Escola Supercomputador SDumont

MC-SD07. Introdução à configuração e gerenciamento de Clusters

Programa de Verão 2021 - Escola SDumont
LNCC – Petrópolis/RJ

André Ramos Carneiro - andrerc@lncc.br
Bruno Alves Fagundes - brunoaf@lncc.br

MC-SD07. Introdução à configuração e gerenciamento de Clusters

- Download do material
- https://www.lncc.br/~brunoaf/MCSD07-slide.zip
- https://www.lncc.br/~brunoaf/MSCD07-vms.ova.zip

  Ou

- http://www.cenapad-rj.lncc.br/tutoriais/materiais-hpc/semana-sdumont/

- Extrair o conteúdo do arquivo e importar as vms através do VirtualBox
- Menu Arquivo > Importar Appliance...

Roteiro

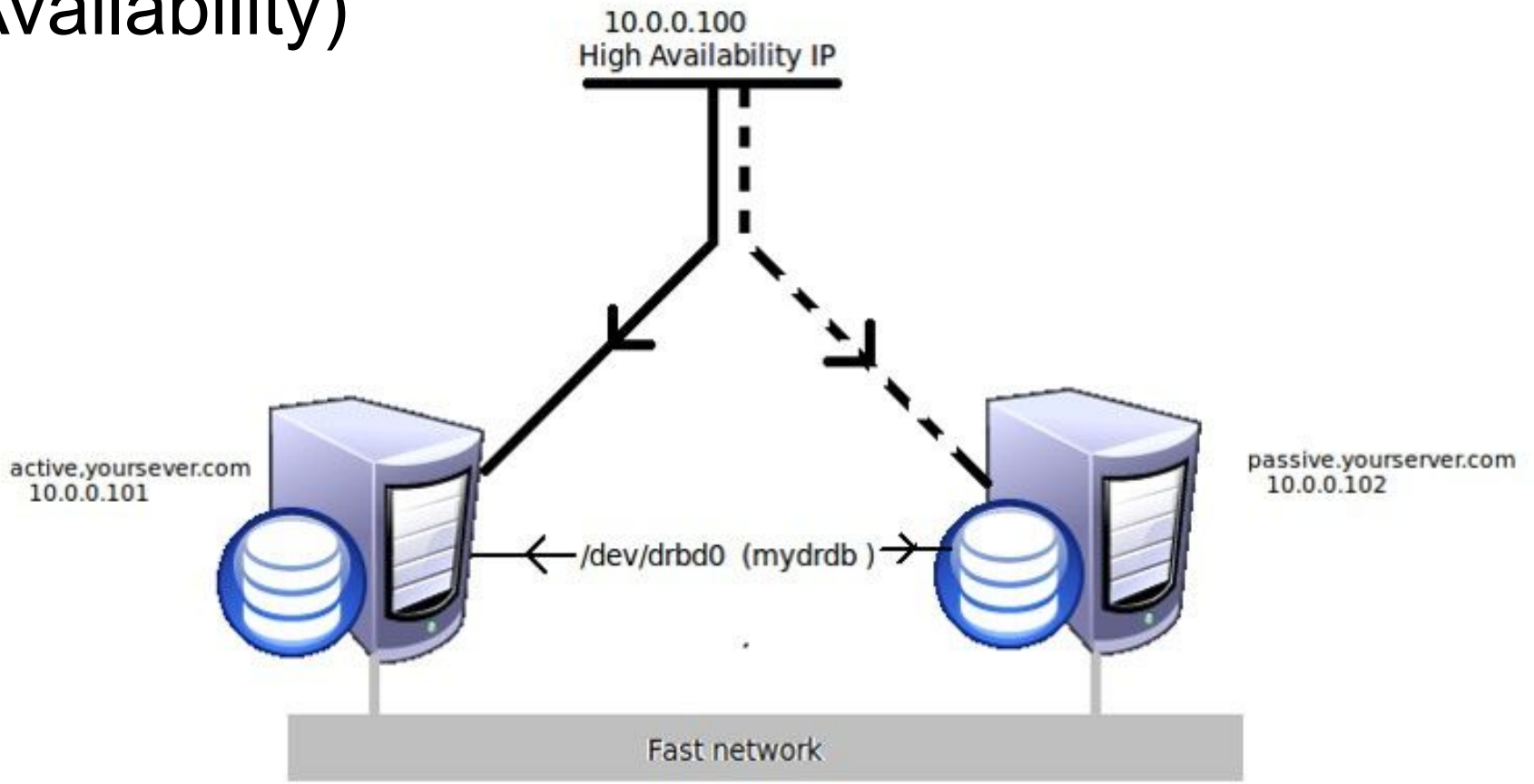
- Overview sobre Clusters para HPC
- Elementos de Infraestrutura
- Configuração do cluster de testes
- Atividades de administração

Overview sobre clusters para HPC

Overview sobre clusters para HPC

Tipos de clusters

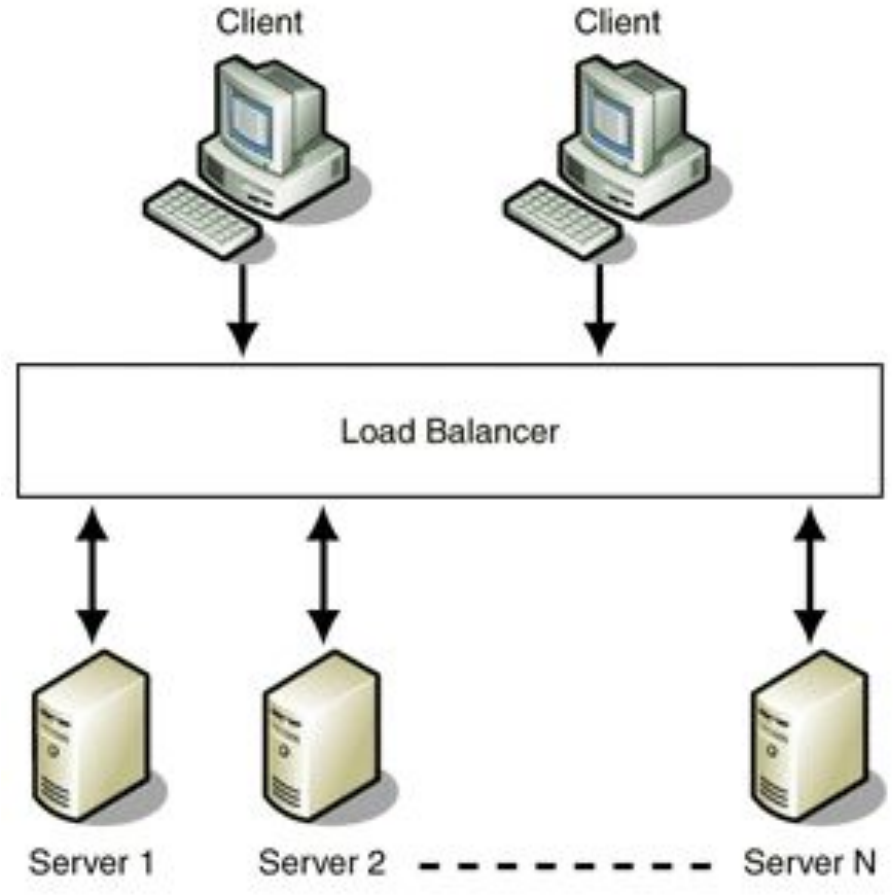
- Alta disponibilidade - HA (Failover, High Availability)



