Sistemas Operacionais
Linux para Concursos

Gustavo Pinto Vilar

ITnerante

PROVAS DE TI
TUDO PARA VOCÊ PASSAR

Gustavo Vilar – Mini CV

- PPF / DPF – Papiloscopista Policial Federal
- Pós-Graduado em Docência do Ensino Superior – UFRJ
- Graduado em Ciência da Computação e Processamento de Dados – ASPER/PB
- Aprovações: PRF 2002, PPF-PF 2004, PCF-PF 2004*, MPU 2010, ABIN 2010, PCF-PF 2013

Gustavo Vilar

- Contatos:

http://www.itnerante.com.br/profile/GustavoPintoVilar

http://www.provasdeti.com.br/index.php/por-professor/gustavo-vilar.html

gustavopintovilar@gmail.com

p3r1t0f3d3r4l@yahoo.com.br

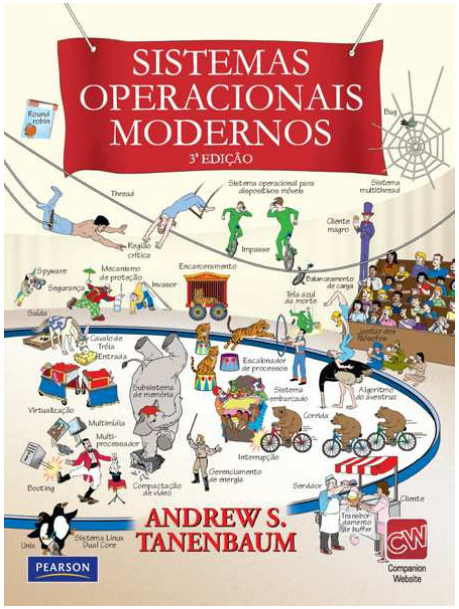
ITnerante

PROVAS DE TI
TUDO PARA VOCÊ PASSAR

Escopo

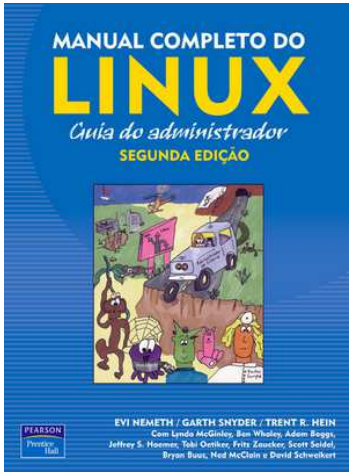
- Abordar os assuntos mais recorrentes e com fortes tendências para concursos atuais
- Familiarizar o concursando com os tipos de questões mais frequentes.
- Abordar as metodologias de resolução de questões das principais bancas

Bibliografia



Google

cespeUnB



Fundação Carlos Chagas

FUNDAÇÃO CESGRANRIO

ITnerante

PROVAS DE TI
TUDO PARA VOCÊ PASSAR

Linux para Concursos – Carga Horária

- **17 vídeo aulas (05h38m58s / 00h19m56s)**
  - Bateria de Questões de Diagnóstico Inicial
  - Definição, Distribuições, Versões do Kernel, Licenciamento, Características
  - Apresentação do SO, Shell, Inicialização
  - Processos, Escalonamento, Prioridades de Processos, Cópia de Processos
  - Compartilhamento de Recursos, Descritor de Processos, Criação e Ataques à Criação de Processos, Estados de um Processo
  - Inicialização, Init, Inittab
  - Gerenciamento de Memória, Algoritmo Companheiro, Algoritmo Alocador de Fatias
  - Descritores de Segurança – RWX
  - Gerenciamento de Usuários
  - Criação de usuários, arquivos passwd e shadow
  - Gerenciamento de usuários - principais comandos
  - Gerenciamento de processos - principais comandos
  - Quatro bateria de questões de aprendizagem

Linux para Concursos

Questões de Diagnóstico Preliminar

ITnerante

PROVAS DE TI
TUDO PARA VOCÊ PASSAR

1. No que se refere ao uso e ao funcionamento de sistemas operacionais modernos e suas características, julgue os itens seguintes.

[95] No Linux, durante a configuração para compilação de um novo kernel, é possível colocar os drivers de placas de rede diretamente no kernel ou como módulo de kernel.

2. Para o administrador do sistema SUSE Linux 11 listar a descrição dos usuários cadastrados no sistema na linha de comando do shell, em ordem alfabética, deve-se executar a linha de comando

A. cat /etc/passwd | cut -d: -f6 | sort

B. cat /etc/passwd | cut -d: -f5 | sort

C. cat /etc/shadow | cut -d: -f6 | sort

D. cat /etc/shadow | cut -d: -f5 | sort -r

E. cat /etc/users | cut -d: -f6 | sort –r

3. O administrador de um computador com sistema operacional Linux deseja saber quais são os usuários que estão "logados" àquele computador no momento. Para isso, ele pode utilizar o comando

A. ps
B. top
C. who
D. finger
E. whoami

4. Com base nas características do sistema operacional Linux, assinale a opção correta.

A. Por meio do comando cut, é possível extrair as últimas linhas de uma arquivo.

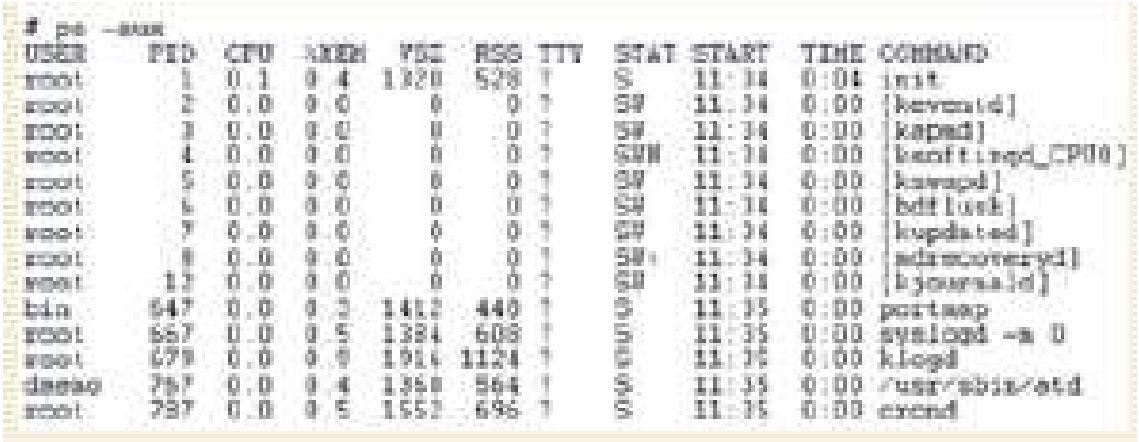
B. O núcleo do sistema Linux é dividido em dois componentes principais: o de gerenciamento de processos; e o de Entrada/Saída, que é responsável pela interação com os dispositivos de rede e armazenamento.

C. Em todo processo no Linux, há um espaço de endereçamento que consiste de dois segmentos: o segmento de código e o de dado. O segmento de código é o local de armazenamento de todas as variáveis do programa e o segmento de dado contém as instruções de máquina que formam o código executável do programa.

D. O sistema de arquivos Ext2 do Linux escreve, em um diário, de forma ordenada, todas as operações de alterações ocorridas em dados e metadados, visando melhoria de desempenho na gravação em disco.

E. As interfaces gráficas do Linux são executadas pelo sistema X Window.

5. Considere a figura a seguir sobre processos em execução de um sistema operacional Linux na sua configuração padrão e responda à questão.



```
# ps -aux
USER       PID  CPU  MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root         1  0.1  0.4   1820   520 ?        S    11:34   0:01 init
root         2  0.0  0.0      0     0 ?        SW   11:34   0:00 [keventd]
root         3  0.0  0.0      0     0 ?        SW   11:34   0:00 [kapmd]
root         4  0.0  0.0      0     0 ?        SWN  11:34   0:00 [ksoftirqd_CPU0]
root         5  0.0  0.0      0     0 ?        SW   11:34   0:00 [kswapd]
root         6  0.0  0.0      0     0 ?        SW   11:34   0:00 [bdflush]
root         7  0.0  0.0      0     0 ?        SW   11:34   0:00 [kupdated]
root         8  0.0  0.0      0     0 ?        SW<  11:34   0:00 [adamcoveryd]
root        12  0.0  0.0      0     0 ?        SW   11:34   0:00 [kjournald]
bin        647  0.0  0.3   1412   440 ?        S    11:35   0:00 portmap
root       667  0.0  0.5   1384   608 ?        S    11:35   0:00 syslogd -a 0
root       679  0.0  0.9   1914  1124 ?        S    11:35   0:00 klogd
daemon     767  0.0  0.4   1360   564 ?        S    11:35   0:00 /usr/sbin/atd
root       787  0.0  0.5   1552   696 ?        S    11:35   0:00 crond
```

As opções a, u e x utilizadas no comando são responsáveis, respectivamente, pelos processos

A. criados, processos que são controlados pelo terminal, nome do usuário e a hora do processo.

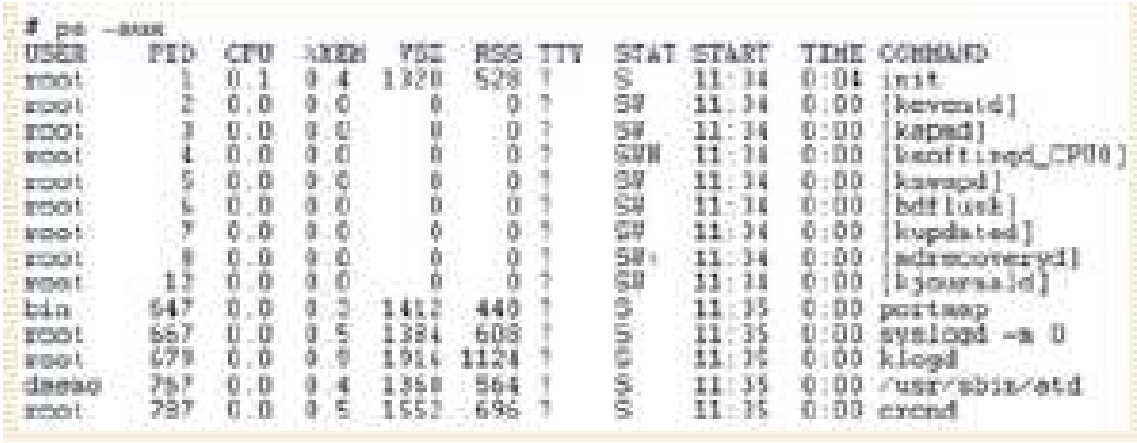
B. terminados, processos que não são controlados pelo terminal, hora do processo.

