

# Introdução ao GNU/Linux

Curso de introdução ao GNU/Linux  
Abordagem Prática para HPC



# Apresentações

- Nome
- 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](http://www.guiafoca.org)
- The Linux Documentation Project – [tldp.org](http://tldp.org)
- Documentação da distribuição GNU/Linux - [www.redhat.com](http://www.redhat.com), [www.debian.org](http://www.debian.org), [www.slackware.com](http://www.slackware.com), etc...

# GNU/Linux uso pessoal

Podem Instalar o GNU/Linux no computador pessoal, executar em um Virtualizador ou executar em modo “Live”. Existem várias opções nessas três configurações mas indicamos:

- Instalação: Ubuntu <https://www.ubuntu.com>
- Virtualização: no Windows com Virtual Box ( <https://www.virtualbox.org> e <https://www.osboxes.org/virtualbox-images> )
- “Live” :
  - <https://tutorials.ubuntu.com/tutorial/tutorial-create-a-usb-stick-on-windows#0>

# HPC

- High-performance computing. Termo em inglês para computação de alto desempenho. São computadores configurados para a execução de programas que requerem grande capacidade de processamento, quantidade de memória, armazenamento ou uma combinação dos dos recursos.
- Lista dos maiores HPC
  - Sistemas de grande porte utilizando GNU/Linux.
  - <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](http://www.gnu.org)
  - Linux: Kernel do sistema operacional desenvolvido para compor um sistema livre junto com o GNU.
    - [www.kernel.org](http://www.kernel.org)
  - Podemos chamae de GNU/Linux ou somente Linux

# Acesso ao ambiente

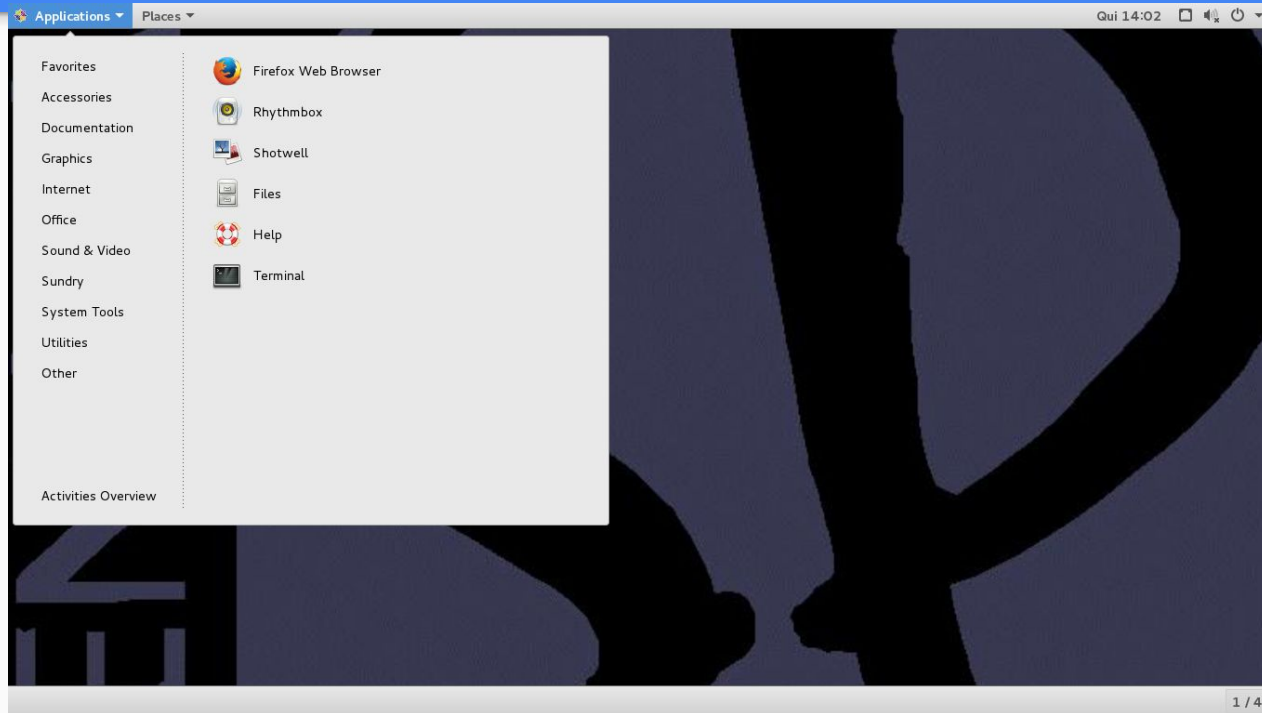
- Acesso ao GNU/Linux
  - Acesso por ambiente Gráfico e Linha de comando
  - Acesso local ( teclado, mouse e monitor ) ou remoto (ssh)
  - Acesso local e remoto podem ser por Linha de comando ou ambiente gráfico.
- Ambiente do curso
  - Acesso linha de comando na Interface gráfica
  - Acesso remoto com o ssh



