,noauto,rw 0 0  
 /dev/hdg /cdrom iso9660 defaults,noauto 0 0

/etc/mtab : mostra apenas os dispositivos montados no sistema

mkswap /dev/sdb1 : cria uma area swap

swapon /dev/sd1 : ativa o swap

swapoff /dev/sdb1 : desativa o swap

cat /proc/swaps : lista as partiçẽs swap

swapon -s : lista as partições swap que eu tenho em sistema.

mkswap -L SWAP /dev/sdb3 : define um label para o swap

apt-get install cryptsetup: instala o pacote para trabalhar com partições criptografadas.

modprobe dm\_mod : carrega o modulo dm\_mod em sistema

modprobe dm\_crypto : carrrega o modulo dm\_crypto em sistema

UM MÓDULO É UM TRECHO DE CÓDIGO QUE É ADICIONA NOVAS FUNCIONALIDADES AO KERNEL, NESTE CASO O KERNEL FARÁ O GERENCIAMENTO E FARÁ O PAPEL DE CRIPTOGRAFAR UMA PARTIÇÃO, AO ADICIONAR UM MÓDULO AO KERNEL O MESMO PASSARÁ A ENTENDER UMA DETERMINADA FUNÇÃO.

automount : É usado apenas quando uma partição será usada, ou seja, a partição fica desmontada e só será montada em caso de uso.

apt-get install autofs5 : instala o pacote do automount

/etc/auto.master : arquivo de configuração padrão do autofs (automount)

**SERVIDOR DE IMPRESSÃO CUPS**

apt-get install cups-bsd cups-pdf : instala os pacotes do cups e a impressora pdf

lp -d IMPRESSORA1 /etc/hosts : Imprime o arquivo hosts usando a impressora IMPRESSORA1

lpinfo -v : Exibe informações da impressora

lpstat -v : Exibe o status da impressora

**SERVIDOR X**

**GPG**

Criptografia simétrica : A mesma chave que criptografa faz a descriptografia

Criptografia assimétrica : Cada um dos lados devem ter a chave pública e privada.

apt-get install gnupg : instala o pacote gnupg (criptografia assimetrica)

**GERENCIAMENTO DE USUÁRIO**

/etc/passwd - Armazena o nome de usuário e informação geral sobre os usuários.

/etc/shadow - Arazena as senhas e politicas de senha dos usuários

/etc/group - Armazena os nomes dos grupos

/etc/gshadow - Armazena as senhas de grupos

/etc/logins.defs - armazena configurações default para criação de usuário.

/etc/skel - Armazena o conteúdo de skel para um novo usuário, ou seja, é um modelo (esqueleto) para a criação de um usuário.

**getent passwd** - Lista os usuários do sistema, este comando mostra também usuários logados remotamente em nosso servidor.

**getend passwd suporte** - Lista as informações do usuário suporte somente.

**id suporte** - Coleta o id do usuário suporte

**groups suporte** - Mostra qual grupo o usuário suporte está inserido

**yum install finger** - Instala o pacote do finger para visualizar atributos de usuáros

**finger suporte** - Traz informações do usuário suporte

**UID 0** - É o ID do usuário root , é padrão nas distribuições

**GIU 0** - É o GID do usuário root

**USUÁRIOS E GRUPOS DE SISTEMA NO DEBIAN E CENTOS 7**

1 a 999 - Usuários de sistema (exemplo, usuários apache, nrpe)

1000 a 65535 - Usuários normais (exemplo, kennedy)

**USUÁRIOS E GRUPOS DE SISTEMA NO CENTOS 6**

1 a 499 - Usuários de sistema (exemplo, usuários apache, nrpe)

500 a 65535 - Usuários normais (exemplo, kennedy)

**COMANDO CHAGE**

Comando que define políticas e informações de usuários

**COMANDO GPASSWD**

gpasswd -a suporte infraestrutura - Adiciona o usuário suporte ao grupo infra

gpasswd -d suporte infraestrutura - Remove o usuário suporte ao grupo infra

**COMANDO ADDUSER**

adduser suporte - adiciona o usuário suporte ao sistema

**SKEL**

É um diretório modelo (esqueleto) , para fazer com que todos os usuários tenham uma pasta chamada Modelo dentro de sua home, basta criar um diretório chaamado Modelo dentro de /etc/skel. Exemplo:

mkdir /etc/skel/Modelos

**COMANDO CHGRP**

comando que altera o grupo de uma determinada pasta.

exemplo : chgrp suporte Nerds - Altera o grupo da pasta Nerds para suporte.