C. terminados, processos que são controlados pelo terminal, hora do processo.

D. criados, processos que são controlados pelo terminal, hora do processo.

E. criados, processos que não são controlados pelo terminal, nome do usuário e a hora do processo.

6. Considere a figura a seguir sobre processos em execução de um sistema operacional Linux na sua configuração padrão e responda à questão.



```
# ps -aux
USER       PID  CPU  MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root         1  0.1  0.4   1320   520 ?        S    11:34   0:01 init
root         2  0.0  0.0      0     0 ?        SW   11:34   0:00 [keventd]
root         3  0.0  0.0      0     0 ?        SW   11:34   0:00 [kapmd]
root         4  0.0  0.0      0     0 ?        SWN  11:34   0:00 [ksoftirqd_CPU0]
root         5  0.0  0.0      0     0 ?        SW   11:34   0:00 [kswapd]
root         6  0.0  0.0      0     0 ?        SW   11:34   0:00 [bdflush]
root         7  0.0  0.0      0     0 ?        SW   11:34   0:00 [kupdated]
root         8  0.0  0.0      0     0 ?        SW<  11:34   0:00 [adamcoverxyd]
root        12  0.0  0.0      0     0 ?        SW   11:34   0:00 [kjournald]
bin        547  0.0  0.3   1412   440 ?        S    11:35   0:00 portmap
root       557  0.0  0.5   1384   608 ?        S    11:35   0:00 syslogd -m 0
root       579  0.0  0.9   1914  1124 ?        S    11:35   0:00 klogd
daemon     757  0.0  0.4   1360   564 ?        S    11:35   0:00 /usr/sbin/atd
root       787  0.0  0.5   1552   696 ?        S    11:35   0:00 crond
```

As colunas RSS, TTY e STAT demonstram, respectivamente,

A. o terminal onde são executados os processos, Soma total da memória física usada pelo processo, Estado do processo.

B. o terminal onde são executados os processos, Tempo total da CPU, Estado do processo.

C. o terminal onde são executados os processos, Tamanho do código da tarefa, Estado do processo.

D. a soma total da memória física, Terminal onde são executados os processos, Estado do processo.

E. a soma total da memória física, Nome do comando do processo, Estado do processo.

7. Quanto ao sistema operacional Linux, marque V para verdadeiro ou F para falso e, em seguida, assinale a alternativa que apresenta a sequência correta.

( ) O init é o primeiro processo inicializado no Linux e é o pai de todos os outros processos.

( ) Se um processo termina e deixa processos-filho ainda executando, o processo init assume a paternidade desses processos.

( ) Quando um usuário trabalha no modo monousuário, um único processo shell é inicializado.

( ) A árvore hierárquica dos processos, tendo o shell como raiz, é chamada de sessão.

A. F/ V/ F/ F
B. F/ F/ V/ F
C. V/ V/ F/ F
D. V/ V/ V/ V
E. F/ V/ F/ V

8. Um comando muito utilizado em distribuições Linux, permite que sejam alteradas as informações de propriedade de usuário e grupo para um determinado arquivo ou diretório, aplicando, inclusive, essas alterações de forma recursiva. O comando em questão, em conjunto com o atributo de recursividade é corretamente exposto em

A. usermod -S
B. chmod --dereference
C. ln --recursive
D. chown -R
E. chggrp -S

9. No Red Hat Linux, há três tipos diferentes de permissões para arquivos, diretórios e aplicações. Estas permissões são usadas para controlar os tipos de acesso permitidos. São usados símbolos diferentes de caractere para descrever cada permissão em uma listagem de diretórios. São usados: r para a permissão de leitura, w para a permissão de escrita e, para a permissão de execução de um arquivo, é atribuída a letra

A. e.
B. x.
C. p.
D. a.
E. l.

10. Arquivos em Linux são protegidos atribuindo-se a cada um deles um código de proteção de 9 bits. O código de proteção consiste em campos de 3 bits, um grupo para qualquer usuário, outro para o usuário do arquivo e um para o grupo ao qual o usuário pertence. Cada campo possui um bit de permissão de leitura, um bit de permissão de escrita e outro de permissão de execução. Por exemplo, o código de proteção de um arquivo definido como “-wxr-xr--" significa que:

A. membros do grupo e o proprietário podem ler, executar e escrever no arquivo e outros usuários podem apenas ler.

B. membros do grupo podem escrever e executar o arquivo, qualquer usuário pode ler e executar o arquivo e o dono do arquivo pode apenas ler o conteúdo do arquivo.

C. qualquer usuário pode escrever e executar o arquivo, o proprietário pode ler e executar o arquivo e membros do grupo podem apenas ler o arquivo.

D. o proprietário pode escrever e executar o arquivo, membros do grupo podem ler e executar o arquivo e qualquer usuário pode ler o arquivo.

E. o proprietário pode ler, escrever e executar o arquivo, membros do grupo podem ler e escrever no arquivo e qualquer usuário pode ler e executar o arquivo.

GABARITO

1. C
2. B
3. C
4. E
5. E
6. D
7. D
8. D
9. B
10. D

Sistemas Operacionais

Linux para Concursos

Linux - História

- Desenvolvido em 1991 por Linus Torvalds
- Baseou-se no Minix que, por sua vez, se baseou no Unix
- A intenção de Torvalds era a de fazer com que o projeto rodasse em um 386
- Em 1991, Linus Torvalds decidiu divulgar abertamente o seu projeto
- O projeto se expandiu pelo mundo

Linux - Definição

- Kernel de código fonte aberto
  - Núcleo do sistema operacional
- Qualquer pessoa ou organização pode juntá-lo a um conjunto de softwares para criar um sistema operacional customizado: Distribuições Linux
- Mantido basicamente pela colaboração voluntária de desenvolvedores de várias partes do mundo.

Linux - Definição

- Há várias distribuições Linux
  - Muitas fazem parte de negócios rentáveis, onde a empresa fornece gratuitamente o sistema operacional e obtém receita a partir de serviços de suporte técnico
- Distribuições do segmento de usuários domésticos são mais populares: Ubuntu
  - São lançadas novas versões do Ubuntu em todos os meses de abril e outubro de cada ano

Linux - Distribuições

- Fedora (ligada à Red Hat);
- Debian;
- Mandriva;
- Linux Mint;
- CentOS (com foco em servidores);
- Slackware.

Linux – Versões do Kernel

- Periodicamente, novas versões do kernel Linux são lançadas
- Cada versão do kernel é representada por três números distintos separados por pontos
  - Ex: 2.6.24
    - versão do kernel
    - última revisão feita
    - revisão menor

Linux – Versões do Kernel

- Antes da série 2.6.x, a numeração do kernel funcionava da seguinte forma: se o segundo número da representação fosse ímpar, significava que aquela série era uma versão instável e em fase de testes ou aperfeiçoamentos.
- Se o número fosse par, significava que aquela série já tinha estabilidade para ser disponibilizada para uso

Linux – Licenciamento

- O Linux utiliza a *GPL* (*GNU Public Licence*).
- Criada pela Free Software Foundation (Richard Stallman)
- A GPL surgiu em 1989, mas foi revisada em 1991 para atender a determinadas necessidades, resultando na GPLv2 (GPL versão dois). Em 2007, surgiu a GPLv3 (GPL versão três)

Linux – Liberdades GPL

0. Liberdade de executar o programa, para qualquer propósito;

1. liberdade de estudar como o programa funciona e adaptá-lo às suas necessidades, sendo o acesso ao código-fonte um pré-requisito para este aspecto;

2. liberdade de distribuir cópias de forma que você possa ajudar ao seu próximo;

3. liberdade de melhorar o programa e liberar os seus aperfeiçoamentos, de modo que toda a comunidade se beneficie.

