

Introdução ao GNU/Linux

Curso de introdução ao GNU/Linux
Abordagem Prática para HPC

Apresentações

- Nome
- Onde trabalha/estuda/estagia
- Conhecimentos/experiência com informática
- Conhecimentos/experiência com GNU/Linux
- Conhecimentos/experiência com HPC
- Inglês para leitura
- O que espera do curso

Objetivos do curso

- Curso básico e prático de utilização do GNU/Linux
- Capacitar usuários para acessarem e utilizarem um ambiente HPC baseado em GNU/Linux.
- Noções de como executar linhas de comandos nesse ambiente. Variáveis de ambiente. Lidar com arquivos e diretórios e suas permissões. Editar arquivos.
- Noções de como compilar, instalar um programa e como executar ele no ambiente HPC.

Material auxiliar

- Materiais que podem ser consultados durante o curso e que podem ser usados depois para aprofundar os conhecimentos.
- Guia Foca Linux - www.guiafoca.org
- The Linux Documentation Project – tldp.org
- Documentação da distribuição GNU/Linux - www.redhat.com, www.debian.org, www.slackware.com, etc...

HPC

- High-performance computing. Termo em inglês para computação de alto desempenho. São computadores de alto desempenho que são configurados para a execução de programas que requerem grande capacidade de processamento, memória, armazenamento ou ambos.
- Tem uma lista de maiores soluções em HPC, top500.org. Essa lista aponta quase a totalidade dela em soluções GNU/Linux e UNIX.

<http://top500.org/statistics/list/>

<http://www.gnu.org>

GNU/Linux

- GNU/Linux seria o nome oficial do sistema operacional que usualmente conhecemos como Linux.
 - GNU (GNU is not Unix) conjunto de ferramentas como shell, compiladores, editores e ferramentas do sistema operacional. Desenvolvido para ser uma alternativa livre as ferramentas do Unix.
www.gnu.org
 - Linux: Kernel do sistema operacional desenvolvido para compor um sistema livre junto com o GNU.
www.kernel.org

Acesso ao ambiente

- O GNU/Linux tem um ambiente gráfico com muitos programas úteis e também tem programas em linhas de comandos, formato texto.
- O GNU/Linux pode ser utilizado localmente no seu computador ou utilizado remotamente utilizando um programa de acesso. O programa de acesso remoto padrão é o ssh.
- Tanto os programas gráficos quanto de linha de comando podem ser utilizados localmente ou remotamente.
- Durante o curso focaremos nos programas em linha de comando porque é o acesso predominante aos ambientes e programas HPC.