**UMASK**

comando e função que define a permissão padrão para um usuário ou um sistema

A formula para calculo do umask para diretórios é : 777 - 002(valor umask) = 775

Para os arquivos dentro deste diretório, o valor de permissão será o resultado o umask do diretório menos 111 , exemplo 775 - 111 = 664.

Exemplo : valor umask = 022

mkdir teste (777 - 022= **755**)

touch teste/file.txt (755-111= **644**)

Ou exemplo. qual o valor do meu umask deve ter para que os arquivos tenham permissão 0440 ?

0440+111(valor que é subtraído para arquivos) = 0551

0777-0551 = 0226 (esse deve ser o valor de umask)

Outro exemplo; Qual valor de umask para que os arquivos tenham permissão de 0440

0440+0111=0551

0777-0551=0226

**PERMISSÕES ESPECIAIS**

Suid bit = 4 (valor literal s) : Em binários no momento da execução herda as permissões do dono.

Sguid bit = 2 (valor literal s) : Em pastas, permite a herança de grupos.

Stick bit =1 (valor literal t): Em pastas, restringe a remoção de arquivos e subpastas apenas para o dono.

**QUOTAS DE DISCO**

quotas de disco são uma limitação de gravação de dados no disco imposta aos usuários do sistema.

yum install quota - Instala o pacote de quota de disco.

**COMANDO MUTT**

comando serva para conferir os e-mails do usuário

yum install mutt - Instala o pacote mutt

**O QUE É O PAM (pluggable authentication modules)**

biblioteca usada para definir acessos ao sistema, entre diversos outros tipos de controle de acesso.

Seu desenvolvimento foi feito pela SUN Microsystems

/etc/pam.d - Localização das bilbiotecas

/lib64/security - Localização das biliotecas do PAM.

/etc/pam.conf - arquivo de configuração do PAM.

**O QUE É UM EMPACOTADOR**

Um empacotador junta diversos arquivos em um mesmo local.

Exemplo de empacotadores: tar , cpio

**Empacotar um arquivo usando tar:**

-c = Cria um arquivo

-v = verbose

-z = comprime ou extrai arquivos tar resultante com o gzip

-j = comprime ou extrai arquivos tar com o bzip2

-f =especifica o arquivo tar a ser usado.

-t = mostra a tabela com o conteúdo do backup

tar -cvf bkp\_home.tar /tmp

tar -tf bkp\_home.tar

**Empacotando com gzip**

gzip home\_bkp.tar

gunzip home\_bkp.tar.gz

**Empacotando com bzip2**

bzip2 home\_bkp.tar

bunzip2 home\_bkp.tar.bz2

Fazer um backup usando o CPIO : find /etc/ | cpio -ov > /backup\_etc.cpio

Pesquisando arquivos no backup com CPIO: cpio -t < /backup\_etc.cpio | grep hostname

Restaurar um backup com CPIO : cpio -iv < /backup\_etc.cpio

**O QUE É UM COMPACTADOR**

O compactador diminuo o tamanho do arquivo empacotado liberando espaço em disco.

**DIFERENÇA ENTRE EMPACOTADOR E COMPACTADOR**

O primeiro centraliza os arquivos, o segundo diminui seu tamanho.

**O QUE É XFS**

É um sistema de arquivos , ele tem uma performance para trabalhar com arquivos grandes, ele trabalha com journal (tira uma foto do arquivo, caso tenha problema no sistema, ele consegue recuperar os arquivos).

**FERRAMENTA XFSDUMP**

utilizada para fazer backup de arquivos e seus atributos em um sistema de arquivos.

**COMO FAZER BACKUP COM XFSDUMP**

xfsdump -l 0 -p 30 -f /backup/meu\_dump\_do\_etc /etc - Faz o backup do diretório etc e salva com o nome de meu\_dump … Em seguida devemos informar o nome do arquivo.

xfsdump -l 0 -p 30 -f /backup/meu\_dump\_do\_etc\_&(date +”%d/%m//%y) /etc - Faz o backup do diretório etc e salva com o nome de meu\_dump … Em seguida devemos informar o nome do arquivo.