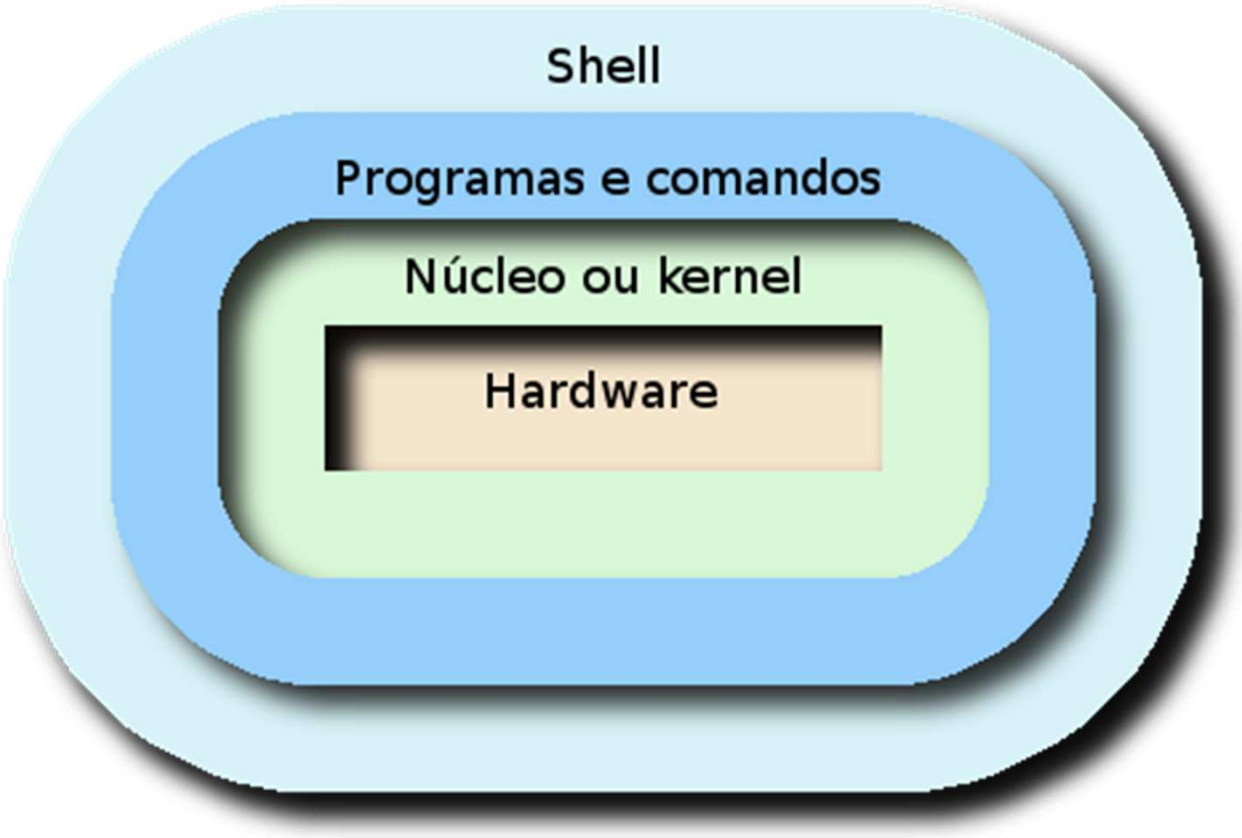
Linux – Características

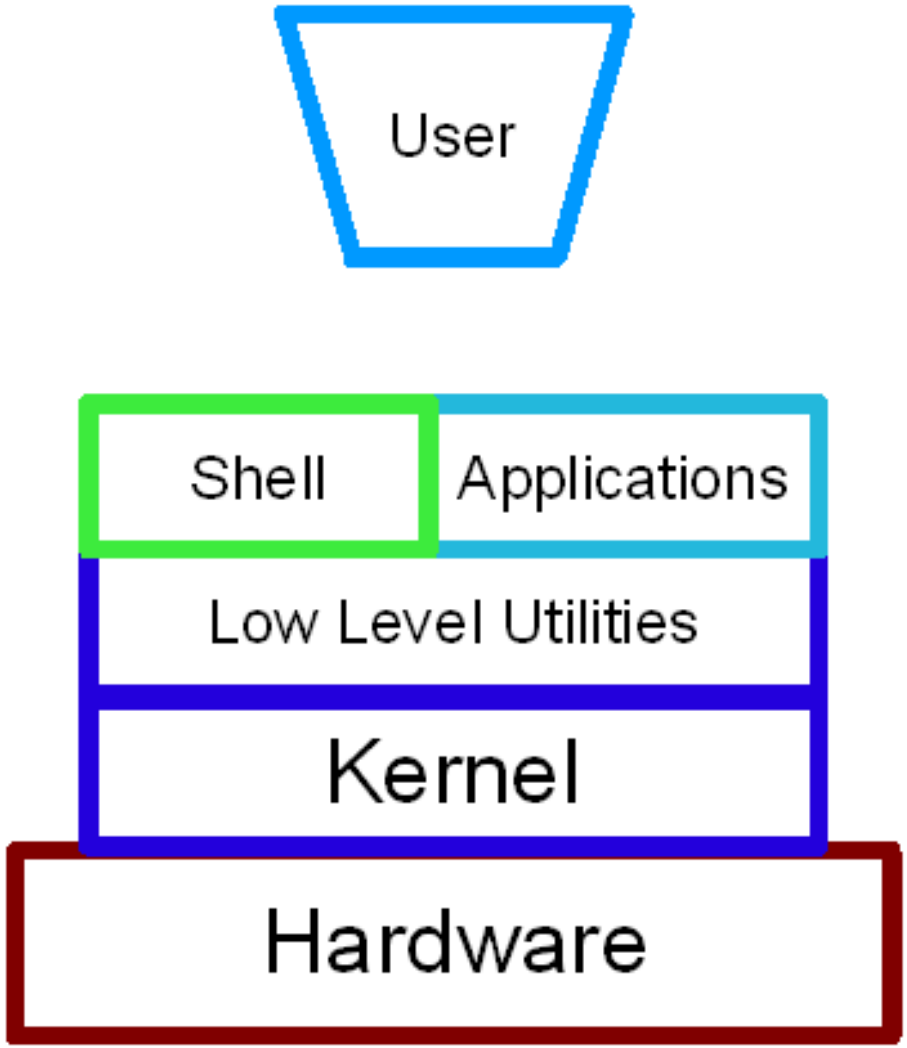
- Projetado por e para programadores
- Utiliza o princípio da surpresa mínima
- Todo programa deve fazer somente uma única coisa e fazer bem feito: Coesão
- Interface de linha de comando é mais natural ao usuário avançado do que ao usuário iniciante
  - Versões PC são mais comuns com interfaces GUI

Linux Ubuntu

Linux – Shells

- Shell é um programa que permite ao usuário interagir com o sistema operacional através de comandos digitados do teclado
  - Sh, Bash, etc...
  - Para saber qual o shell: echo $SHELL
- São apenas programas comuns que rodam no espaço do usuário
- Quando inicia-se o shell, a ele são fornecidos a ele a entrada padrão (teclado), saída padrão (monitor) e erro padrão (monitor)
  - É possível modificá-los e redirecioná-los (inclusive no mesmo comando)
    - >, <, 2>





Linux – Inicialização Linux

Inicializadores:

LInux LOader

Grub

1. Computador é ligado
2. Primeiro setor do disco de boot é lido (MBR) para memória ALTA e executado
   - Setor de 512 bytes que contém programa independente "boot" (bootstrap)
   - 446 bytes - bootstrap
   - 66 bytes - tabela de partições
   - Pode apontar para o setor zero do SO.
     - Este setor zero prossegue no processo de inicialização
3. Boot lê diretório raiz do dispositivo de boot
4. Lê o núcleo do S.O. e transfere o controle para ele (Boot sai de cena)

Linux – Inicialização Linux

5. Alocação das estruturas de dados do núcleo. Sistema começa a autoconfiguração (hardware e drivers)
   - Linux carrega os drivers de dispositivos dinamicamente
6. Depois do hardware configurado, Processo 0 continua a inicialização, a montagem do sistema de arquivos raiz, do processo 1 (init) e do processo 2 (daemos de paginação)
   - Init está relacionado com a inicialização e finalização dos processos
7. Init checa o arq de configuração INITTAB para saber se é...
   - Monousuário
     - Cria um processo que executa o shell e espera seu término
   - Multiusuário
     - Cria um processo para executar um script de shell (/etc/rc)
8. Lê /etc/ttys que relaciona os terminais e suas propriedades
9. Aguarda o Login

Linux – Processos

- Programa em execução
  - Vai para a memória do mesmo jeito que é em disco
  - Processo rodando em segundo plano: Deamon
- Comunicação
  - Via Pipe
  - Via Interrupção de software
    - Estes sinais são enviados somente para os membros de seu grupo de processo
      - Ancestrais
      - Irmãos
      - Descendentes

Linux – Escalonamento

- Baseado em threads em vez de processos
  - Todos os threads são de núcleo
- Classes
  - FIFO em tempo real
    - Maior prioridade
    - Só são interrompidos por eles mesmos
  - Round Robin em tempo real
    - Iguais ao FIFO de tempo real, exceto pelo fato de poderem ser interrompidos pelo relógio
  - Round Robin

Linux – Escalonamento

- Observações
  - O nome tempo real é uma nomenclatura herdada, não havendo prazos nem garantias
  - Prioridade pode ser alterada pelo comando nice (1 até 40)
  - Nice: EXECUTA programa com determinada prioridade
  - Renice: ALTERA a prioridade de um programa em execução
- Além da prioridade, os threads possuem um quantum

Linux – Prioridades dos processos

- São 140 - 0 a 139
  - Escalonador Linux possui 140 filas diferentes de escalonamento
  - Prioridades diferentes = fatias de tempo diferentes
- Algoritmo de filas múltiplas
  - 0 a 99 tempo real
  - 100 a 139 - tempo compartilhado
- Prioridade estática = 120 + nice (-20 a +19)
  - Somente Adm pode elevar a prioridade
  - Usuário comum só pode rebaixar a prioridade
  - nice padrão é "0“
- Prioridade Dinâmica
  - Recompensa Threads interativos (até -5)
  - Pune threads que controlam a CPU (até +5)
- Controle em nível de CPU
  - Um vetor por CPU
  - 140 entradas por vetor

Linux – Cópia dos processos

- Cópia é um processo custoso e lento por natureza
  - Cópia da área de binário + dados
- Solução - Copy on write
  - Não copia o binário
    - segmento de código não cresce, não diminui nem se altera de nenhuma maneira
  - Copia apenas a parte da área de dados alterada para outra área de memória, que cresce aos poucos
    - Inicializados e não inicializados
  - Custo de criação de novos processos é muito pequeno
-

Linux – Compartilhamento de Recursos

- Processos
  - Cópia completa de recursos
- Threads
  - Compartilhamento de recursos
- LW Process
  - Compartilhamento de ALGUNS recursos (os não alterados)
  - Aproximação do Linux para resolver o problema do bloqueio e criação de threads
  - Todo processo no Linux é um LWProcess

Linux – Descritor de Processos

- PID
  - Numero de identificação do processo
  - Máximo é 2^15
  - Quando atingido não cria mais nenhum processo
  - São recicláveis: 0 a 32.767 (32768-1)
- Relações familiares entre processos
  - Pai: Processo iniciador
  - Filho: Processos chamados pelo pai; cópias ou outros
  - Irmão/siblings: processos filhos de um mesmo pai
- Todo processo tem pai
- Nem todo processo tem filho, nem irmãos

Linux – Criação de Processos

- Fork ()
  - Função usada para criar processos
  - Processo chamador passa para o modo núcleo
  - Pode ser cópia do processo (fork) ou a chamada para outro processo (fork + exec)
  - Retorna 0 para o filho e o PID do processo filho (<>0) para o pai
    - Processo pai recebe o PID do filho
    - Útil na construção de árvores de processos e controle de pendências de tarefas