Overview sobre clusters para HPC

Tipos de clusters

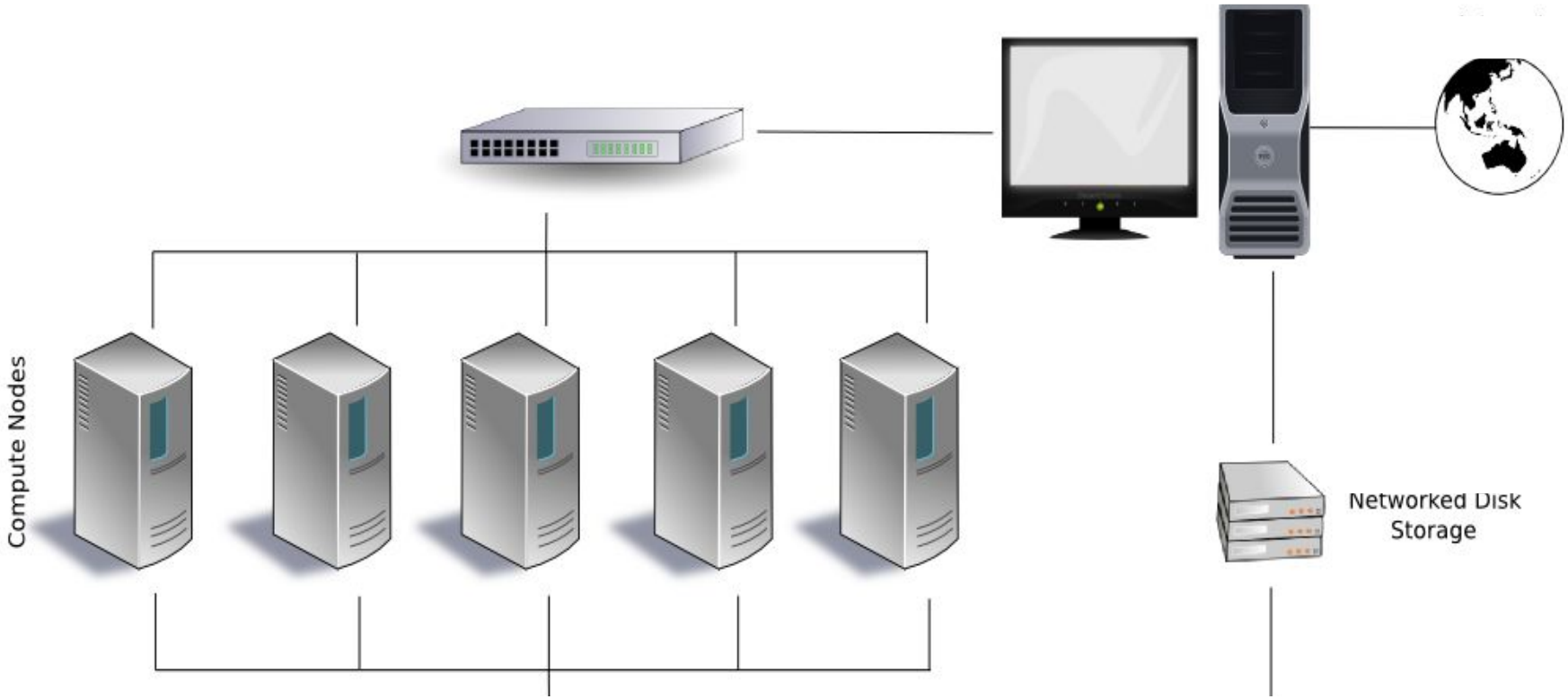
- Balanceamento de carga (Load Balancing)



Overview sobre clusters para HPC

Tipos de clusters

- Alto desempenho - HPC (High Performance Computing)



Overview sobre clusters para HPC

Clusters de alto desempenho

- Resolver problemas complexos
- Executa tarefas em paralelo
- Rede de alta performance
- Biblioteca de troca de mensagens (MPI, PVM etc)
- Sistema de armazenamento compartilhado