**FAZENDO BACKUP INCREMENTAL XFSDUMP**

O backup incremental faz um backup de arquivos que são novos, ou seja, que são diferentes do backup inicial, para que isso seja feito é necessário passar o parâmetro 1 na flag -l (éle) (níveis de backup) .

xfsdump -l 1 -p 30 -f /backup/meu\_dump\_do\_etc\_&(date +”%d/%m//%y) /etc - Faz o backup do diretório etc e salva com o nome de meu\_dump … Em seguida colocar o mesmo nome do label colocado inicialmente, exemplo: backup

**COMO FAZER RESTORE USANDO XFSRESTORE**

xfsrestore -f /backup/meu\_dump /tmp/ - O resultado será uma restauração do arquivo backup no diretório /tmp

**COMANDO WATCH**

O comando watch serve para acompanhar em tempo real uma tarefa. Exemplo : **watch date**

**O QUE É CRONTAB**

cron serve para realizar o agendamento de tarefas, geralmente usado para tarefas rotineiras, ou seja, que não são executadas somente uma vez.

crontab -l - Lista os agendamentos para com o usuário de sistema

contab -e - cria uma tarefa no crontab

crontab -lu suporte - lista os agendamentos para o usuário suporte

crontab -ru suporte - remove os agendamentos para o usuário suporte

crontab -r - remove os agendamentos para o usuário de sistema

/etc/crontab.d - Diretório do crontab

echo suporte >> /etc/cron.deny - Arquivo cron.deny responsável por bloquear o usuário de criar agendamentos.

**O QUE É AT ?**

é um executor de tarefas, mas diferente do cron ele executa somente uma tarefa

exemplo de uso:

at 21:00 (enter)

cp -ar /etc/ /tmp/ (enter)

ctrl+c

o serviço do at chamasse atd , **systemctl start atd**.

para bloquear um usuário de criar agendamentos com at ,precisa criar um arquivo dentro de /etc/at.denny ou at.allow.

**COMANDOS ADMINISTRATIVOS**

Comandos administrativos são os comandos que devem ser usados constantemente por um administrador de sistema.

getent : Comando que mostra informações em tempo real sobre determinado arquivo

getent passwd aluno - Ver informações sobre um usuário aluno.

getent passwd - Ver quantos usuários temos no sistema.

vipw : para alterar o passwd (use este ao invés de usar vi /etc/passwd)

vipw -s : abre o arquivo /etc/shadow de modo administrativo

visudo : alterar o arquivo /etc/sudoers de modo administrativo.

vigr : abre o arquivo /etc/group de modo administrativo.

**COMO CONFIGURAR UMA REDE**

Os endereços de rede são definidos pela classe a,b e c . No ipv4 o número de bits em uma rede são 32. Para calcular uma máscara de rede é utilizada a fórmula.

Mas, como Saber sem Usar o Bash para Converter?

11111111.11111111.11111111.11100000

8 + 8 + 8 + 3 = 27

128 64 32 16 8 4 2 1 Soma-se os Bits 1

1 1 1 1 1 1 1 1 128+64+32+16+8+4+2+1 = 255

1 1 1 0 0 0 0 0 128+64+32 = 224

255.255.255.224

Para saber quantas subredes posso criar é só fazer o número de bits elevado a 2. Por exemplo, na rede acima (/27), estão em uso 3 bits, então o cálculo é 2 elevado a 3.

2 x 2 x 2 = 8 (ou seja, é possível ter até 8 redes e usamos a tabela para identificar quantos hosts são possíveis para cada rede. Neste caso :

128 + 64 + 32 = 224 / 8 = 28 hosts em cada subrede.

**UTILITÁRIO MTR**

É um traceroute melhorado, que faz tanto o papel de ping quando o papel de traceroute.

mtr google.com - Envia um ping e rastrea o caminho até chegar no google.

**UTILITÁRIO IPCALC (debian)**

ipcalc 192.168.0.0/24 /25 - divide a rede de classe c em subredes de 25 bits.

**RSYSLOG**

é o pacote de instalação responsável por centralizar os arquivos de log em um único local

, o arquivo de configuração do rsyslog no centos é /etc/rsyslog.conf .

Por padrão ele utiliza a porta 514 tanto para UDP quanto para TCP .

No arquivo de configuração @host (udp) @@host(tcp)

**MUDAR A HORA DO RELÓGIO DA BIOS**

Utilize o comando hwclock -w

**FERRAMENTA VMSTAT**

vmstat é uma ferramenta de acompanhamento do consumo de memória e apresenta em resumo.

vmstat -s - Apresenta um relatório do uso de memória

vmstat 1 10 - APresenta a saída com uma taxa de atualização e o número de linhas

**FERRAMENTA SYSSTAT**

Ferramenta de capacty plaining, que contém um conjunto de comandos para análise de desempenho do sistema.

**PACOTE STRESS**

Ferramenta para realização de teste de stress no servidor.

GRUB

Grub é o gerenciador de inicialização do linux.

No GRUB1

/boot/grub/ - Diretório de armazenamento das configurações do grub.