Linux – Ataques à Criação de Processos

- Fork Bomb
  - Criação indefinida de processos
  - 2^15 (0 a 2^15-1)
  - Mesmo que a tabela não seja lotada, o escalonamento round-robin de processos fica inviável

Linux – Estados dos Processos

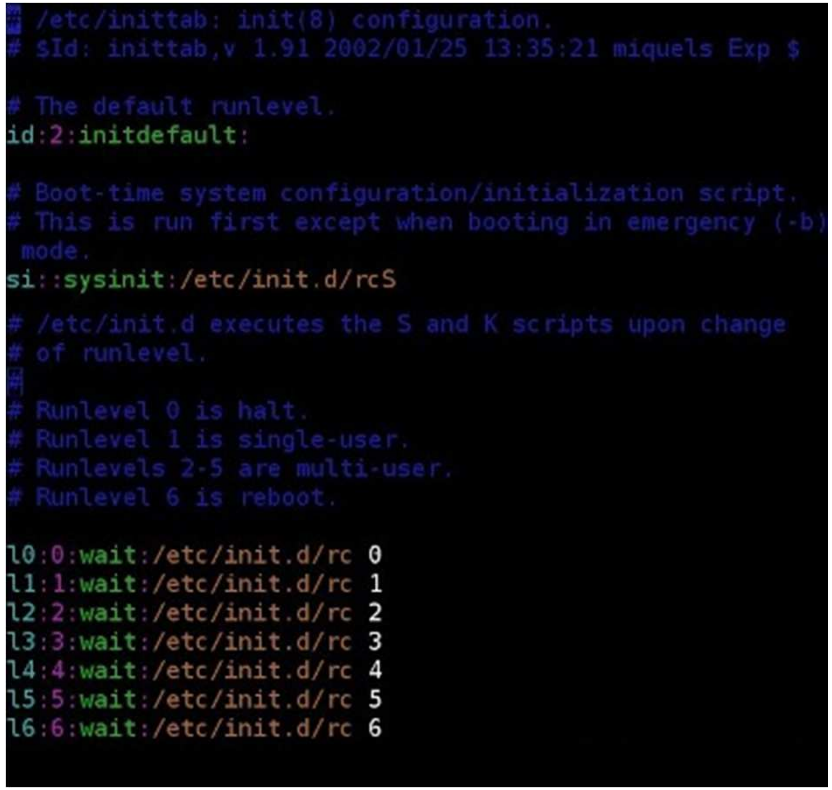
- **Task Running (R)**
  - Kernel Preemptivo: O próprio kernel cuida do escalonamento
  - pronto para usar ou usando o processador
- **Task Interruptible (S)**
  - Pode ser morto enquanto aguarda um retorno
  - Aceita sinais (kill)
- **Task Uninterruptible (D)**
  - Não morrem
  - Ignoram o sinal
  - Starvation
  - Geralmente em operações com hardware
- **Exit Zombie (Z)**
  - Processo terminado e aguardando finalização pelo processo pai
  - captura dos dados do filho
  - O processo pai ainda pode estar caminhando para o wait(), enquanto o filho já terminou
  - Se o pai morrer, init passa a ser o pai
    - Para o processo morrer ele precisa da permissão do pai
- **Exit Dead**
  - Processo terminado e finalizado
- **Task Stopped (T)**
  - Processo explicitamente pausado
- **Task Traced**
  - Processo sendo depurado

Linux –Processo init

- PID 1
  - Criado pelo Kernel
- Processo pai de quase todos
- Adota os processos filhos dos processos terminados abruptamente
- Caçador de Zumbis por adoção
  - processo cujo pai morreu
  - Ignora as informações dos filhos adotados
- Não morre enquanto a máquina estiver em execução
- Primeiro a iniciar e Último a morrer

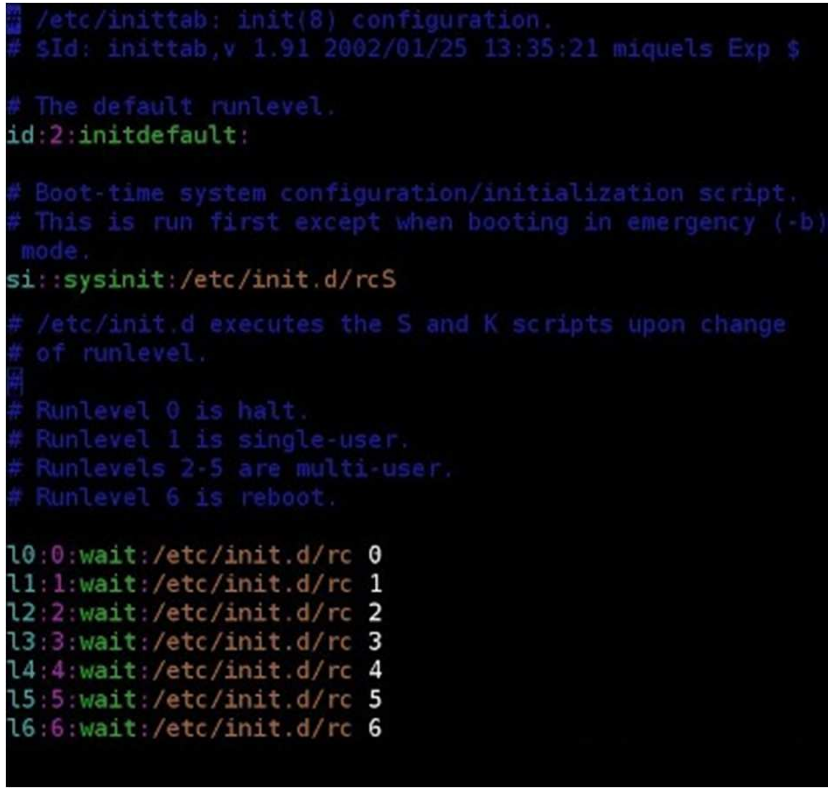
Linux – Níveis de inicialização

- Níveis
  - 0 – sistema completamente desligado
  - 1 ou S – representa o modo monousuário – S é o nível 1 com o acréscimo da senha de root para inicialização como tal.
  - 2 ou 3 – multiusuário normal.
    - 2 - sem rede
    - 3 - com rede
  - 4 – raramente usado.
  - 5 – multiusuário mais o Xwindows
  - 6 – reinicialização.
- O linux suporta até 10 níveis de inicialização, mas os níveis de 7 a 9 são indefinidos.



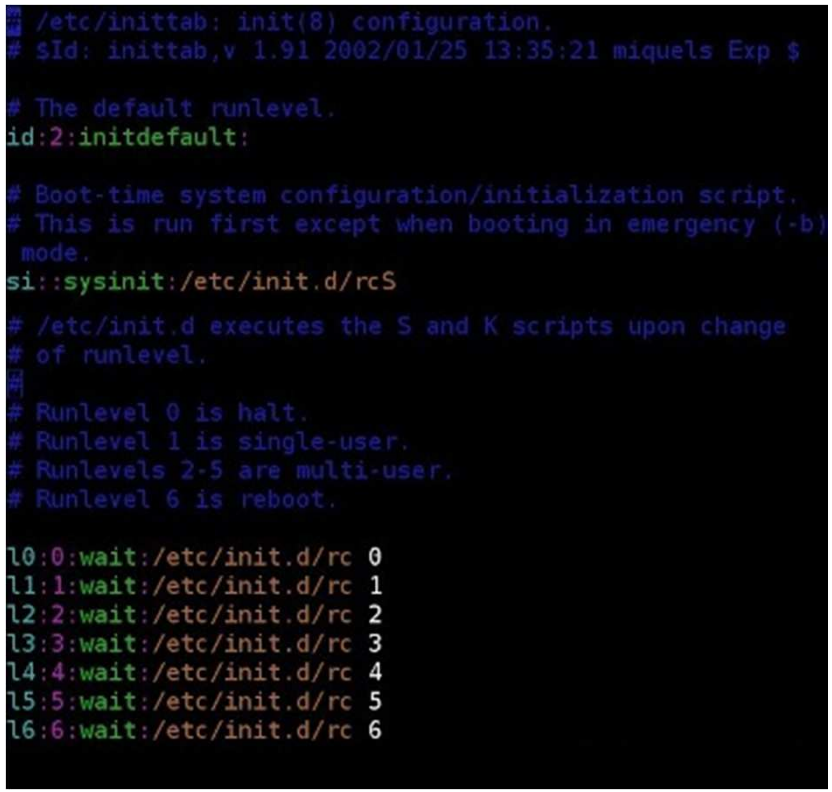
Linux – Mais sobre inicialização

- Start x Kill
  - Os diretórios rc<nível>.d contêm tipicamente links simbólicos que apontam de volta para os scripts contidos no diretório init.d.
  - O último rc a ser chamado é o rc.local
  - Os nomes desse links simbólicos começam com S (start) ou K (kill), seguidos por um número e o nome do serviço que o script controla.
  - Executa-se os scripts iniciados com K, depois os iniciados com S
  - Os números servem para ordenar os serviços que serão executados. Ex S34named e S20network.



