

GNU/Linux-Bacula



GNU/Linux-Bacula

Bacula

Bacula 属于一款开源的跨平台网络备份工具，它提供了基于企业级的 C/S 架构的备份恢复解决方案。

系统管理员可对数据进行备份、恢复，以及完整性验证等操作。同时，还提供了许多高级存储管理功能，使系统管理人员能够很容易的发现并恢复丢失或损坏的文件

GNU/Linux-Bacula



Bacula

当业务数据量巨大时，普通的 tar 打包方式进行低级备份并且没有相关的本地或异地的冗余备份。此时，就应该使用 rsync/lsync 或 Bacula。

Bacula 可以将数据备份到任意一个远程主机上，用户只需要对 bacula 进行简单的设置即可自动完成数据备份。



GNU/Linux-Bacula

A traditional Chinese ink wash painting is visible in the background. In the upper right corner, a dragon is depicted with its head turned back, looking towards the left. In the lower right corner, a crane stands on a small patch of ground, facing left. The painting is executed in a minimalist, expressive style with varying line weights and ink tones.

Bacula

Bacula 也可以将业务数据从服务器自动备份到存储设备上（如：磁盘阵列、磁带等）。

Bacula 的功能完全可以媲美 legato 和 veritas 等。并且因为其开源，更可以通过修改源代码方式扩展企业需要的自有功能。

GNU/Linux-Bacula

Bacula

Bacula 支持的备份方式

1. 完整备份
2. 增量备份
3. 差异备份

建议：

1. 周一至周六：增量或差异
2. 周日 / 月底 / 年底：完整性备份



GNU/Linux-Bacula

Bacula

Bacula 支持的恢复方式

1. 可以恢复某个目录、文件到指定位置，恢复时自动恢复数据的原始结构
2. 可以恢复所有数据到指定位置，恢复时自动恢复数据的原始结构

GNU/Linux-Bacula

Bacula

Bacula 支持的恢复方式

3. 可以保存恢复文件或目录的权限、属主、访问时间等属性

4. 可以恢复某个时间点的备份到指定位置，恢复时自动恢复数据的原始结构



GNU/Linux-Bacula

Bacula

Bacula 支持各种备份介质

1. 支持把备份写到磁盘
2. 支持把备份写到磁带
3. 支持把备份写到磁盘阵列
4. 支持把备份写到光盘



GNU/Linux-Bacula

Bacula

Bacula 支持多种操作系统

1. Linux
2. Unix
3. Mac
4. Windows



GNU/Linux-Bacula

Bacula

Bacula 强大的内部功能

1. 支持定时备份，无需人工干预
2. 支持终端命令控制，更加灵活
3. 支持正则表达式，可以对备份文件进行更严格的匹配
4. 支持 md5 和 SHA1 两种签名校验
5. 支持压缩备份，备份效率更高，传输更快
6. 支持报表自动绘制功能，可以自动生成备份报表和恢复报表