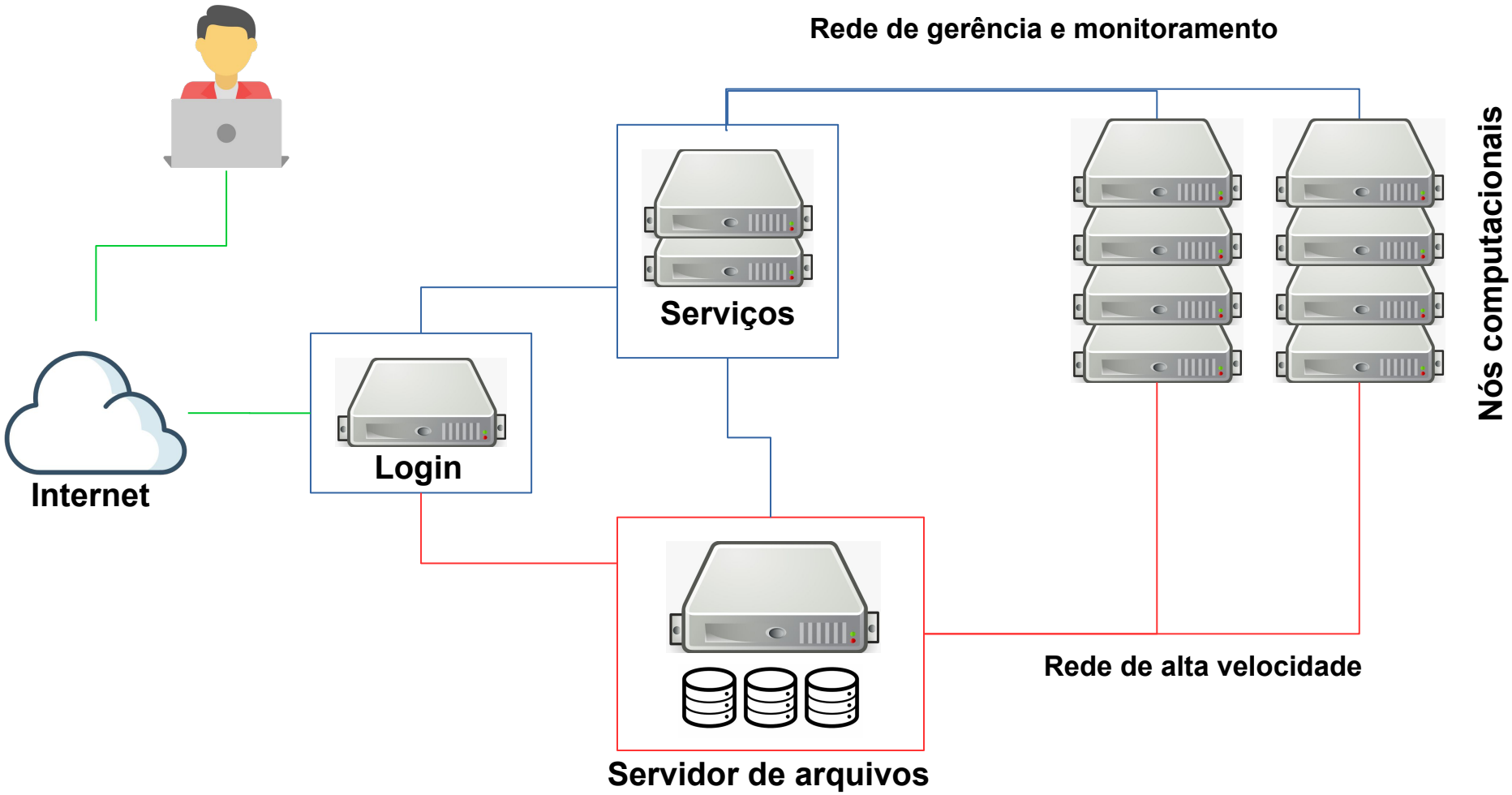
Overview sobre clusters para HPC

Desafios do HPC

- Tamanho do problema
- Comunicação entre processos
- Acesso compartilhado aos dados

Overview sobre clusters para HPC

Arquitetura básica de um cluster para HPC



Overview sobre clusters para HPC

Supercomputadores

Soluções completas de grandes fabricantes

Clusters de PCs

Projetos customizados
utilizando máquinas pessoais

Projeto Carcará (2000)
Cluster de 32 PCs

Infraestrutura

- Gerenciador de recursos
- Sincronismo dos relógios
- Repositório de imagens
- Sistema de arquivos compartilhado
- Contas de usuários centralizadas
- Autenticação sem senha entre os nós
- Módulos de ambiente
- Rede de alta velocidade

Infraestrutura

- Gerenciadores de recursos

Open Grid Scheduler

Univa Grid Engine

Moab HPC Suite

Infraestrutura

- Sincronismo dos relógios

Protocolo NTP – Network Time Protocol

```
# yum install ntp -y
```

Servidores NTP.br

a.st1.ntp.br	200.160.7.186 e 2001:12ff:0:7::186
b.st1.ntp.br	201.49.148.135
c.st1.ntp.br	200.186.125.195
d.st1.ntp.br	200.20.186.76
a.ntp.br	200.160.0.8 e 2001:12ff::8
b.ntp.br	200.189.40.8
c.ntp.br	200.192.232.8
gps.ntp.br	200.160.7.193 e 2001:12ff:0:7::193

Infraestrutura

- Repositório de imagens
  - Aplicação de patches e correções
  - Manter versões diferentes do sistema operacional
  - Provisionamento/deployment

- É possível utilizar um software de automação para aplicar as configuração nos nodes.

Infraestrutura

- Sistema de arquivos compartilhado

Sistemas de arquivos paralelos

lustre®
File System

IBM Spectrum Scale

Sistemas de arquivos distribuídos

NFS

Network File System

Infraestrutura

- Contas de usuários centralizadas

NIS

Network Information Service

Infraestrutura

- Autenticação sem senha entre os nós computacionais (usuário ou host)

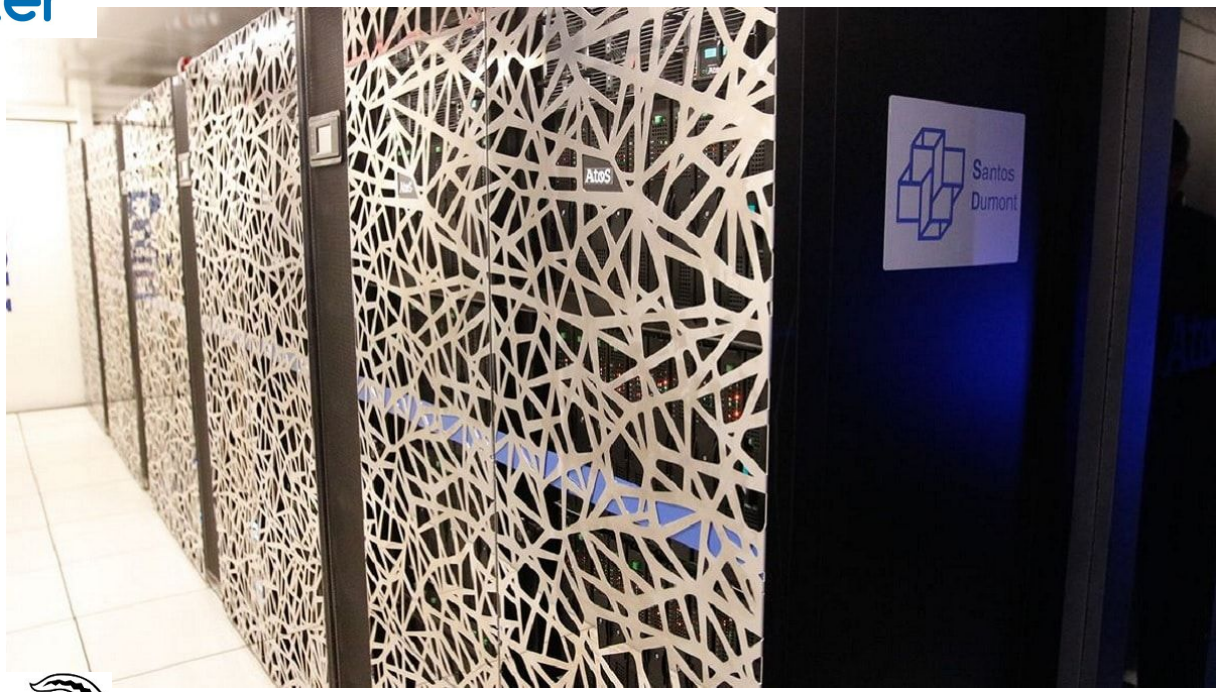
Comunicação MPI

Gerenciamento de patches

Atividades de monitoramento

Infraestrutura

- Módulos de ambiente



Infraestrutura

- Rede de alta velocidade

Infiniband

- SDR 8 Gb/s
- DDR 16 Gb/s
- QDR 32 Gb/s
- FDR 56 Gb/s
- EDR 100 Gb/s
- HDR 200 Gb/s
- NDR 400 Gbit/s
- XDR 1000 Gbit/s (2023)