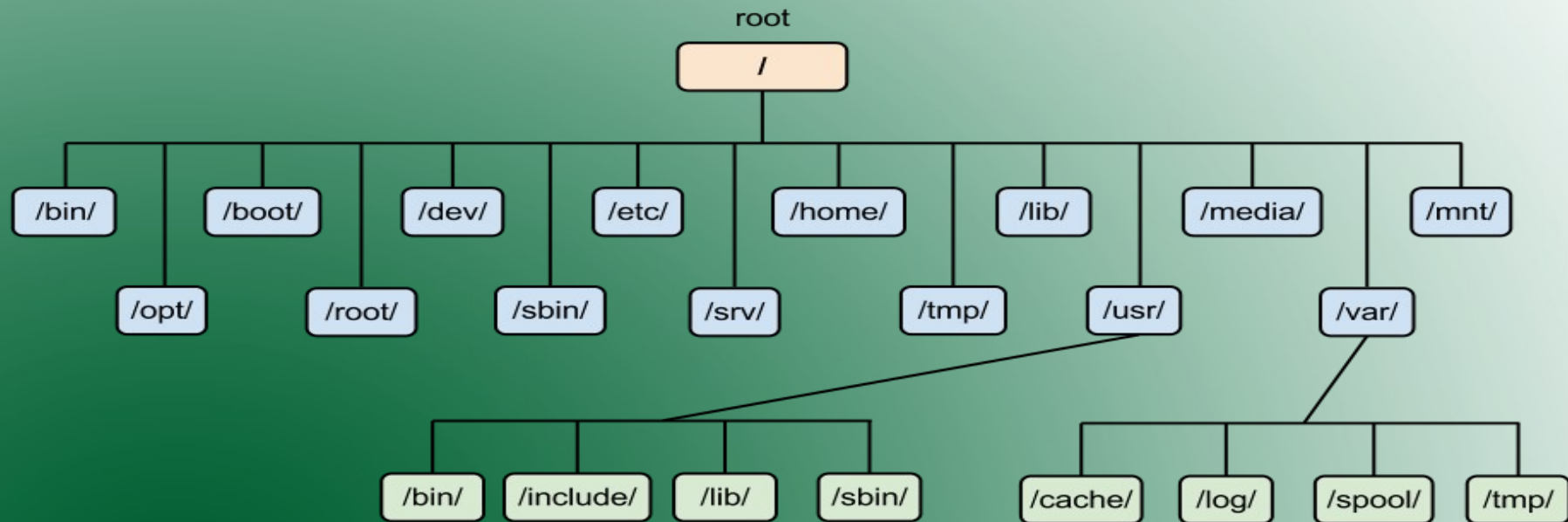
Acesso ao ambiente

- Acesso local (micro local)
 - Aplicativos -> Utilitarios -> Terminal
 - Obs: todos compartilham os mesmos arquivos
 - Dentro do terminal criar diretório com o seu nome e trabalhar os exemplos nele
 - Utilizaremos esse acesso todos os exercícios exceto o com o torque
- Acesso Remoto (dentro da sala de cursos)
 - No Terminal
 - `ssh curso@sandman.cenapad.unicamp.br`
 - Utilizaremos esse acesso somente no exercício do torque

Usuários, Grupos e Permissões

- O Sistema GNU/Linux é multiusuário. O acesso a recursos do sistema são baseados em permissões baseadas por usuário e grupo.
- O usuário e grupo root tem acesso irrestrito no sistema.
- O GNU/Linux implementa permissões simples padrão UNIX como UGO (user,group,other). Também complexas como ACLs (access control lists), atributos estendidos e outras configurações de acesso mais complexos como AppArmor, SELinux, Smack e TOMOYO

Estrutura de arquivos



Estrutura de arquivos

- Organização hierárquica conhecida como árvore invertida (organograma)
- Estrutura implementada através de arquivos especiais chamados de diretórios.
- Cada diretório é uma bifurcação em uma hierarquia, de onde outros ramos podem derivar
- Diretórios podem conter outros diretórios, arquivos ou ambos
- Tanto diretórios e arquivos são case sensitive.
- Diretórios são estruturados funcionalmente para programas, configurações, pontos de montagem, arquivos de usuários, arquivos de sistema, manuais, bibliotecas e código fonte.

Estrutura de arquivos

/ Raiz

/bin – Programas essenciais para o sistema operacional

/boot – Kernel e arquivos de sistema

/etc – Arquivos de configuração do sistema

/home – arquivos dos usuários

/lib bibliotecas essenciais para o sistema operacional

/proc – arquivos do sistema operacional

/sbin – Programas essenciais de administração do sistema operacional

/tmp – Arquivos temporários

/usr – Estrutura que contem programas e utilitários para uso geral.

/usr contem: **/usr/bin**, **/usr/lib**, **/usr/sbin** e outros como **/usr/include**, **/usr/share**

/usr/local – Estrutura que contem programas e utilitários específicos do host

/usr/local contem: **/usr/local/bin**, **/usr/local/include**, **/usr/local/lib**, **/usr/local/<software>**

Tipos de arquivos

- Arquivos comuns
 - Arquivos de usuários, textos, código fonte, etc..
 - Programas
 - Localizado em um diretório
- Diretórios
 - Estrutura hierárquica de arquivos
 - Pode conter diretórios e arquivos
- Arquivos especiais
 - Arquivos que representam hardware
 - Arquivos que servem de entrada e saída