GNU/Linux-Bacula

Bacula

一个完整的 bacula 备份系统，由下面 5 个部分组成

1. Director Daemon：负责监听所有的备份、恢复、验证、存档事务，以及定制备份和恢复文件计划等，并将整个系统运行状况记录在一个数据库文件中。其配置文件为 bacula-dir.conf。

GNU/Linux-Bacula

Bacula

一个完整的 bacula 备份系统，由下面 5 个部分组成

2. Storage Daemon (SD) : 主要负责将数据备份到存储介质上，而在数据恢复时，负责将数据从存储介质中传送出去。其配置文件为 bacula-sd.conf 。

GNU/Linux-Bacula

Bacula

一个完整的 bacula 备份系统，由下面 5 个部分组成

3. File Daemon (FD) : 安装在需要备份数据的机器上的守护进程，在备份数据时，它负责把文件传出，在恢复数据时负责接收数据并执行恢复操作。配置文件为 bacula-fd.conf 。

GNU/Linux-Bacula

Bacula

一个完整的 bacula 备份系统，由下面 5 个部分组成

4. Console：管理控制台。可以通过这个控制台连接到 Director Daemon 进行管理备份与恢复操作。



GNU/Linux-Bacula

Bacula

一个完整的 bacula 备份系统，由下面 5 个部分组成

5. Monitor : 进程监控端。



GNU/Linux-Bacula

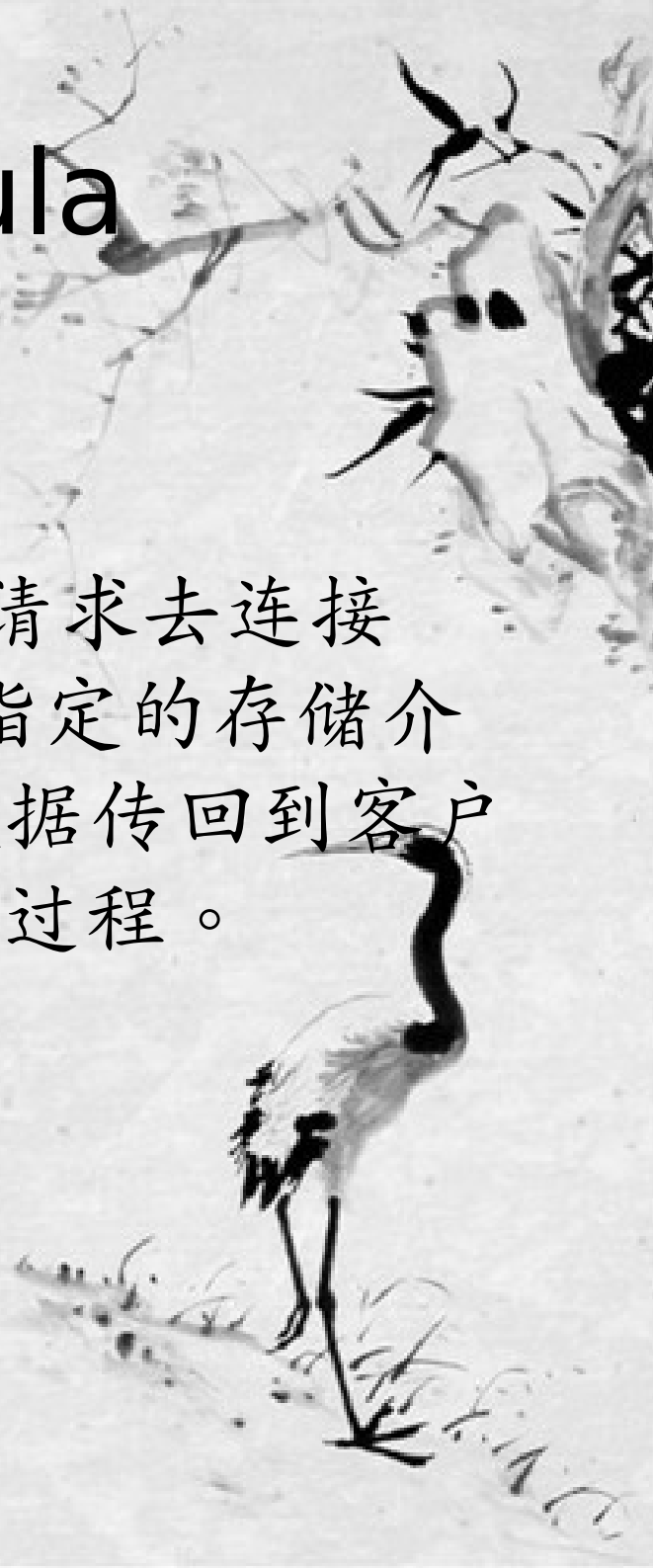
Bacula 工作流程

1. 通过 console 连接到 Director 端，备份恢复操作开始。
2. Director 端从自己的数据库中调出记录信息对存储端 SD 与客户端 FD 的任务进行协调。
3. 客户端 FD 负责验证 Director 的操作许可，如果证通过，则允许连接到存储端 SD