Ethernet

- 10Gbit/s
- 25Gbit/s
- 50Gbit/s
- 100Gbit/s
- 200Gbit/s
- 400Gbit/s
- 800Gbit/s (previsão)
- 1600 Gbit/s (previsão)

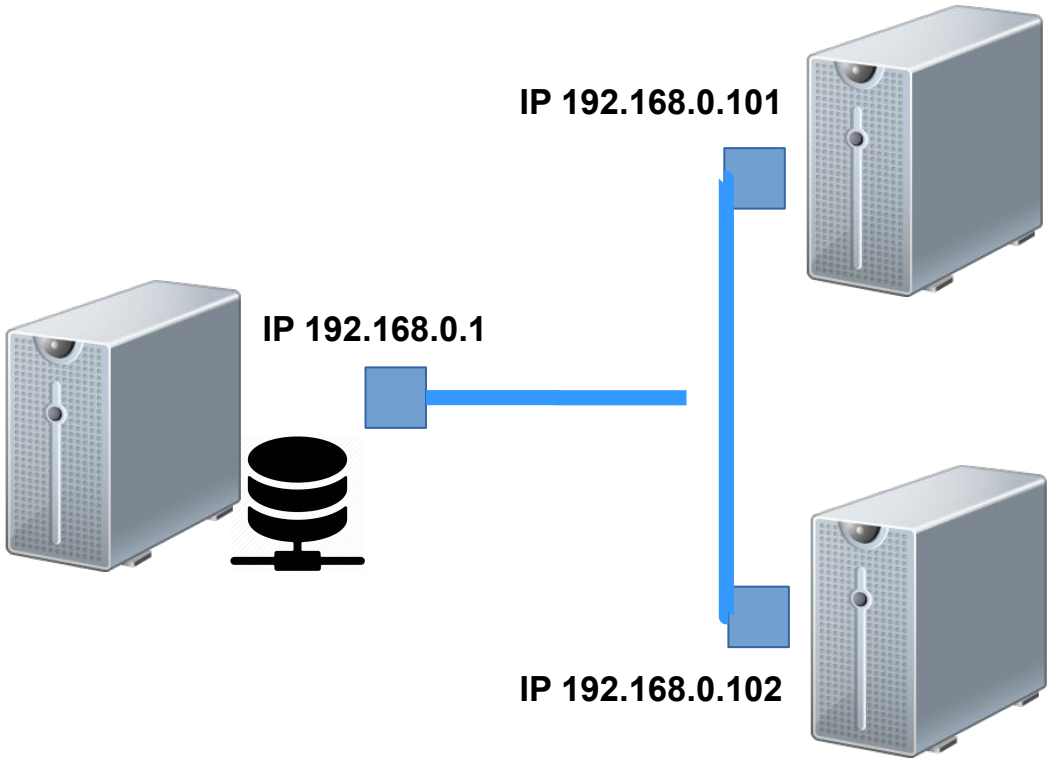
Infraestrutura do Lab

- VirtualBox
  - 3 máquinas virtuais
  - CentOS 7.9
- Configurações existentes
  - Cliente e Servidor NIS
  - Cliente e Servidor NFS
  - Cliente NTP
  - SSH HostbasedAuthentication

Laboratório de Atividades

login.local

- Servidor NIS
- Servidor NFS
- Slurmctld
- Slurmdbd
- MariaDB



node1.local

- Cliente NIS
- Cliente NFS
- Slurmd

node2.local

- Cliente NIS
- Cliente NFS
- Slurmd

MC-SD07. Introdução à configuração e gerenciamento de Clusters

- Download do material
- https://www.lncc.br/~brunoaf/MC-SD07-slide.zip
- https://www.lncc.br/~brunoaf/MC-SD07-vms.ova.zip

  Ou

- http://www.cenapad-rj.lncc.br/tutoriais/materiais-hpc/semana-sdumont/

- Extrair o conteúdo do arquivo e importar as vms através do VirtualBox
- Menu Arquivo > Importar Appliance...

Ferramentas para infraestrutura

- ClusterShell
  - Framework em python que permite executar comandos de forma unificada em um conjunto de máquinas.
  - Execução em paralelo
  - Suporte a grupos e node groups (nodeset class)
  - Aumenta a produtividade

Ferramentas para infraestrutura

- ClusterShell
  - Instalação a partir do repositório
  - # yum install -y clustershell
  - Arquivo de configuração dos grupos (/etc/clustershell/groups.d/local.cfg)
  - Incluir:
    - all: @nodes,login
    - nodes: node1,node2

Ferramentas para infraestrutura

- ClusterShell
  - Outros exemplos de configuração:
    - tux: node[1000-1100]
    - tux_gpu: node[2000-2050]
    - tux_mngt: node[1000-1100]-net1
  - Utilização:
    - clush -g NOME_DO_GRUPO <comando>
    - clush -w NOME_DO_HOST <comando>

Ferramentas para infraestrutura

- Environment Modules
  - Permite gerenciar o ambiente shell Linux de forma dinâmica;
  - Suporte para diversos tipos de shell, incluindo bash, ksh, zsh, sh, csh, tcsh e fish;
  - Escritos com comandos Tcl;
  - Os arquivo de módulo devem iniciar com a tag (magic cookie) #%Module seguida da versão.

    #%Module1.0

Ferramentas para infraestrutura

- Environment Modules
  - Instalação a partir do repositório
  - # yum install -y environment-modules
  - Diretório padrão dos arquivos de módulo:
    - /usr/share/Modules/modulefiles
  - Para adicionar novos diretórios ao path, editar o arquivo /usr/share/Modules/init/.modulepath