Atributos dos arquivos

- Tipos de arquivos
 - - arquivo
 - d diretório
 - l link
 - b, c e s especiais
- Tamanho em bytes
- Dono e Grupo
- Data de criação, modificação e acesso
- Permissões

Permissões

- Arquivos
 - r ler arquivo
 - w alterar o arquivos
 - x executar o arquivo
- Diretórios
 - r listar o diretório comando ls
 - w escrever no diretório, modificar o conteúdo dele
 - x executar no diretório comando cd

Permissões (Extra)

- Atributos de Arquivos e Diretórios
 - `man chattr, man lsattr`
 - Atributos como `i` – immutable, `a` append only
- ACL
 - `man chacl, man get getfacl`
 - Permissões por usuário e por grupo estendendo o padrão UGO

Extensão de arquivos

- Arquivos podem ser classificados pela sua extensão separado por ponto (.) como um arquivo texto com o nome arquivo.txt
- Existem muitas extensões válidas mas as mais usadas no GNU/Linux:
 - .a biblioteca estática de Programa
 - .bz2, .gz, .xz Arquivos compactos
 - .tar Arquivos agrupados
 - .conf Arquivo de configuração
 - .so Biblioteca compartilhada
 - .o Código Objeto
 - .c, .C, .f, .F, .h Arquivo código fonte
 - .sh Arquivo bash (não é obrigatório a extensão)

Nomes de arquivos e pastas

- Os nomes de arquivos e pastas são case sensitive
- Nomes com até 256 caracteres (padrão)
- Nomes podem ser compostos por letras, números e símbolos Não pode usar todos os caracteres exceto barra (/) que é reservado para caminho.
- Apesar de poder usar qualquer caractere, é recomendado que somente utilize letras, números e ponto (.). Outros caracteres dificultam a utilização e a portabilidade.
- Arquivos que começam com “.” são arquivos escondidos (hidden).

SHELL (Ambiente de Comandos)

- Possibilita a comunicação usuário e sistema.
- Interpretar e executar comandos.
- Linguagem de programação.
- Variáveis shell.
- Execução em foreground e background.
- Redirecionamento de Input e Output.
- Comandos Alias (apelidar comandos)
- Existem várias implementações de shell para GNU/Linux. O Shell padrão nesse ambiente é o bash

bash - GNU Bourne-Again SHell

- Implementação shell baseada no shell do UNIX o sh feito pelo Stephen Bourne.
- Interpretador de comandos e linguagem de programação
- Interativo (teclado) e não interativo (rotinas agendadas).
- Execução síncrona e assíncrona
- Comandos úteis implementados internamente
- Linguagem de programação com variáveis, controle de fluxo e funções
- Linguagem de programação voltada para o uso interativo
- Histórico de comandos e aliases (apelidos)

Linhas de comando

- Comandos em Linha aceitam a passagem de parâmetros No GNU/Linux o padrão é:
 - Sinal de menos (-) mais uma letra
 - `ls -a`
 - `ls -al`
 - Dois sinais de menos (- -) e uma palavra
 - `ls -all`
 - Dois sinais de menos sinaliza final de opções
 - `ls --all`
 - O formato padrão dos comandos
 - `<comando> [espaço] -[letra] [espaço] --[palavra]`

Linhas de comando

Exemplos	Certo	Errado
Separação	<code>\$ls -l /tmp</code>	<code>\$ls -l/tmp</code>
	<code>\$ls -l /bin</code>	<code>\$ls-l /bin</code>
Múltiplas opções	<code>\$ls -l -F</code>	<code>\$ls -l-F</code>
	<code>\$ls -lF</code>	<code>\$ls -l F</code>
	<code>\$ls -l --classify</code>	<code>\$ls -lclassify</code>
Múltiplos argumentos	<code>\$ls arquivo1 arquivo2</code>	<code>\$ls arquivo1,arquivo2</code>

Manipular arquivos/diretórios

- Exibir o diretório atual
 - pwd
- Listar conteúdo do diretório
 - ls
- Trocar de diretório
 - cd pasta
 - cd .. (não existe ... como no DOS)
 - cd /caminho/pasta
 - cd ~
 - cd -

Manipular arquivos/diretórios

- Podemos criar um diretório com mkdir
 - mkdir <diretório>
- Podemos apagar um diretório com rmdir ou rm (diretórios vazios)
 - rmdir <diretório>
 - rm -dir (alguns GNU/Linux desabilitam essa opção)
- Podemos apagar diretórios com arquivos
 - rm -r
- Arquivos apagados não são recuperados facilmente no Linux.