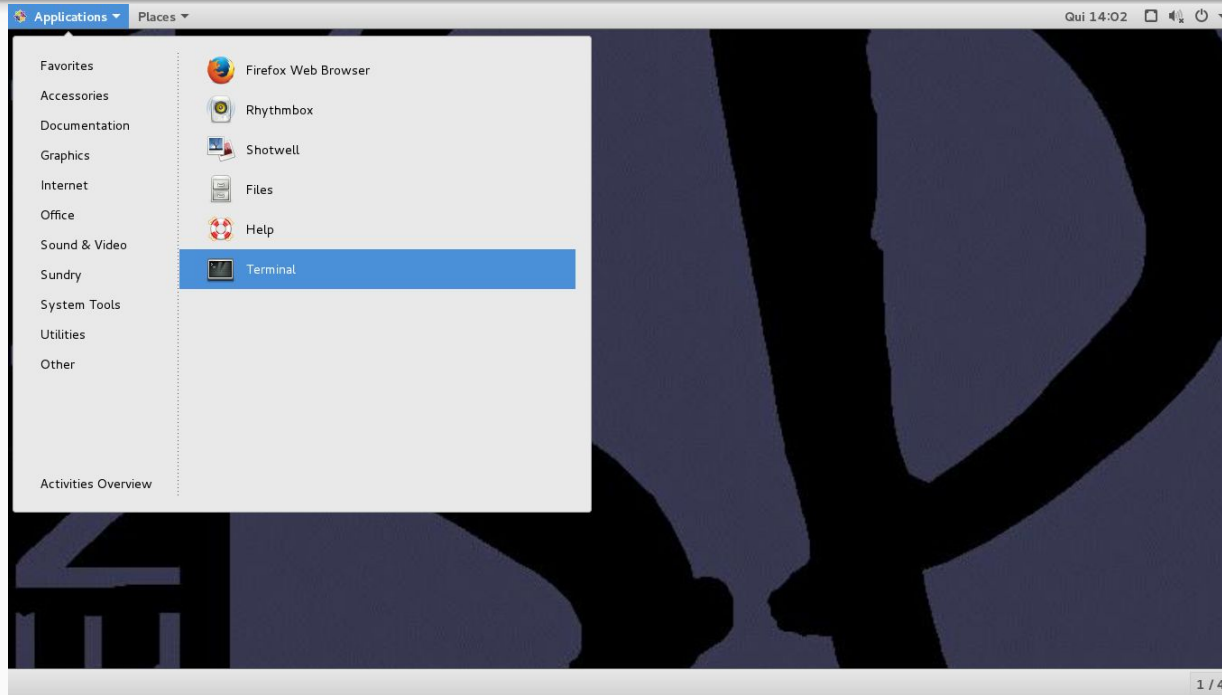
# Acesso ao ambiente

- Acesso local ( micro local )
  - Aplicativos -> Utilitários -> Terminal
  - Utilizaremos esse acesso todos os exercícios exceto na última aula.
- Acesso Remoto ( dentro da sala de cursos )
  - No Terminal
  - `ssh curso@sandman.cenapad.unicamp.br`
  - Utilizaremos esse acesso somente no exercício do torque