Ferramentas para infraestrutura

- Environment Modules

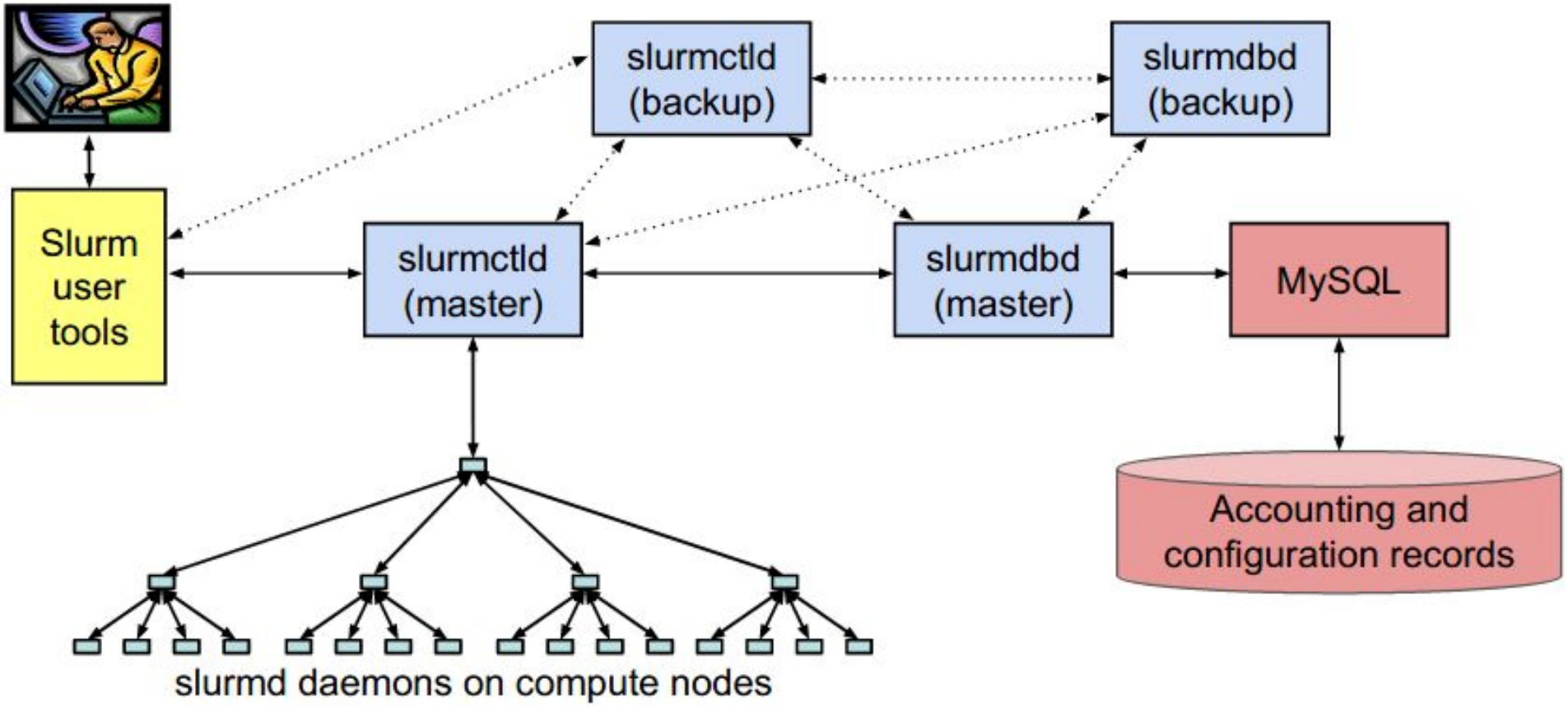
```
#%Module1.0
##
proc ModulesHelp { } {
        puts stderr
        "Informações sobre como utilizar o módulo"
}
module-whatis “Descrição do módulo: nome, versão e etc”

set rootdir /software/foo/bar
prepend-path PATH $rootdir/bin
prepend-path MANPATH $rootdir/share/man
prepend-path LD_LIBRARY_PATH $rootdir/lib
```

SLURM

- **S**imple **L**inux **U**tility for **R**esource **M**anagement
- Open source (GPL version 2)
- Tolerante a falhas
- Seguro
- Comunidade ativa
- Altamente escalável
- Suporte à diversos plugins

SLURM

- Arquitetura



SLURM

- slurmctld – Central controller daemon
- slurmdbd – Database daemon
- slurmd – Compute node daemon
- slurmstepd – Job step daemon

SLURM

- Principais arquivos de configuração

slurmdbd.conf

- Definições sobre archive/purge dos dados de accounting
- Configurações de acesso ao banco de dados
- Método de autenticação entre os daemons do slurm

slurm.conf

- Configurações gerais
- Ativação de plugins
- Parâmetros de escalonamento
- Definição dos nós
- Configuração das partições

SLURM

- slurmdbd.conf

Diretiva	Função
AuthType=auth/munge	Método de comunicação entre os componentes do SLURM
DbdAddr=localhost	Nome pelo qual DbdHost deve ser referenciado para comunicação.
DbdHost=localhost	Nome do host onde o slurmdbd estará executando.
DbdPort=6819	Porta onde o daemon slurmdbd estará executando.
SlurmUser=slurm	Usuário que executa o daemon slurmdbd
StorageType= accounting_storage/mysql	Indica como as informações de accounting serão armazenadas
StorageHost=localhost	Nome do host servidor de banco de dados
StoragePass=slurm_pass	Senha do StorageUser para acessar o banco de dados
StorageUser=slurm_user	Usuário para acessar o banco de dados
StorageLoc=slurm_acct_db	Nome do banco de dados.

SLURM

- slurm.conf

Diretiva	Função
ClusterName	Nome do cluster que será referenciado no banco de accounting
SlurmctldHost	Hostname da máquina onde o daemon do slurmctld está executando. Substitui a diretiva *ControlMachine*
AuthType	Método de autenticação entre os componentes do Slurm
SlurmdSpoolDir	Path completo do diretório onde o slurmd irá escrever as informações sobre os jobs.
Epilog	Path completo para o script que será executado assim que o job for concluído em todos os nós alocados.
Prolog	Path completo para o script que será executado antes do job ser iniciado em todos os nós alocados.

SLURM

- slurm.conf

#SCHEDULING

Diretiva	Função
SchedulerType	Define qual será a política de escalonamento de recursos (backfill, builtin e hold)

#LOGGING

Diretiva	Função
SlurmctldDebug	Nível de detalhes no log do daemon slurmctld
SlurmctldLogFile	Arquivo de log do daemon slurmctld
SlurmdDebug	Nível de detalhes no log do daemon slurmd
SlurmdLogFile	Arquivo de log do daemon slurmd

SLURM

- slurm.conf

#NODES

Diretiva	Função
NodeName	Nome do nó que será referenciado pelo Slurm
NodeAddr	Nome utilizado para comunicação (ex: no01-ib ou IP)
Sockets	Número de sockets físicos do nó
CoresPerSocket	Total de núcleos de um único processador (socket)
RealMemory	Tamanho total de memória em megabytes
CpuSpecList	Id das CPUs reservadas para uso do sistema.
MemSpecLimit	Total de memória (MB) reservada para uso do sistema
State	Status do nó ao iniciar o slurm. Default é UNKNOWN
Weight	Prioridade do nó para escalonamento das tarefas

SLURM

- slurm.conf (Nodes)

NodeName=tux[1000-1200] *Weight*=1 *Sockets*=2 *CoresPerSocket*=8 *ThreadsPerCore*=1 *RealMemory*=64000 *State*=UNKNOWN

NodeName=tux[2000,3000-3020] *Sockets*=2 *CoresPerSocket*=24 *ThreadsPerCore*=1 *RealMemory*=128000 *State*=UNKNOWN