GNU/Linux-Bacula

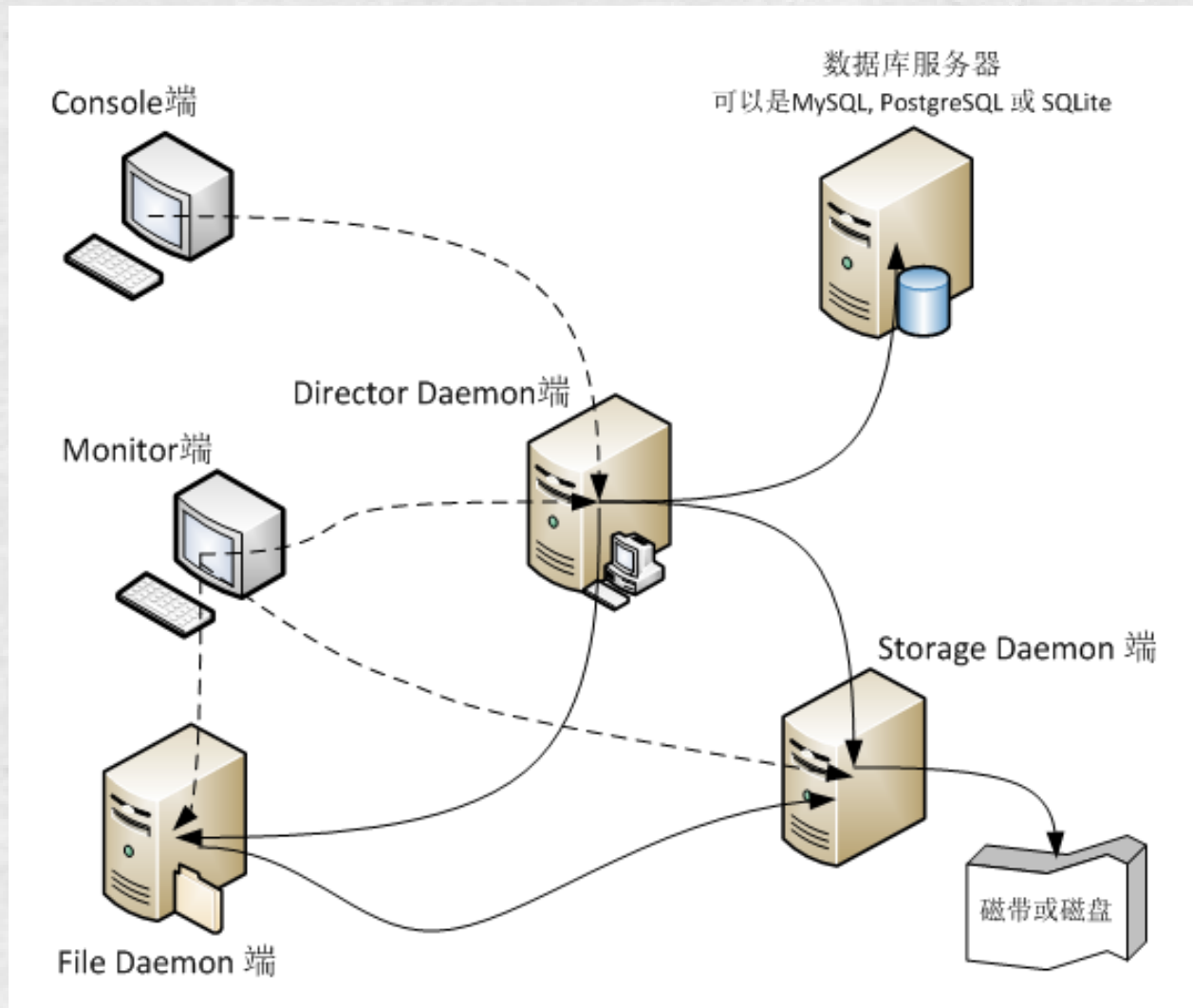
Bacula 工作流程

4. 客户端 FD 根据 Director 发出的请求去连接 SD，将 FD 端的数据备份到存 SD 指定的存储介质上，或者将 SD 端存储介质中的数据传回到客户端 FD 指定的位置上，完成备份恢复过程。