**ATALHOS TERMINAL**

* Pressione a tecla Del para apagar o caracter acima do cursor.
* Pressione CTRL+A para mover o cursor para o inicio da linha de comandos.
* Pressione CTRL+E para mover o cursor para o fim da linha de comandos.
* Pressione CTRL+U para apagar o que estiver à esquerda do cursor. O conteúdo apagado é copiado para uso com CTRL+y.
* Pressione CTRL+K para apagar o que estiver à direita do cursor. O conteúdo apagado é copiado para uso com CTRL+y.
* Pressione CTRL+L para limpar a tela e manter o texto que estiver sendo digitado na linha de comando (parecido com o comando clear).
* Pressione CTRL+Y para colocar o texto que foi apagado na posição atual do cursor.

**PARTIÇÕES E DISCOS**

A identificação de discos rígidos no GNU/Linux é feita da seguinte forma:

/dev/sda1  
 | | | |  
 | | | |\_Número que identifica o número da partição no disco rígido.  
 | | |  
 | | |\_Letra que identifica o disco rígido (a=primeiro, b=segundo, etc...).  
 | |  
 | |\_Sigla que identifica o tipo do disco rígido (sd=SATA/SCSI, sd=IDE, xt=MFM).  
 |  
 |\_Diretório onde são armazenados os dispositivos existentes no sistema.

Abaixo algumas identificações de discos e partições em sistemas Linux:

* */dev/fd0* - Primeira unidade de disquetes.
* */dev/fd1* - Segunda unidade de disquetes.
* */dev/sda* - Primeiro disco rígido na primeira controladora SATA ou SCSI.
* */dev/sda1* - Primeira partição do primeiro disco rígido SATA ou.
* */dev/sdb* - Segundo disco rígido na primeira controladora SATA ou SCSI.
* */dev/sdb1* - Primeira partição do segundo disco rígido SATA ou SCSI.
* */dev/sr0* - Primeiro CD-ROM SATA ou SCSI.
* */dev/sr1* - Segundo CD-ROM SATA ou SCSI.
* */dev/hda* - Primeiro disco rígido na primeira controladora IDE do micro (primary master).
* */dev/hda1* - Primeira partição do primeiro disco rígido IDE.
* */dev/hdb* - Segundo disco rígido na primeira controladora IDE do micro (primary slave).
* */dev/hdb1* - Primeira partição do segundo disco rígido IDE.
* */dev/xda* - Primeiro disco rígido XT.
* */dev/xdb* - Segundo disco rígido XT.

**IPTABLES**

Qual a tabela padrão do iptables ?

tabela filter.

**DNS**

Para configurar um dns secundário

yum install bind bind-utils - instala os pacotes bind e bind-utils

#Editar o arquivo abaixo

vim /etc/named.conf

#Adicionar ao final um novo arquivo ao qual ele fará a leitura.

include “/etc/named.conf.local”;

#No Resolv.conf adicione dois endereços dns, um para o próprio host e o dns primário

Quando o serviço de DNS é contratado (no brasil a entidade é o Registro.br), o domínio selecionado fica armazenado em 13 servidores globais, que são responsáveis pela resolução dos nomes.

Os pacotes para instalação de um servidor de cache no debian é bind9, no centos é bind. O servidor de cache é útil para economizar banda e aumentar o tempo de resposta nas buscas de site. O pacote bind-utils também deve ser instalado.

No DNS a zona é a configuração de um domínio, esse servidor terá um arquivo de configuração que apontará para os registros( MX, CNAME, etc).

named.conf - Arquivo principal de configuração do DNS.

/var/named - Diretório de armazenamento dos arquivos de registro no Centos

/var/cache/bind/ - Diretório de armazenamento dos arquivos de registro no Debian

rndc - Comando para manipulação de registros no DNS.

host e dig - Ferramentas para verificação de endereços DNS.

MX - Entrada para servidores de e-mail.

A - Re

**APACHE**

/etc/apache2/sites-available - Sites disponíveis

/etc/apache2/sites-enabled - Sites só são de fato disponibilizados quando a configuração estão dentro deste diretório.

**CERTIFICADOS NO APACHE**

Os certificados devem ser gerados por uma entidade certificadora, para que possamos solicitar o fluxo é :

-Gerar uma chave privada

-Criar uma requisição a partir da chave privada e mandando a saída para um arquivo .csr.

-Enviar o csr para a unidade certificadora, ela vai retornar o csr e demais arquivos.

-Para fazer um certificado auto-assinado, usamos a chave .key e o arquivo .csr e a saida será um .crt .

**POSTFIX**

MTA - Mail transport agent , é o responsável pelo envio das mensagens utilizando o protocolo SMTP (simple mail transfer protocol)

MDA - Mail delivery agent, é o responsável pela entrega dos e-mails , utilizam os protocolos POP/IMAP

MUA - Mail user agent , são os clientes de e-mail .