SLURM

- slurm.conf

#PARTITIONS

Diretiva	Função
PartitionName	Nome da partição (fila)
Allow[Accounts,Groups]	Lista de account e grupos habilitados a usar a fila
Hidden	A partição e seus jobs ficam ocultos aos comandos
Nodes	Lista de nós associados a uma partição
Default	Indica que será a partição padrão
MaxTime / DefaultTime	Limite máximo de tempo de execução dos jobs
MinNodes / MaxNodes	Minimo e máximo de nós que podem ser requisitados por job
Shared	Permite o uso compartilhado dos recursos
State	Define o status da partição (Up, Down, Drain e Inactive)

SLURM

- slurm.conf (Partitions)

PartitionName=cpu *Nodes*=tux[1000-1100] *Shared*=NO *Default*=YES *State*=UP *MinNodes*=10 *DefaultTime*=48:00:00

PartitionName=cpu_shared *Nodes*=tux[1101-1200] *Shared*=YES *Default*=NO *State*=UP *DefMemPerCPU*=4000 *MaxNodes*=10

PartitionName=gpu *Nodes*=tux[2000,3000] *Shared*=No *State*=UP *AllowGroups*=grupo1 *Hidden*=YES

SLURM

- Principais comandos

Submissão	Modificação	Informação	Accounting
salloc sbatch srun	scontrol scancel	squeue sinfo sstat sview smap	sacct sacctmgr sreport

SLURM

- Aplicando as alterações
  - Adicionar ou remover nós
  - > reiniciar slurmctld e slurmd
  - Alterações nas partições e demais configurações
  - > editar os arquivos de configuração
  - > scontrol reconfigure
  - Adição ou alterações nas associações
  - > não requer ação, são aplicadas imediatamente

SLURM

- Instalação

Super Quick Start

https://slurm.schedmd.com/quickstart_admin.html

- Recomenda o uso de contas de usuários e relógios sincronizados
- Utilizar o **MUNGE** para autenticação dos daemons do SLURM
- Baixar o código-fonte
- Descompactar
- Compilar
- Instalar
- Iniciar os daemons

SLURM

- Pré-requisitos
  - **MUNGE**
    - Serviço de autenticação criado especialmente para alta escalabilidade em ambientes de HPC;
    - Utiliza uma chave criptografada para autorizar a comunicação entre processos de máquinas diferentes;
    - Código-fonte disponível em https://dun.github.io/munge/ ou pelo repositório epel-release do CentOS;
    - Deve ser instalado em todos os nós do cluster e a chave deve ser idêntica em todas as máquinas.

SLURM

- Pré-requisitos
  - **MUNGE**
    - Instalar o munge em todos os nodes

```
# clush -g all yum install -y munge munge-libs munge-devel
```

    - Gerar uma chave e distribuí-la em todos os nodes

```
# dd if=/dev/urandom bs=1 count=1024 > /etc/munge/munge.key
# chmod 0600 /etc/munge/munge.key

# clush -g nodes -p -c /etc/munge/munge.key
# clush -g all chown munge:munge /etc/munge/munge.key
```

SLURM

- Pré-requisitos
  - **MUNGE**
    - Iniciar o serviço e testar a comunicação entre os nós

```
# clush -g all systemctl enable munge
# clush -g all systemctl start munge
```

Testando a comunicação:

```
# munge -n | unmunge
# munge -n | ssh node1 unmunge
# munge -n | ssh node2 unmunge
```

SLURM

- Pré-requisitos
  - **MySQL/MariaDB**
    - Será utilizado como banco de dados do sistema de accounting
    - Utilizar o repositório do mariadb

```
# curl -sS https://downloads.mariadb.com/MariaDB/mariadb_repo_setup | bash
```

Login/Service node

- MariaDB-client
- **MariaDB-shared**
- MariaDB-devel
- MariaDB-server
- MariaDB-backup

Computer nodes

- MariaDB-client
- **MariaDB-shared**
- MariaDB-devel

SLURM

- Pré-requisitos
  - **MySQL/MariaDB**
    - Instalação do MariaDB no Service Node
      - login# yum install -y MariaDB-client MariaDB-shared MariaDB-devel MariaDB-server MariaDB-backup
    - Instalação das libs nos computer nodes
      - nodes# yum install -y MariaDB-client MariaDB-shared MariaDB-devel

SLURM

- Pré-requisitos
  - **MySQL/MariaDB**
    - Ajustar configurações do banco de dados para os valores recomendados*.
    - Criar o arquivo /etc/my.cnf.d/innodb.cnf com o conteúdo abaixo:
      - login# vi /etc/my.cnf.d/innodb.cnf
      - [mysqld]
      - innodb_buffer_pool_size=1024M
      - innodb_log_file_size=64M
      - innodb_lock_wait_timeout=900

(*) https://slurm.schedmd.com/accounting.html

SLURM

- Pré-requisitos
  - **MySQL/MariaDB**
    - Iniciar o serviço e habilitar para inicializar automaticamente.
    - Criar o banco de dados e o usuário para acesso ao mysql

```
login# mysql
mysql> create database slurm_acct_db;
mysql> create user 'slurm'@'localhost' identified by 'slurm_pass';
mysql> grant all on slurm_acct_db.* TO 'slurm'@'localhost';
mysql> FLUSH PRIVILEGES;
mysql> exit;
```

SLURM

- Pré-requisitos
  - **MySQL/MariaDB**
    - Testando o acesso:

```
login# mysql -u slurm -D slurm_acct_db -p
```

SLURM

- Instalação
  - Criar o usuário e o grupo slurm
  - Baixar o código-fonte
  - Gerar os arquivos RPM
  - Instalar os pacotes
  - Criar estrutura de diretórios de log e spool
  - Configurar e iniciar os serviços

SLURM

- Instalação – Criando usuário e grupo slurm

Criando o grupo:
clush -g all “groupadd -g 991 slurm”

Criando o usuário:
clush -g all ‘useradd -m -c “Slurm workload manager” -d /var/lib/slurm -u 991 -g slurm -s /bin/bash slurm’

Verificando:
#clush -g all getent passwd slurm
#clush -g all getent group slurm

SLURM

- Instalação – Baixar o código-fonte e gerar o rpm

Baixando o fonte:

login# wget https://download.schedmd.com/slurm/slurm-19.05.5.tar.bz2

Criando os pacotes

login# rpmbuild -ta slurm-19.05.5.tar.bz2

login# mkdir -p /scratch/app/slurm-rpms

login# cp /root/rpmbuild/RPMS/x86_64/slurm-* /scratch/app/slurm-rpms

SLURM

- Instalação – service/login node
  - Instalar os pacotes rpm

```
login# cd /scratch/app/slurm-rpms
login# yum --nogpgcheck localinstall -y \
    slurm-19.05.5-1.el7.x86_64.rpm \
    slurm-perlapi-19.05.5-1.el7.x86_64.rpm \
    slurm-slurmctld-19.05.5-1.el7.x86_64.rpm \
    slurm-slurmdbd-19.05.5-1.el7.x86_64.rpm \
    slurm-example-configs-19.05.5-1.el7.x86_64.rpm \
    slurm-devel-19.05.5-1.el7.x86_64.rpm \
    slurm-contrib-19.05.5-1.el7.x86_64.rpm
```