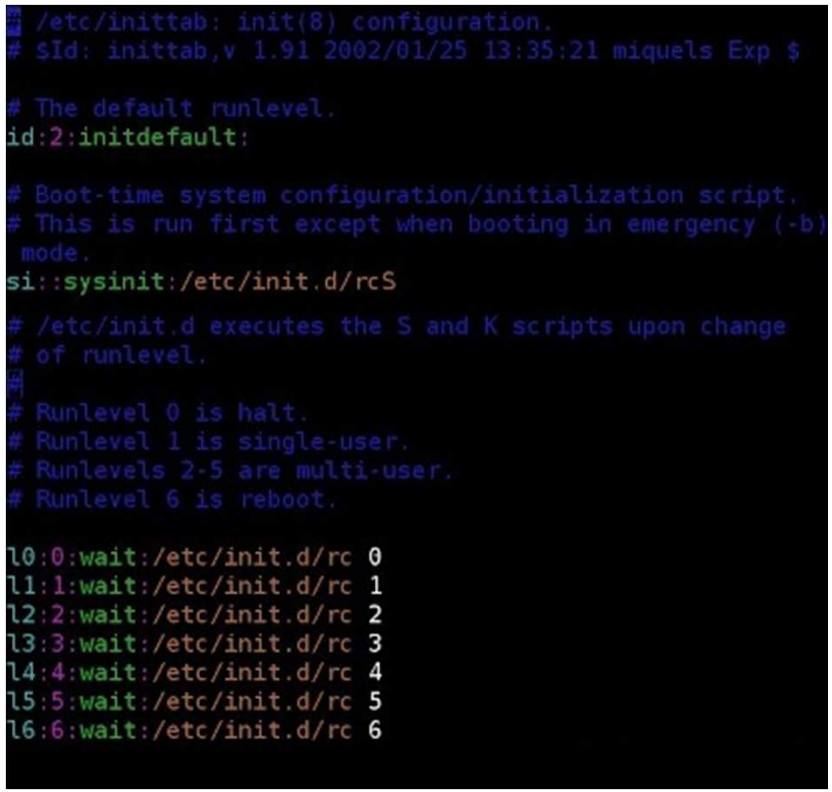
Linux – /etc/inittab

- Descreve quais processos serão iniciados no boot de acordo com os níveis de execução
  - id:runlevels:action:process
- **Id**: Identificador da linha: 1 - 4 caracteres
- **Runlevels**: Nível de execução que a ação será processada. Pode conter mais de um nível ao mesmo tempo. Ex: 12345
  - É ignorado em algumas ações (sysinit, boot, bootwait)



Linux – /etc/inittab

- **Action**: Informa ao init o que fazer com a entrada, cujos tipos são:
  - Initdefault: interpreta o campo runlevel como sendo o nível default
  - Sysinit: processo executado durante o boot do sistema
  - Wait: executa o processo e aguarda até que seja encerrado
  - Ctrlaltdel: executa o processo após o recebimento de um sinal das teclas CTRL+ALT+DEL
  - Respawn: em caso de encerramento o processo será reiniciado

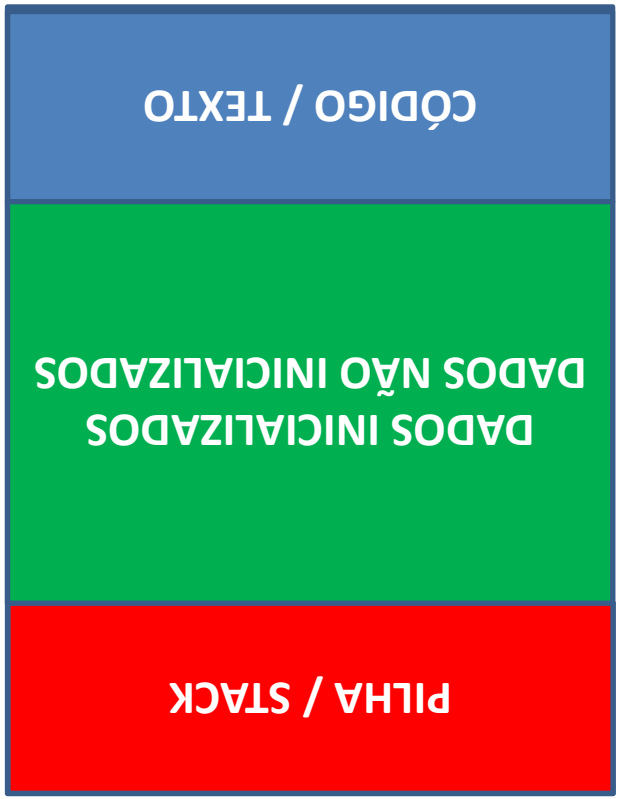


Linux – /etc/inittab

- **Once**: o processo é executado apenas uma vez
- **Process**: Nome do script ou programa iniciado pelo init
  - Ex: /etc/init.d/rc <nível>
- Nas distribuições derivadas do ubuntu não existe o inittab, mas sim outro conceito baseado em eventos chamado Upstart

Linux – Gerência de Memória

- Processo dividido em...
  - **Código (Executável) / texto**
    - Não pode ser alterado
    - Sistemas unix suportam os segmentos de código compartilhados
  - **Dados**
    - Inicializados
    - Não inicializados / BSS / Block Started by Symbol
    - Pode ser alterado
    - Nunca são compartilhados, exceto após um fork ()
  - **Pilha**
    - Inicia no topo do espaço e cresce “para baixo”
    - Nunca são compartilhados, exceto após um fork ()



Linux – Gerência de Memória

- Cada processo “possui" 4 GB de espaço de endereçamento (x86)
  - 3 GB para modo usuário (em espaço de 32 bits)
  - 1 GB para modo núcleo (em espaço de 32 bits)
    - Memória dividida em zonas
      - DMA: até 16 MB
      - Normal: após 16MB até 896MB
      - HIMEM: A partir de 896 até 1 GB
        - » Não é visível quando o processo executa no modo usuário
        - » Acessível quando o processo faz chamadas ao núcleo

Linux – Gerência de Memória

- Núcleo fica TOTALMENTE residente na memória
- Memória dividida em 3 partes sendo usada por:
  - Núcleo
    - Região fixa
  - Mapa de memória
    - Região fixa
  - Restante
    - Região não fixa

Linux – Gerência de Memória

- Caches (Em RAM)
  - Memória física foi feita para ser usada
    - O que não é usado pelos processos pode ser usado para cache
  - Buffer cache
  - Page cache
  - Swap cache
  - Volátil

Linux – Gerência de Memória

- Uso do algoritmo companheiro
  - Solicitação arredondada para potência de 2 mais próxima
  - Memória é encontrada e dividida na sequência até atingir o tamanho requerido arredondado para mais
  - A desmontagem é o inverso da montagem
  - Gera considerável fragmentação interna
    - Ex: 65 bytes = 128 bytes
  - Para driblar essa deficiência é usado um segundo algoritmo de alocação chamado ALOCADOR DE FATIAS

Linux – Descritores de Segurança

- UID – User ID
  - 2^16 possibilidades
  - Arquivos e processos são marcados com o UID
  - Um usuário deve pertencer a pelo menos um grupo
  - Um usuário pode pertencer a mais de um grupo
    - Grupo principal armazenado em passwd
    - Demais grupos em etc/group
  - Igual a 0 - root

Linux – Descritores de Segurança

- GID – Group ID
  - 2^16 possibilidades
  - Associação feita manualmente pelo administrador
  - Grupo padrão serve para gerar atributos na criação de objetos
  - Os demais grupos servem para proporcionar acesso aos objetos

Linux – Descritores de Segurança

- Cada processo carrega o GID e UID de seu proprietário, bem como o rwx determinados pelo processo criador
- Diretórios
  - R - Listagem, VER conteúdo SEM ENTRAR = comando ls
  - W - Criação ou exclusão
  - X - Acesso, ENTRAR, comando CD

Linux – Descritores de Segurança

- Os bits próprios de proteção só podem ser alterados pelo proprietário e pelo root
- Se o modo de proteção é alterado enquanto o arquivo está aberto, os processos que já estão com o arquivo aberto não são afetados
- Modo de proteção do diretório modificado, implica em reflexo imediato no diretório

Linux – Gerenciamento de Usuários

– Usuários comuns

- Podem se conectar
- Possuem um diretório base
- /home/usuário

– Usuários de sistema

- Não podem se conectar
- São contas usadas para propósitos específicos do sistema e não são de propriedade de uma pessoa em particular
- Geralmente associados a serviços
- Ex. **nobody**: responsável, normalmente, por manipular as solicitações HTTP. **lp:** manipula solicitações de impressão

– Root

- Tem controle total sobre todo o sistema operacional
- Qualquer conta com uma identificação de usuário igual a 0 em /etc/passwd é um usuário root.

Linux – Gerenciamento de Usuários

- Informações sobre usuários ficam são distribuídas em três arquivos/sistemas diferentes:
  - /etc/passwd
    - Usuário e grupo primário
  - /etc/shadow
    - Senha
  - /etc/group
    - Grupos secundários

Linux – Gerenciamento de Usuários

- passwd
  - Relaciona as informações principais sobre os usuários
  - Relaciona o grupo primário do qual o usuário faz parte.
  - Os usuários são cadastrados no sistema através deste arquivo
  - O próprio usuário pode alterar a senha usando o comando "passwd"
    - Usuário precisa saber a senha antiga
    - Root não precisa saber a senha

Linux – Gerenciamento de Usuários

- Shadow
  - Sistema shadow, onde as senhas são armazenadas de forma encriptada em um arquivo separado, o "/etc/shadow"
    - As senhas são encriptadas usando um algoritmo de mão única
      - » Senha pode ser mudada, mas nunca recuperada
      - » 1 indica MD5
      - » 5 6 indica SHA 1 ou 2
      - » NP ou ! ou null = conta sem senha
      - » LK ou * - Conta bloqueada
      - » !! - Senha expirada

Linux – Gerenciamento de Usuários

- O comando mais básico é o "adduser" e o arquivo /etc/passwd
  - Criar usuários
  - Adicionar os usuários desejados ao grupo
  - Os usuários são cadastrados no sistema através do arquivo "/etc/passwd“
- Atenção: Via de regra, Instruções que se iniciam por um verbo na verdade são scripts. A mesma instrução iniciada pelo substantivo é o comando propriamente dito
  - ADDUSER <> USERADD

Linux – Gerenciamento de Usuários

- **Estrutura do passwd**
  - Nome de login
  - Senha criptografada ou marcador de posição da senha
  - UID
  - GID primário/padrão
  - Informações GECOS (*General Electric Comprehensive Operating System)*
  - Diretório inicial (home)
  - Shell de Login

Linux – Gerenciamento de Usuários

- **Estrutura do shadow**
  - **Nome de login**
  - **Senha criptografada**
  - Data da última mudança de senha
  - Número mínimo de dias entre mudança de senha
  - Número máximo de dias entre mudança de senha
  - Número de dias antecipados para alertar os usuários sobre a expiração de suas senhas
  - Número de dias após a expiração da senha que a conta será desabilitada
  - Data de expiração da conta

- OBS: Dias a contar de 1970

Linux – Gerenciamento de Usuários

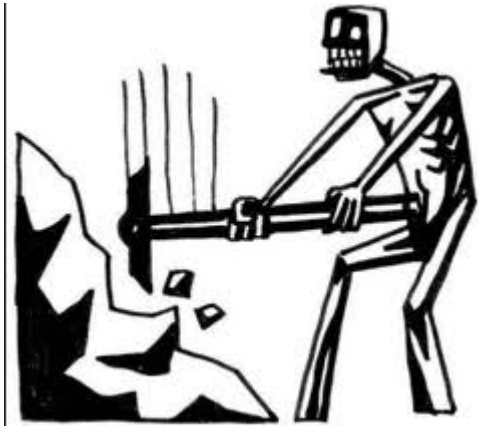
- Comando adduser / useradd
  - Usuário criado = grupo com mesmo nome + home do usuário (exceto usuário do sistema)
  - UID (primeira disponível na faixa de IDs em /etc/adduser.conf.
  - Usuário pertence a pelo menos um grupo, nunca a nenhum grupo. Exigência do Kernel.
  - Dados vão para /etc/passwd e para /etc/shadow.
  - O arquivo de /etc/shadow é configurado para não ser lido por qualquer um.
  - Somente o root poderá ler e escrever no arquivo /etc/shadow.