SMTP - trabalha na porta 25, para criptografia é utilizada a porta 465.

IMAP - Faz uma cópia dos seus e-mails no servidor para o cliente, porta 143 e 993 (para SSL)

POP - O pop diferente do IMAP ele move s arquivos do servidor para o cliente. Porta padrão é 110 e 995 (TLS)

main.cf - arquivo principal de configuração do postfix

master.cf - arquivo com configurações adicionais de segurança(mudança de porta, etc).

/etc/mailbox - arquivo com o nome do servidor.

/var/spool/mail/suporte - diretório de armazenamento dos e-mails, a pasta suporte são os e-mails do usuário suporte.

Mailbox - Todas as mensagens ficam armazenadas dentro de um único arquivo (/var/spool/mail/suporte), onde suporte é um **arquivo**

Maildir - Usa um sistema de diretórios e cada e-mail é armazenado em um arquivo diferente, descomente a linha no main.cf **home\_malbox = Maildidr/**

No maildir quando usamos o comando **maildirmake** para criar a estrutura de diretórios (ver slide), ele cria algumas estruturas (new=e-mails novos, cur= quando o arquivo já foi lido, tmp=temporário).

/etc/aliases - arquivo de apelidos de e-mail. exemplo de configuração:

helpdesk: suporte, meu\_usuario. // **quando eu enviar um e-mail para helpdesk, o suporte e meu\_usuario também receberão, é uma lista de distribuição.**

newaliases (sem parâmetro nenhum) - comando utilizado após criar um alias, dentro de /etc/aliases, ele vai re-ler o arquivo.

SASL - Simple authentiction and secure layer é o parametro utilizado para aplicar autenticação no postfix, quando habilitamos o SASL não precisamos fazer o bloqueio por rede no arquivo main.cf (my\_networks).

/etc/default/saslauthd - arquivo de configuração do sasl, para habilitar basta configurar como START=yes.

/etc/postfix/sasl/smtpd.conf - Arquivo que deve ser criado para usar autenticação de usuário para acesso ao e-mail usando sasl .

roundcube - Cliente de e-mail via web, trabalha em conjunto com um servidor web (apache, nginx, etc).

No DNS as configurações de SPF e MX devem ser configuradas para que os e-mails não chegue como SPAM, o proxy reverso também deve estar configurado.

postqueue ou mailq - comandos para trabalhar com filas de e-mail.

ex :

postqueue -p

postqueue -f

**OPENLDAP**

Servidor de diretórios, com ele é possível centralizar a autenticação dos usuários em diversos serviços (e-mail, proxy, etc) em um só lugar.

Trabalha na porta 389 por padrão, porta com criptografia é 636.

Ele trabalha com schemas, onde schemas são objetos que dentro de cada objeto existem atributos. Exemplo:

schema impressora -> atributo nome da impressora, ip da impressora, colorida ... etc

A diferença básica entre um servidor de autenticação LDAP e autenticação via banco de dados (método normal) é que o openldap e fortemente preparado para realizar leituras, uma vez os dados importados, não se faz necessário ficar escrevendo, já em um banco de dados a leitura e escrita acontece a todo momento.

/var/lib/ldap/ - Diretório onde ficam armazenados as bases do ldap

ldapsearch - comando para realizar buscas no ldap

ldapdelete - comando para deletar da base de dados.

ldapadd - comando para adicionar a base de dados

ldappasswd - comando para alterar a senha.

/etc/ldap.conf - arquivo de configuração de cliente.

slapcat - comando para exibir a base

**OPENVPN**

Host to gateway - VPN onde existe um servidor openvpn, e outras diversos computadores se conectam a este servidor e caem na mesma rede.

host to host - apenas duas máquinas conectadas, a configuração é mais simples, apenas as duas máquinas se comunicam.

gateway to gateway - duas redes conectadas, a conexão é mais complexa, as redes por inteiro se conectam. o IPSec é um método mais utilizado para esse tipo de comunicação.

**SERVIDOR NTP**

o Servidor NTP possui hierarquias que são denominadas “strata” , cada um deles corresponde a um “stratum”. Os stratums vão desde o stratum 0 até o stratum 16. Esses stratums ficam sincronizados entre si.

O ntp trabalha na porta 123 utilizando o protocolo udp.

A ferramenta de monitoramento do ntp é a ntpq, através dela é possível obter uma lista de servidores ntp, entre outras opções.

ntpq -p - Lista os peers de servidor ntp disponíveis .