SLURM

- Instalação – computer node
  - Instalar os pacotes rpm

```
nodes# cd /scratch/app/slurm-rpms
nodes# yum --nogpgcheck localinstall -y  \
       slurm-19.05.5-1.el7.x86_64.rpm  \
       slurm-perlapi-19.05.5-1.el7.x86_64.rpm  \
       slurm-slurmd-19.05.5-1.el7.x86_64.rpm  \
       slurm-pam_slurm-19.05.5-1.el7.x86_64.rpm
```

SLURM

- Instalação – Estrutura de diretórios de log e spool

```
# clush -g all “mkdir -p /var/log/slurm/ /var/spool/slurm/d /var/spool/slurm/ctld”
# clush -g all “chown -R slurm:slurm /var/log/slurm /var/spool/slurm”
```

SLURM

- Configuração do slurmdbd.conf (login)

Usar o arquivo de exemplo como base para configuração

login# cp /etc/slurm/slurmdbd.conf.example /etc/slurm/slurmdbd.conf

login# vi /etc/slurm/slurmdbd.conf

Diretiva	Valor
AuthType	auth/munge
DbdAddr	localhost
DbdHost	localhost
DebugLevel	verbose
SlurmUser	slurm
StorageType	accounting_storage/mysql
StorageHost	localhost
StoragePass	slurm_pass
StorageUser	slurm
StorageLoc	slurm_acct_db
LogFile	/var/log/slurm/slurmdbd.log
PidFile	/var/run/slurmdbd.pid

SLURM

- Iniciar o serviço do slurmdbd e verificar os logs

```
login# systemctl start slurmdbd
```

O log do slurmdbd deve apresentar uma saída semelhante a descrita abaixo:

```
login# cat /var/log/slurm/slurmdbd.log
[data] Accouting storage MYSQL plugin loaded
[data] slurmdbd version 19.05.5 started
```

SLURM

- Configuração do slurm.conf

Diretiva	Valor
ClusterName	verao21
SlurmctldHost	login
SlurmdUser	slurm
AuthType	auth/munge
StateSaveLocation	/var/spool/slurm/ctld
SlurmdSpoolDir	/var/spool/slurm/d
SchedulerType	sched/backfill
SlurmctldDebug	info
SlurmctldLogFile	/var/log/slurm/slurmctld.log
SlurmdDebug	info
SlurmdLogFile	/var/log/slurm/slurmd.log

Usar o arquivo de exemplo como base para configuração

login# cp /etc/slurm/slurm.conf.example /etc/slurm/slurm.conf

login# vi /etc/slurm/slurm.conf

SLURM

- Configuração do slurm.conf

Nodes e Partitions

#Nodes
NodeName=node[1-2] CPUs=1 State=UNKNOWN
#Partitions
PartitionName=teste Nodes=node1 Default=Yes MaxTime=INFINITE State=UP

SLURM

- Iniciar o slurmctld no nó de login

```
login# systemctl enable slurmctld
login# systemctl start slurmctld
```

- Propagar o arquivo de configuração e iniciar o slurmd nos nodes

```
login# clush -w node[1-2] mkdir /etc/slurm     *****
login# clush -w node[1-2] -c /etc/slurm/slurm.conf

login# clush -w node[1-2] systemctl enable slurmd
login# clush -w node[1-2] systemctl start slurmd
```

SLURM

- Testando a configuração
  - Dump da configuração atual:

    $ scontrol show config

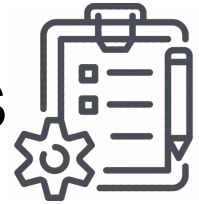
  - Obtendo as informações dos nodes

    login# clush -w node1 slurmd -C

NodeName=node1 CPUs=1 Boards=1 SocketsPerBoard=1 CoresPerSocket=1 ThreadsPerCore=1 RealMemory=360

SLURM

- Verificando o status da partição e dos nodes



```
login# sinfo
PARTITION   AVAIL   TIMELIMIT   NODES    STATE   NODELIST
teste*         up    infinite       1     idle    node1
```

  - Submetendo um job

```
user01@login~$ srun -N 1 -p teste hostname
node1
```

Controle de acesso e accounting

- Hierarquia de accounting
  1) Cluster
  2) Account
  3) User
  4) Resource

A combinação desses elementos possibilita criar diversas associações que permitem a definição de limites no uso dos recursos

Controle de acesso e accounting

- Cluster
  - **Name**: nome do cluster
- Account
  - **Cluster**: nome do cluster associado
  - **Description**: texto descritivo da account
  - **Name**: nome da account
  - **Organization**: nome da organização
  - **Parent**: transforma a account em uma filha da account informada

Controle de acesso e accounting

- User
  - **Account**: nome da account do usuário
  - **AdminLevel**: nível de privilégio do usuário (none, operator e admin)
  - **Cluster**: nome do cluster do usuário
  - **DefaultAccount**: nome da account default
  - **Name**: nome do usuário
  - **Partition**: nome da partição que será utilizada para criar uma associação

Controle de acesso e accounting

- Resources
  - **Fairshare**: valor inteiro que indica a prioridade
  - **MaxJobs**: número total de jobs em execução
  - **MaxSubmitJobs**: número máximo de jobs submetidos
  - **MaxCPUs**: número máximo de CPUs alocadas por job
  - **MaxNodes**: número máximo de nós alocados por job
  - **MaxWall**: walltime máximo de um job

Controle de acesso e accounting

- Habilitando

  Arquivo slurm.conf

  - AccountingStorageType

  - **accounting_storage/none:** não registra as informações de accounting dos jobs. Essa é a opção padrão.

  - **accounting_storage/slurmdbd**: armazena as informações de accounting em um banco de dados através do daemon do slurmdbd.

  - **accounting_storage/filetext**: grava as informações de accounting diretamente em um arquivo.

Controle de acesso e accounting

- Habilitando

  Arquivo slurm.conf

  - AccountingStorageEnforce
    - **associations**: impede que os usuários executem jobs se não existir uma associação no banco de dados.
    - **limits**: aplica limites as associações existentes no banco de dados. Ao habilitar essa opção, a opção associations também é definida.
    - **qos**: exige que todos os jobs informem um QOS válido.
    - **safe**: só aceitará jobs que a associação ou QOS tenham um GrpCPUMins (tempo máximo de CPU) definido. Implica em associations e limits habilitados.
    - **wckeys**: previne que usuários executem jobs com uma chave (WCKey) que não possuem acesso. Implica em associations habilitada.