Linux – Gerenciamento de Usuários

- Remover um usuário - "deluser"
  - Por questão de segurança, o comando remove apenas a conta, sem apagar o diretório home, ou outras pastas
  - Pode-se bloquear temporariamente um usuário, sem remover o /home ou qualquer outro arquivo usando o comando "passwd -l" / "passwd -u" para reverter

Gerenciamento de Usuários

- Instalar um ambiente Linux
  - Outra máquina
  - Máquina virtual
- Treinar os comandos mostrados a seguir.
- Identificar as diferenças de sintaxes e resultados obtidos



Linux – Principais Comandos no Gerenciamento de Usuários

- **adduser** - Adiciona um usuário ou grupo no sistema
- **addgroup** – Adiciona um novo grupo de usuários no sistema
- **passwd** - Modifica a parâmetros e senha de usuário. Um usuário somente pode alterar a senha de sua conta, mas o superusuário (root) pode alterar a senha de qualquer conta de usuário, inclusive a data de validade da conta, etc. Os donos de grupos também podem alterar a senha do grupo com este comando.
  - SU altera de todos, entretanto o usuário só altera a dele
  - Donos de grupos podem alterar a senha do grupo
  - Nome, endereço, telefone, também podem ser modificados com este comando.
  - Não confundir com PWD - lista diretório corrente
  - O próprio usuário pode alterar a senha usando o comando "passwd"
  - Usuário comum precisa saber a senha antiga
  - Root não precisa saber a senha

Linux – Principais Comandos no Gerenciamento de Usuários

- **gpasswd -** Modifica parâmetros e senha de grupo. Um usuário somente pode alterar a senha de seu grupo, mas o superusuário (root) pode alterar a senha de qualquer grupo de usuário, inclusive definir o administrador do grupo.
- **newgrp -** Altera a identificação de grupo do usuário. Para retornar a identificação anterior, digite exit e tecle Enter. Para executar um comando com outra identificação de grupo de usuário

Linux – Principais Comandos no Gerenciamento de Usuários

- **groupdel** – apaga um grupo do sistema. OBS: Você não pode remover o grupo primário de um usuário. Remova o usuário primeiro.
- **groups** – Exibe os grupos aos quais o usuário pertence
- **userdel** – Apaga usuário do sistema (-r) apaga também o home
  - Caso o usuário esteja no sistema, a conta não poderá ser removida
- **su** – Substituir usuário sem fazer logout
  - Se o comando vier do root não é preciso indicar a senha do usuário no parâmetro.

Linux – Gerenciamento de Usuários

- **sudo** – Substituir/trocar usuário sem fazer logout, somente por um comando
  - Linkado com o arquivo /etc/sudoers, que armazena quem pode fazer sudo
  - vantagens do sudo:
    - Limitar acesso de root;
    - Não necessita que todos os sysadmins tenham acesso a senha de root;
    - Controlar os comandos usados pelos sysadmin (auditar);
    - Possibilidade de criação de users para trabalhos específicos, como: user de backup, user de contas e etc... .

Linux – Gerenciamento de Usuários

- **whoami**
  - Informar qual é o usuário
- **who**
  - Exibir os usuários logados no sistema.
- **chage**
  - Modifica as informações de expiração de senha de usuários

Linux – Gerenciamento de Usuários

- **id**
  - Retorna a identificação atual do usuário e grupos aos quais pertence ou passados por parâmetro
- **logname**
  - mostra seu login
- **chown**
  - Somente o root ou dono do arquivo podem alterar o usuário proprietário ou o grupo
- **chgroup**
  - Somente o root ou dono do grupo podem alterar o usuário proprietário
- **usermod**
  - Modifica um usuário do sistema
  - o "/etc/passwd" também pode ser editado manualmente

Linux – Gerenciamento de Usuários

- **gpasswd**
  - Modifica parâmetros e senha de grupo
  - Um usuário somente pode alterar a senha de seu grupo, mas o superusuário (root) pode alterar a senha de qualquer grupo de usuário
- **newgrp**
  - Altera a identificação de grupo do usuário
  - Para retornar a identificação anterior, digite exit e tecle Enter
- **lastlog**
  - Mostra o último login dos usuários cadastrados no sistema
  - É mostrado o nome usado no login, o terminal onde ocorreu a conexão e a hora da última conexão
  - Estes dados são obtidos através da pesquisa e formatação do arquivo /var/log/lastlog

Linux – Gerenciamento de Usuários

- **last**
  - Mostra uma listagem de entrada e saída de usuários no sistema
  - A listagem é mostrada em ordem inversa, ou seja, da data mais atual para a mais antiga
- **sg**
  - Análogo ao su
  - Executa um comando com outra identificação de grupo
- **chfn**
  - Altera o campo GECOS

Linux – Gerenciamento de Usuários

- **logname**
  - Mostra seu login (username).
- **users**
  - Mostra os nomes de usuários usando atualmente o sistema
  - obtidos do arquivo /var/log/wtmp
- **history**
  - Mostrar os últimos comandos executados pelo usuário.
- **pwck**
  - Verifica a integridade do arquivo de senhas

Linux – Gerenciamento de Usuários

- **chmod**
  - Ajustar as permissões dos arquivos e pastas
- Permissões especiais de arquivos:
  - **Sticky bit/flag**
    - No ext2, ext3 e ext4 tem uma funcionalidade chamada Sticky Bit - que é usada para compartilhamento de diretórios. Essa opção habilitada impede que usuários apaguem arquivos que não foram criados por ele mesmo! Para habilitar essa opção é só invocar o "chmod" usando a letra "t" ou o número "1"
      - diretório /tmp
    - número 1
    - chmod o+t

Linux – Gerenciamento de Usuários

- **GUID bit /flag**
  - Diretório
    - todos os arquivos que forem criados no respectivo diretório possuirão como grupo padrão o grupo do diretório e não do usuário
  - Executável
    - possuirá as permissões atribuídas ao grupo do arquivo e não ao grupo do usuário
  - número 2
  - chmod g+s teste.sh

Linux – Gerenciamento de Usuários

- **SUID bit /flag**
  - Serve para que quando executado um arquivo ele funcione com as permissões do proprietário e não do usuário que está executando. Ex. de atribuição: chmod 4777 teste/
    - comando passwd
  - número 4
  - chmod u+s teste.sh

Linux – Gerenciamento de Usuários

- **Resumindo o chmod**
  - "U G O/A"
  - Root continua tendo privilégios nos diretórios, mesmo quando retirados os direitos
  - Usuário, apesar de não poder entrar, poderá modificar as permissões, caso seja o proprietário
  - + para adicionar, - para retirar
  - Pode-se usar flags de 1 a 7 para configurar as permissões em vez de rwx
  - O bit de permissão especial antecede os demais, totalizando 4 bits

Linux – Comandos relacionados com processos

- **ps**
  - **mostrar os processos em execução (snapshot)**
- **pstree**
  - **Exibir a árvore de processos**
- **top**
  - **Mostra os processos e o respectivo consumo de recursos (dinamicamente)**
- **nice**
  - **Executa um programa com prioridade modificada, ou seja, ditada pelo usuário**
- **renice**
  - **Modifica a prioridade de um processo já em execução**
- **kill**
  - **Envia sinais aos processos (identificados pelo PID)**
- **killall**
  - **Envia sinais aos processos (identificados pelo nome)**
- **pkill**
  - **Envia sinais aos processos (identificados por atributos)**

Linux – Comandos relacionados com processos

- **Envio de sinais para os processos**
- **sighup**
  - relê a configuração
  - evita parar o serviço
- **sigkill**
  - morrer forçado
  - Não é tratado pelo processo
- **sigstp**
  - muda para Task Stopped
  - Para a execução do processo
- **sigterm**
  - Finalização graciosa
  - Dá tempo ao processo que implementa o handler para sigterm
- OBS: Únicos sinais que NÃO são tratáveis são o sigstp e sigkill

Linux – Comandos relacionados com processos

- **bg**
  - **Envia um job para background**
- **fg**
  - **Trás um job de background para foreground**
- **at**
  - **Agendar comandos para o execução futura**
- **chroot**
  - **Executa comando ou shell interativo com o diretório especial de root.**
  - **Muda o diretório root do processo corrente e de seus processos filhos.**
  - **Um programa que é "re-rooted" para um outro diretório não pode acessar arquivos fora daquele diretório, e o diretório é chamado de "prisão chroot"**

Linux – Comandos relacionados com processos

- **cron**
  - Executa comandos agendados
- **crontab**
  - Edita o arquivo onde são especificados os comandos a serem executados e a hora e dia de execução pelo cron.
- **pgrep**
  - Procura por todos os processos com determinado nome e retorna o seu ID
- **pidof**
  - Encontra o ID de um programa em execução
- **sleep**
  - Pausar shell scripts
- **time**
  - Executa programas e sumariza o uso dos recursos do sistema

Linux Para Concursos

Bateria de Questões de Aprendizagem

ITnerante

PROVAS DE TI
TUDO PARA VOCÊ PASSAR

1. No que se refere ao uso e ao funcionamento de sistemas operacionais modernos e suas características, julgue os itens seguintes.