Teclas especiais no GNU/Linux

- Teclas de movimento
 - Ctrl+A Inicio do comando
 - Ctrl+E Fim do comando
- Teclas de controle de programas
 - Ctrl+C Termina o programa
 - Ctrl+Z Pausar o programa
- Teclas úteis
 - Ctrl+L Limpa a tela
 - Ctrl+U apaga toda a linha de comando
 - Tab completar comandos

Curingas

- “*” : Nome completo (*.c, a*,a*.h)
- “?” : Uma posição (arquivo?.c, arquivo.?)
- [padrão]
 - [a-z][0-9]
 - [a,b,c][0,2,4]
 - [a-c,s]
- {padrão}
 - X{0,1,2} X0 X1 X2
 - X{1..5} X1 X2 X3 X4 X5
 - X{0,4-6} X0 X4 X5 X6

Obtendo ajuda no GNU/Linux

- Ajuda do programa -h ou --help
 - ls --help
- Manual on-line man
 - man <comando>
 - man -k <expressão>
- Documentos info
 - info <comando>
 - Info -k <expressão>
- Comando help
 - Ajuda para comandos internos do bash
 - help <expressão>

Man

- O man está dividido em áreas de acordo com o número dela
 1. Programas e comandos shell
 2. Chamadas de sistema
 3. Chamadas de bibliotecas
 4. Arquivos especiais do GNU/Linux
 5. Formatos de arquivos, configurações e convenções
 6. Jogos

Man

7 Diversos

8 Comandos de administração do GNU/Linux

9 Rotinas do Kernel (Não é padrão)

- `man <n> <palavra>`
- `ls /usr/share/man/man<n>`

Exercícios – Preparar o ambiente

- 1) No diretório home criar uma pasta “GNULINUX” e criar subpastas “aula01”, “aula02”, “aula03”, “aula04” e “aula05”. Dentro de cada subpasta “aulaXX” criar subpasta “exemplos” e “exercicios”.
- 2) Dentro da pasta ~/GNULINUX/aula01/exemplos execute o comando
`touch {a..z}{i,ii,iii}{1..10}.{a,c,so,o}`
Esse comando irá demorar um pouco para criar todos os arquivos.

Exercícios

1) Dentro da pasta `~/GNULINUX/aula01/exemplos` execute os comandos:

- `ls, ls -m, ls -l, ls -l -v`
- `ls a*, ls a*.c, ls a??.c, ls a?????.c, ls ?iii?.c, ls *9.c`
- `ls ai[1-5].c, ls [a-c]ii1.c`
- `echo a*, ls a*.c, ls a??.c, ls a?????.c, ls ?iii?.c, ls *9.c`
- `echo ai[1-5].c, ls [a-c]ii1.c`

Escolha 3 comandos desses e descreva com suas palavras o que ele faz. Salve o texto em `~/GNULINUX/aula01/exercicios/ex01.txt`

Exercícios

2) Execute `man ls` e veja 3 opções que podem ser úteis para listar arquivos. Salve o texto em `~/GNULINUX/aula01/exercicios/ex02.txt`

3) Execute

`man <n> intro` (n de 1 a 8) e leia a parte DESCRIPTION

Por exemplo:

`man 1 intro`

Destaque um manual e descreva com suas palavras a parte DESCRIPTION no arquivo `~/GNULINUX/aula01/exercicios/ex03.txt`

Exercícios

- 4) Leia os manuais dos comandos: ls, mkdir, rmdir e rm Destaque uma opção útil de cada manual e salve em ~/GNULINUX/aula01/exercicios/ex04.txt