# Acesso ao ambiente



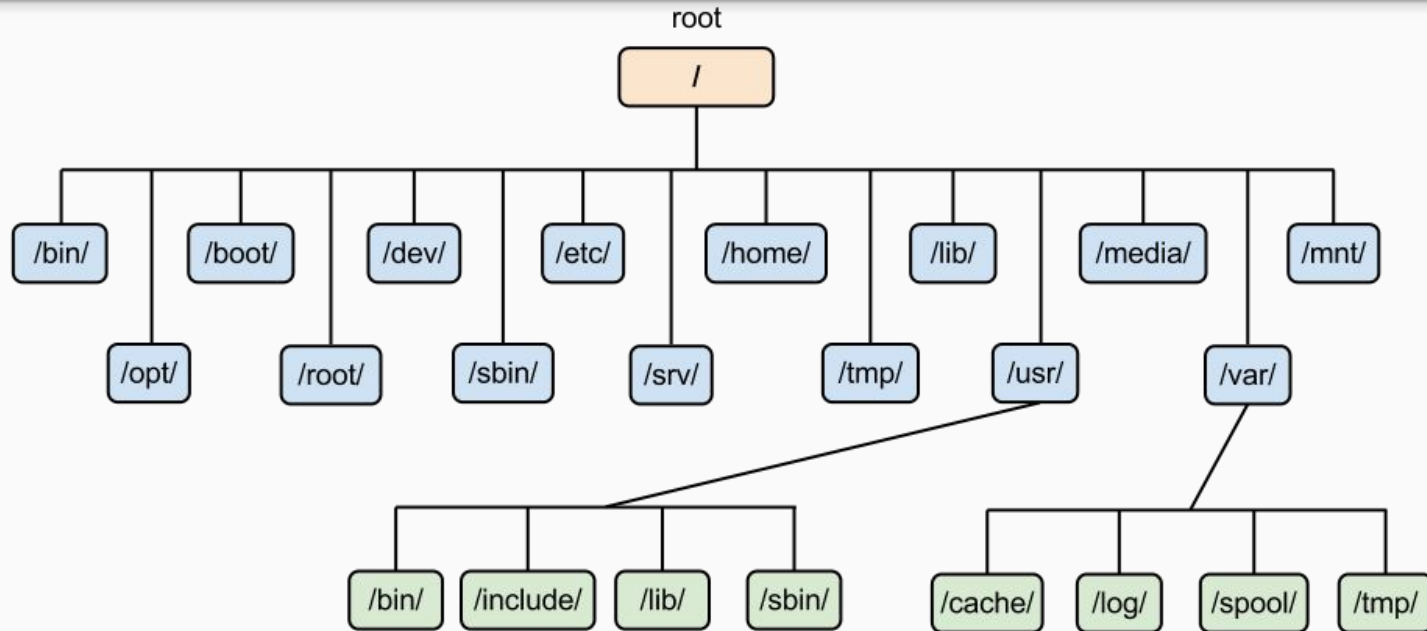
# Acesso ao ambiente



# Usuários, Grupos e Permissões

- O Sistema GNU/Linux é multiusuário.
- O padrão de acesso a recursos é feito utilizando usuário e/ou grupo.
- O usuário root tem acesso irrestrito no sistema.
- O GNU/Linux implementa permissões simples padrão UNIX como UGO (user,group,other).
- Existem outras opções como ACLs (access control lists), atributos estendidos, AppArmor, SELinux, Smack e TOMOYO.
  - Curso abordará UGO

# 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

# Estrutura de arquivos

- **/sbin** – Programas essenciais de administração do sistema operacional
- **/tmp** – Arquivos temporários
- **/usr** – Estrutura que contém programas e utilitários para uso geral.
  - **/usr** contém: /usr/bin, /usr/lib, /usr/sbin e outros como /usr/include, /usr/share
- **/usr/local** – Estrutura que contém programas e utilitários específicos do host
  - **/usr/local** contém: /usr/local/bin, /usr/local/include, /usr/local/lib
  - /usr/local/<software>
  - /home/cursos/<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` ou `../../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 Início 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:
  - a. `ls, ls -m, ls -l, ls -l -v`
  - b. `ls a*, ls a*.c, ls a??.c, ls a????.c, ls ?iii?.c, ls *9.c`
  - c. `ls ai[1-5].c, ls [a-c]ii1.c`
  - d. `echo a*, ls a*.c, ls a??.c, ls a????.c, ls ?iii?.c, ls *9.c`
  - e. `echo ai[1-5].c, ls [a-c]ii1.c`

Escolha 3 comandos acima 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. Descreva eles com suas palavras `~/GNULINUX/aula01/exercicios/ex02.txt`
3. Execute
  - a. `man <n> intro` ( `n` de 1 a 8 ) e leia a parte `DESCRIPTION`
  - b. Por exemplo:
  - c. `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 descreva eles com suas palavras no arquivo `~/GNULINUX/aula01/exercicios/ex04.txt`