Controle de acesso e accounting

- Comando sacctmgr
  - sacctmgr [opções] [comando]

sacctmgr add <ENTRY> <SPECS>

sacctmgr list <ENTRY> <SPECS>

sacctmgr modify <ENTRY> where <SPECS> set <SPECS>

sacctmgr delete <ENTRY> where <SPECS>

Controle de acesso e accounting

- Criando um cluster
  - $ sacctmgr add cluster verao21
- Criando um account
  - $ sacctmgr add account alunos Description=”Alunos”
- Criando um user
  - $ sacctmgr create user name=aluno123 account=alunos

Controle de acesso e accounting

- Alterando elementos
  - $ sacctmgr modify account where name=alunos set Description=”Alunos programa de verao 2021”
  - $ sacctmgr modify user where name=aluno123 set DefaultAccount=none
  - $ sacctmgr modify user where name=aluno123 set MaxJobs=10

Controle de acesso e accounting

- Removendo elementos
  - $ sacctmgr remove user aluno123 where account=alunos
- Outros comandos
  - $ sacctmgr show configuration
  - $ sacctmgr show stats
  - $ sacctmgr list user aluno123
  - $ sacctmgr list assoc user=aluno123

Controle de acesso e accounting

- Configuração do slurm.conf

Diretiva	Valor
AccountingStorageEnforce	limits
AccountingStorageType	accounting_storage/slurmdbd

- Propagar a configuração para os nodes e reiniciar os daemons

```
login# clush -w node[1-2] -c /etc/slurm/slurm.conf
login# clush -w node[1-2] systemctl start slurmd
login# systemctl restart slurmctld
```

Controle de acesso e accounting

- Verifique os logs do slurmctld

Error: slurmdbd: Issue with call DBD_REGISTER_CTLD(1434): 4294967295(This cluster hasn’t been added to accounting yet)
fatal: You need to add this cluster to accounting if you want to enforce associations, or no jobs will ever run.

Controle de acesso e accounting

- Ao habilitarmos o controle de limites de recursos devemos ter, pelo menos, a associação do cluster cadastrada

```
...
fatal: You need to add this cluster to accounting if you want to enforce
associations, or no jobs will ever run.
```

- O nome do cluster deve ser idêntico ao valor do parâmetro ClusterName do arquivo slurm.conf

```
$ sacctmgr add cluster verao21
```

Controle de acesso e accounting

- Reinicie o daemon do slurmctld e verifique o arquivo de log.

  # systemctl restart slurmctld

```
login# cat /var/log/slurm/slurmctld.log

[data] slurmctld version 19.05.5 started on cluster verao21
...
```

Controle de acesso e accounting

- Submeta um job de teste

  user01@login~$ srun -N 1 -p teste hostname

  srun: error: Unable to allocate resources: Invalid account or account/partition combination specified

- Verifique os acessos do usuário user01

  $ sacctmgr list user user01

  $ sacctmgr list assoc user=user01

Controle de acesso e accounting

- Cadastrando um account sem restrições

```
# sacctmgr create account alunos
```

- Cadastrando um account com limitações
  - Número máximo de nós por job: 1
  - Número máximo de jobs submetidos: 2
  - Tempo máximo de execução dos jobs: 60 segundos

```
# sacctmgr create account restrito MaxNodes=1 MaxSubmitJobs=2
MaxWall=00:01:00
```

Controle de acesso e accounting

- Criando associações entre usuários e accounts

Usuário	Account	Restrições
user01	alunos	-
user02	alunos	Default Account = restrito

```
# sacctmgr create user user01 account=alunos

# sacctmgr create user user02 account=alunos DefaultAccount=restrito
```

Controle de acesso e accounting

- Submeta com as contas de usuário criadas.

```
# sudo -u user01 srun -N 1 -p teste sleep 200&

# sudo -u user01 srun -N 1 -p teste sleep 200&
```

- Liste as informações da fila e da partição

```
# squeue -p teste

# sinfo -p teste
```

Controle de acesso e accounting

- Verifique as informações dos jobs

```
# scontrol show job JOBID1

# scontrol show job JOBID2
```

Atenção para os campos account e timelimit

Atividades de administração

- Submeta o job abaixo:

```
# sudo -u user02 srun -N 2 -p teste sleep 100&
```

- Liste as informações da fila e dos jobs

```
# squeue -p teste
# scontrol show job JOBID
```

Atividades de administração

- O comando scontrol permite modificar os recursos solicitados por um job que está na fila.

```
# sudo -u user02 scontrol update jobid=JOBID propriedade=valor ...
```

```
# sudo -u user02 scontrol update jobid=JOBID ReqNodes=1
NumNodes=1-1 NumCpus=1 NumTasks=1

# scontrol update jobid=JOBID TimeLimite=+01:00:00  (apenas root)
```

man scontrol
Sessão: SPECIFICATION FOR UPDATE COMMAND, JOBS

Atividades de administração

- Gerenciando nodes
  - Coloca um nó em manutenção

    # scontrol update NodeName=node1 State=Drain Reason=”Motivo”

    # sinfo -p teste -R

  - Retornando um nó para a fila

    # scontrol update NodeName=node1 State=Resume

    # sinfo -p teste

Atividades de administração

- Partitions (filas)
  - Alterando configurações
  - Uso exclusivo x compartilhado
  - Tempo default
  - Status (up, down, drain, inactive)

scontrol update partition PartitionName=NOME chave=valor

Atividades de administração

- Partitions (filas)
  - As alterações feitas com o comando scontrol não são permanentes;
  - Caso o daemon do slurmctld for reiniciado, serão carregas as configurações contidas no arquivo de configuração do slurm.

Atividades de administração

- Reservations
  - Garante uma quantidade de recursos por um período de tempo;
  - Não necessita criar associação;
  - Restrição por usuário ou account;
  - São removidas automaticamente.

Atividades de administração

- Reservations

```
# scontrol create reservation
StartTime=2021-01-22T00:00:00 Duration=120
User=user01 Nodes=node1


# scontrol show res


# sudo -u user01 srun -N1 --reservation user01_1
hostname
```

Relatórios de accounting

- sacct

Exibe as informações sobre os jobs do banco de dados de accounting.
Permite filtros por data de inicio e fim, nome do usuário, account, partição, nós onde os jobs executaram entre outros.

sacct -a -X -N node1 -S 2021-01-01 -E 2021-01-15

Relatórios de accounting

- sreport

  Gera relatórios consolidados sobre o consumo de recursos em um determinado período.

- Utilização por usuário
- Utilização por account
- Utilização de todo o cluster
- Total de recursos ocupados por Reservations
- Top users do cluster

Relatórios de accounting

- sreport

sreport user topUser Start=2021-01-01 End=2021-02-01

sreport cluster utilization

sreport cluster AccountUtilizationByUser account=alunos