[95] No Linux, durante a configuração para compilação de um novo kernel, é possível colocar os drivers de placas de rede diretamente no kernel ou como módulo de kernel.

2. Para o administrador do sistema SUSE Linux 11 listar a descrição dos usuários cadastrados no sistema na linha de comando do shell, em ordem alfabética, deve-se executar a linha de comando

A. cat /etc/passwd | cut -d: -f6 | sort

B. cat /etc/passwd | cut -d: -f5 | sort

C. cat /etc/shadow | cut -d: -f6 | sort

D. cat /etc/shadow | cut -d: -f5 | sort -r

E. cat /etc/users | cut -d: -f6 | sort –r

3. A respeito do ambiente Red Hat, julgue os próximos itens.

[72] Para alterar a prioridade de um processo que esteja em estado de execução, deve-se utilizar o comando *nice*.

[73] Considere-se que um script chamado *usuariosonline.sh* precise ser executado por um usuário que não possua direitos de administração. Nessa situação, se o usuário em questão souber a senha do usuário root, ele pode executar o script por meio do comando *su -c 'usuariosonline.sh'*.

4. A respeito de sistemas operacionais, julgue os itens que se seguem.

[53] Em sistemas Unix, a proteção de arquivos é efetuada pelo controle dos campos dono, grupo e universo, compostos de três bits (rwx), que definem se um usuário pode ler, escrever ou executar o arquivo

5. O administrador de um computador com sistema operacional Linux deseja saber quais são os usuários que estão "logados" àquele computador no momento. Para isso, ele pode utilizar o comando

A. ps
B. top
C. who
D. finger
E. whoami

GABARITO

1. C
2. B
3. E,C
4. C
5. C

Linux Para Concursos

Bateria de Questões de Aprendizagem

ITnerante

PROVAS DE TI
TUDO PARA VOCÊ PASSAR

6. Com base nas características do sistema operacional Linux, assinale a opção correta.

A. Por meio do comando cut, é possível extrair as últimas linhas de um arquivo.

B. O núcleo do sistema Linux é dividido em dois componentes principais: o de gerenciamento de processos; e o de Entrada/Saída, que é responsável pela interação com os dispositivos de rede e armazenamento.

C. Em todo processo no Linux, há um espaço de endereçamento que consiste de dois segmentos: o segmento de código e o de dado. O segmento de código é o local de armazenamento de todas as variáveis do programa e o segmento de dado contém as instruções de máquina que formam o código executável do programa.

D. O sistema de arquivos Ext2 do Linux escreve, em um diário, de forma ordenada, todas as operações de alterações ocorridas em dados e metadados, visando melhoria de desempenho na gravação em disco.

E. As interfaces gráficas do Linux são executadas pelo sistema X Window.

7. Distribuições Linux, permitem que um usuário sem privilégios especiais para a execução de um determinado comando, o execute, simplesmente precedendo este comando a ser executado com um comando que irá então solicitar as credenciais necessárias para sua execução. O comando em questão é chamado

A. grant
B. adm
C. root
D. admin
E. sudo

8. São comandos que exibem informações sobre os processos em execução em um sistema operacional Linux:

I. top

II. df

III. ps

Os itens CORRETOS são

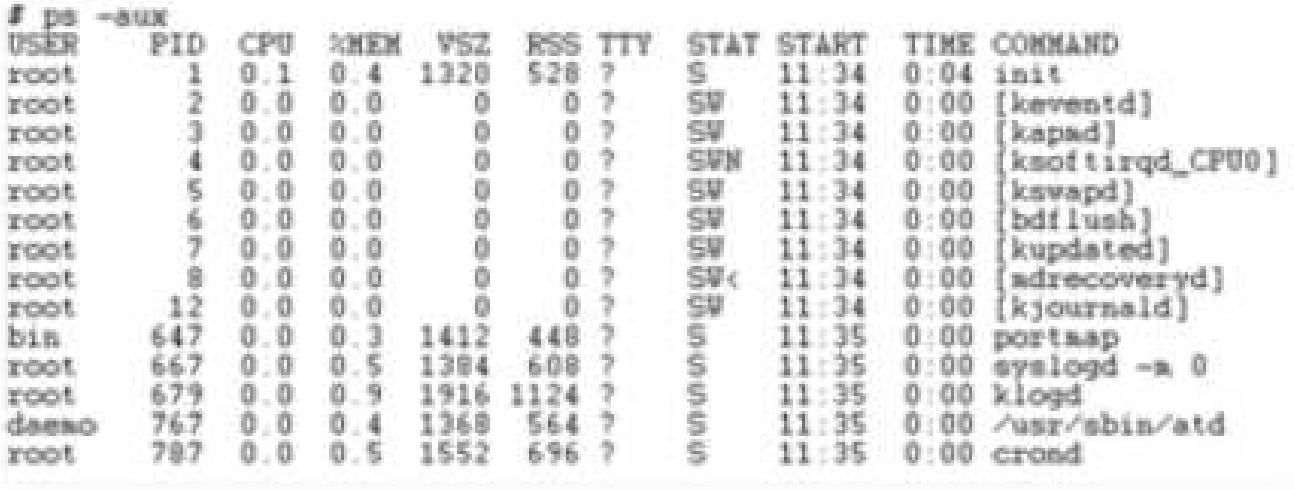
A. I e II, apenas.

B. I e III, apenas.

C. II e III, apenas.

D. I, II e III.

9. Considere a figura a seguir sobre processos em execução de um sistema operacional Linux na sua configuração padrão e responda à questão.



```
# ps -aux
USER       PID  CPU  %MEM   VSZ   RSS TTY      STAT START   TIME COMMAND
root         1  0.1  0.4  1320   528 ?        S    11:34   0:04 init
root         2  0.0  0.0     0     0 ?        SW   11:34   0:00 [keventd]
root         3  0.0  0.0     0     0 ?        SW   11:34   0:00 [kapmd]
root         4  0.0  0.0     0     0 ?        SWN  11:34   0:00 [ksoftirqd_CPU0]
root         5  0.0  0.0     0     0 ?        SW   11:34   0:00 [kswapd]
root         6  0.0  0.0     0     0 ?        SW   11:34   0:00 [bdflush]
root         7  0.0  0.0     0     0 ?        SW   11:34   0:00 [kupdated]
root         8  0.0  0.0     0     0 ?        SW<  11:34   0:00 [mdrecoveryd]
root        12  0.0  0.0     0     0 ?        SW   11:34   0:00 [kjournald]
bin        647  0.0  0.3  1412   448 ?        S    11:35   0:00 portmap
root       667  0.0  0.5  1384   608 ?        S    11:35   0:00 syslogd -m 0
root       679  0.0  0.9  1916  1124 ?        S    11:35   0:00 klogd
daemon     767  0.0  0.4  1368   564 ?        S    11:35   0:00 /usr/sbin/atd
root       787  0.0  0.5  1552   696 ?        S    11:35   0:00 crond
```

As opções a, u e x utilizadas no comando são responsáveis, respectivamente, pelos processos

A. criados, processos que são controlados pelo terminal, nome do usuário e a hora do processo.

B. terminados, processos que não são controlados pelo terminal, hora do processo.

C. terminados, processos que são controlados pelo terminal, hora do processo.

D. criados, processos que são controlados pelo terminal, hora do processo.

E. criados, processos que não são controlados pelo terminal, nome do usuário e a hora do processo.

10. Considere a figura a seguir sobre processos em execução de um sistema operacional Linux na sua configuração padrão e responda à questão.

```
# ps -aux
USER       PID  CPU  %MEM   VSZ   RSS TTY      STAT START   TIME COMMAND
root         1  0.1   0.4  1320   528 ?        S    11:34   0:04 init
root         2  0.0   0.0     0     0 ?        SW   11:34   0:00 [keventd]
root         3  0.0   0.0     0     0 ?        SW   11:34   0:00 [kapmd]
root         4  0.0   0.0     0     0 ?        SWN  11:34   0:00 [ksoftirqd_CPU0]
root         5  0.0   0.0     0     0 ?        SW   11:34   0:00 [kswapd]
root         6  0.0   0.0     0     0 ?        SW   11:34   0:00 [bdflush]
root         7  0.0   0.0     0     0 ?        SW   11:34   0:00 [kupdated]
root         8  0.0   0.0     0     0 ?        SW<  11:34   0:00 [mdrecoveryd]
root        12  0.0   0.0     0     0 ?        SW   11:34   0:00 [kjournald]
bin        647  0.0   0.3  1412   448 ?        S    11:35   0:00 portmap
root       667  0.0   0.5  1384   608 ?        S    11:35   0:00 syslogd -a 0
root       679  0.0   0.9  1916  1124 ?        S    11:35   0:00 klogd
daemo      767  0.0   0.4  1368   564 ?        S    11:35   0:00 /usr/sbin/atd
root       787  0.0   0.5  1552   696 ?        S    11:35   0:00 crond
```

As colunas RSS, TTY e STAT demonstram, respectivamente,

A. o terminal onde são executados os processos, Soma total da memória física usada pelo processo, Estado do processo.

B. o terminal onde são executados os processos, Tempo total da CPU, Estado do processo.

C. o terminal onde são executados os processos, Tamanho do código da tarefa, Estado do processo.

D. a soma total da memória física, Terminal onde são executados os processos, Estado do processo.

E. a soma total da memória física, Nome do comando do processo, Estado do processo.

GABARITO

6. E
7. E
8. B
9. E
10. D

Linux Para Concursos

Bateria de Questões de Aprendizagem

ITnerante

PROVAS DE TI
TUDO PARA VOCÊ PASSAR

11. Um programa presente em várias distribuições do Linux permite a exibição dinâmica dos processos em execução, efetuando automaticamente, a atualização dos processos na tela sem a necessidade de uma nova execução. Trata-se do comando

A. task.
B. ps.
C. df.
D. process.
E. top.

12. Em ambientes operacionais Linux, um símbolo, quando utilizado em conjunto com o comando cd em uma janela de terminal (shell), permite que a mudança de diretório seja efetuada para o diretório home do usuário. Este símbolo é

A. ~ (til).
B. ^ (circunflexo).
C. \ (barra invertida).
D. | (barra vertical).
E. * (asterisco).

13. Assinale a alternativa que apresenta o valor numérico da permissão utilizando o chmod de "-rwxrwxrwx" no sistema operacional Linux.

A. 625.
B. 125.
C. 777.
D. 888.
E. 327.

14. Quanto ao sistema operacional Linux, marque V para verdadeiro ou F para falso e, em seguida, assinale a alternativa que apresenta a sequência correta.

( ) O init é o primeiro processo inicializado no Linux e é o pai de todos os outros processos.

( ) Se um processo termina e deixa processos-filho ainda executando, o processo init assume a paternidade desses processos.

( ) Quando um usuário trabalha no modo monousuário, um único processo shell é inicializado.

( ) A árvore hierárquica dos processos, tendo o shell como raiz, é chamada de sessão.

A. F/ V/ F/ F
B. F/ F/ V/ F
C. V/ V/ F/ F
D. V/ V/ V/ V
E. F/ V/ F/ V

15. Com relação ao sistema operacional Linux, analise as assertivas abaixo.

I. A função fork cria um processo-filho que se diferencia a partir do processo-pai somente em suas PID e PPID e a utilização de recursos é selecionada para 0 (zero).

II. A função fork, em caso de sucesso, devolve a PID do processo filho na thread-pai de execução e 0 (zero) é retornado na thread-filha de execução.

III. A função fork, quando não é executada com sucesso, retorna -1 para o processo-pai, nenhum processo filho será criado e a mensagem de erro adequada será emitida.

É correto o que se afirma em

A. I, apenas.
B. II, apenas.
C. III, apenas.
D. II e III, apenas.
E. I, II e III.

16. Com relação ao sistema operacional Linux, julgue os itens a seguir.

[116] A nomenclatura dos discos no Linux é semelhante ao ambiente Windows, visto que a estrutura de diretórios inicia com unidades de disco definidas por letras.

[117] Os comandos do Linux são arquivos com permissão para serem executados e estão armazenados, em sua maioria, no diretório /bin.

[118] Ao se executar o comando de administração ps -aux, serão apresentados todos os processos correntes no sistema Linux.

[119] Durante a instalação do Linux, é criada uma conta root — uma conta de administrador ou superusuário —, que garante ao usuário root o direito de realizar qualquer atividade no sistema.

[120] Por meio do comando rm –r, é possível acessar diretamente o diretório raiz do ambiente Linux.

17. No que diz respeito ao sistema operacional Linux, julgue os itens que se seguem.

[54] Altera-se a prioridade de um processo em execução, por intermédio do comando *renice*.

18. Julgue os itens seguintes, com relação ao Linux.

[116] No Linux, os usuários são cadastrados no sistema no arquivo */home*, que guarda uma entrada para cada usuário, incluindo-se o diretório e o shell.

19. Os arquivos e diretórios no sistema operacional Linux possuem atributos descritos conforme estrutura apresentada a seguir:

Nessa estrutura, os atributos de números 2 e 4 correspondem, respectivamente:

A. o grupo e outros.
B. o grupo e o proprietário.
C. o proprietário e outros.
D. o proprietário e o grupo.
E. outros e o proprietário.

20. O sistema operacional Linux foi desenvolvido, desde a sua origem, para operar em um ambiente de rede de computadores no qual há o acesso remoto dos usuários. Para verificar quais usuários estão logados no sistema, pode-se utilizar o comando:

A. ps
B. who
C. user
D. finger
E. logged

21. Qual dos seguintes comandos abaixo busca no arquivo texto "text.txt" as linhas que possuem o texto UFPE?

A. grep UFPE text.txt
B. grep -v UFPE text.txt
C. awk UFPE text.txt
D. awk -v UFPE text.txt
E. sed UFPE text.txt

22. Os sistemas operacionais utilizados em computadores do tipo servidor devem disponibilizar recursos diferenciados para o gerenciamento dos arquivos, usuários e da segurança do sistema. Nesse contexto, os sistemas operacionais Linux, como o Mandriva 2007, disponibilizam recursos nativos para essas finalidades. Por exemplo, as informações dos usuários são armazenadas em um arquivo distinto do arquivo das respectivas senhas, e que são, respectivamente,

A. /boot/users e /etc/passwd.
B. /boot/login e /etc/shadow.
C. /home/users e /etc/passwd.
D. /etc/passwd e /etc/shadow.
E. /etc/users e /etc/passwd.

GABARITO

11. E
12. A
13. C
14. D
15. E
16. E, C, C, C, E
17. C
18. E
19. C
20. B
21. A
22. D

Linux Para Concursos

Bateria de Questões de Aprendizagem

ITnerante

PROVAS DE TI
TUDO PARA VOCÊ PASSAR

23. Considere a seguinte sequência de comandos executados em um sistema Linux:

$ ls -l file.1

-rwx---r-x 1 meg rh 588 Nov 14 14:51 file.1

$ whoami

jack

$ groups

jack rh adm

Com base nos resultados, é possível constatar que o usuário atual

A. é capaz de ler o conteúdo do arquivo “file.1”, pois o arquivo pode ser lido por todos os usuários do sistema.

B. é capaz de ler ou alterar o conteúdo do arquivo “file.1”, pois é membro do grupo “adm” que garante direitos administrativos no sistema.

C. é capaz de alterar o arquivo “file.1”, pois pertence ao mesmo grupo que o usuário “meg”.

D. não é capaz de ler o conteúdo do arquivo “file.1”, pois ele pode ser acessado apenas pelo usuário “meg”.

E. não é capaz de ler o conteúdo do arquivo “file.1”, pois é membro do grupo “rh” que não possui direitos de acesso a esse arquivo.

24. No sistema operacional Linux, deseja-se atribuir ao arquivo “xpto.sh” as seguintes permissões:

- Dono do arquivo: leitura e escrita apenas.
- Usuário do grupo do arquivo: leitura apenas.
- Outros usuários: leitura e execução apenas.

Para que tais permissões sejam atribuídas ao arquivo indicado, é necessário executar o comando:

A. chmod 0315 xpto.sh
B. chmod 0546 xpto.sh
C. chmod 0513 xpto.sh
D. chmod 0645 xpto.sh
E. chmod 0777 xpto.sh

25. Com relação ao sistema operacional Linux, julgue os itens subsequentes.

[63] O Linux permite logins simultâneos de vários usuários. Para visualizar os usuários logados no Linux em determinado momento, deve-se executar o comando *who*.

[64] No Linux, um processo que consome grande quantidade de memória deve ser terminado de forma imediata, ação que pode ser realizada utilizando-se o comando *ps* seguido do número que identifica o processo.

[65] Ao instalar o Linux, é necessário gravar informações no master boot record, que fazem referência aos arquivos encarregados de inicializar o sistema operacional.

[66] Para selecionar um novo fuso horário para o Linux, pode-se executar o comando *hwclock*.

26. Um administrador do SUSE Linux 11 deseja permitir que todos os usuários possam criar arquivos na pasta /projeto do sistema de arquivos, garantindo que os usuários possam apagar apenas seus próprios arquivos. Para isso, o administrador deve executar o comando

A. chmod 0777 /projeto
B. chmod 1777 /projeto
C. chmod 2777 /projeto
D. chmod 4777 /projeto
E. chmod 6777 /projeto

27. Na distribuição Linux Red Rat, o comando useradd é utilizado para adicionar novos usuários ao sistema. Um de seus atributos informa que o diretório home do usuário deve ser criado. Este atributo é o

A. -h
B. -m
C. -c
D. -C
E. -d

28. No Red Hat Linux, há três tipos diferentes de permissões para arquivos, diretórios e aplicações. Estas permissões são usadas para controlar os tipos de acesso permitidos. São usados símbolos diferentes de caractere para descrever cada permissão em uma listagem de diretórios. São usados: r para a permissão de leitura, w para a permissão de escrita e, para a permissão de execução de um arquivo, é atribuída a letra

A. e.
B. x.
C. p.
D. a.
E. l.

29. Um comando muito utilizado em distribuições Linux, permite que sejam alteradas as informações de propriedade de usuário e grupo para um determinado arquivo ou diretório, aplicando, inclusive, essas alterações de forma recursiva. O comando em questão, em conjunto com o atributo de recursividade é corretamente exposto em

A. usermod -S
B. chmod --dereference
C. ln --recursive
D. chown -R
E. chggrp -S

30. Arquivos em Linux são protegidos atribuindo-se a cada um deles um código de proteção de 9 bits. O código de proteção consiste em campos de 3 bits, um grupo para qualquer usuário, outro para o usuário do arquivo e um para o grupo ao qual o usuário pertence. Cada campo possui um bit de permissão de leitura, um bit de permissão de escrita e outro de permissão de execução. Por exemplo, o código de proteção de um arquivo definido como “-wxr-xr--" significa que:

A. membros do grupo e o proprietário podem ler, executar e escrever no arquivo e outros usuários podem apenas ler.

B. membros do grupo podem escrever e executar o arquivo, qualquer usuário pode ler e executar o arquivo e o dono do arquivo pode apenas ler o conteúdo do arquivo.

C. qualquer usuário pode escrever e executar o arquivo, o proprietário pode ler e executar o arquivo e membros do grupo podem apenas ler o arquivo.

D. o proprietário pode escrever e executar o arquivo, membros do grupo podem ler e executar o arquivo e qualquer usuário pode ler o arquivo.

E. o proprietário pode ler, escrever e executar o arquivo, membros do grupo podem ler e escrever no arquivo e qualquer usuário pode ler e executar o arquivo.

GABARITO

23. E
24. D
25. C, E, C, E
26. B
27. B
28. B
29. D
30. D