GNU/Linux-Bacula

Bacula 工作流程示意图



GNU/Linux-Bacula

Bacula 的备份恢复流程如下：

1. 通过 Console 连接到 Director 端，备份恢复操作开始。
2. Director 端从自己的数据库中调出记录信息对存储端 SD 与客户端 FD 的任务进行协调。

GNU/Linux-Bacula

Bacula 的备份恢复流程如下：

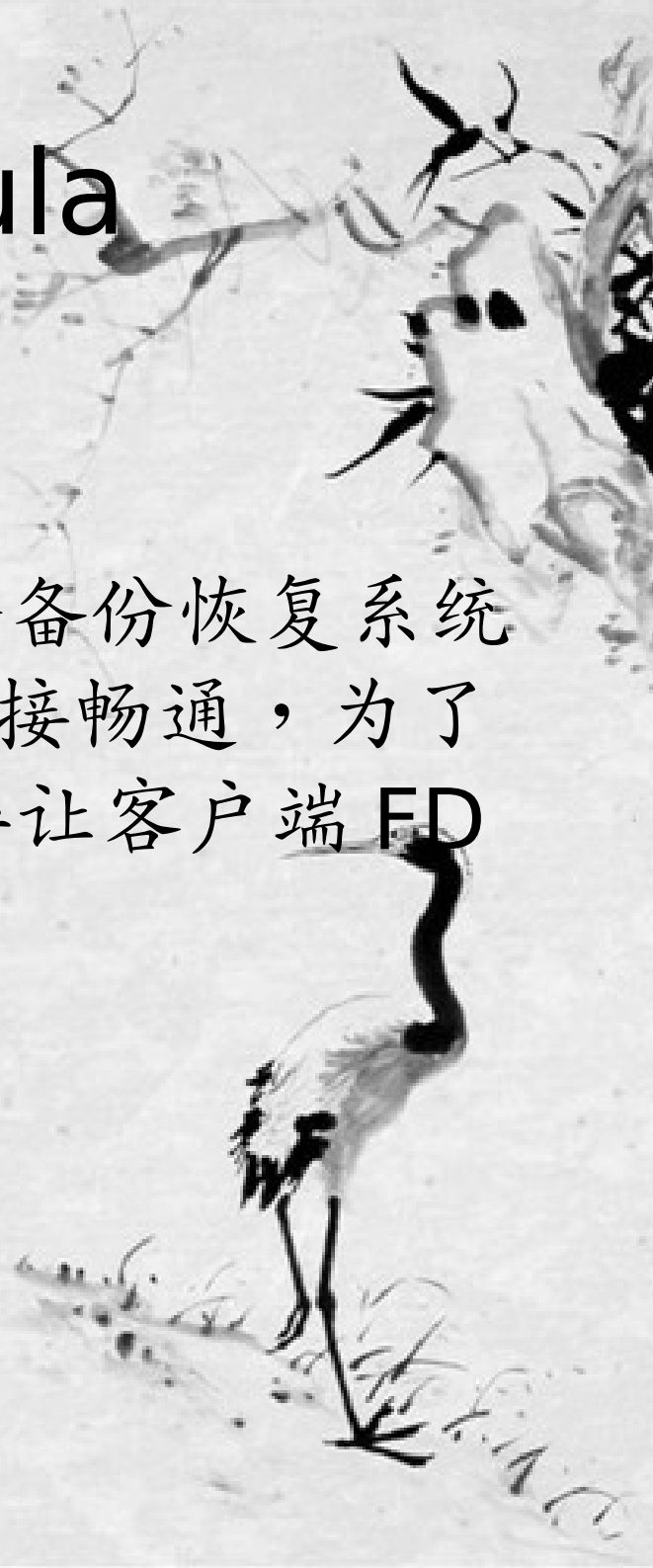
3. 客户端 FD 负责验证 Director 的操作许可，如果验证通过，则允许连接存储端 SD。

4. 客户端 FD 根据 Director 发出的请求去连接 SD，将 FD 端的数据备份到存 SD 指定的存储介质上，或者将 SD 端存储介质中的数据传回到客户端 FD 指定的位置上，完成备份恢复过程。

GNU/Linux-Bacula

Bacula 的备份恢复流程如下：

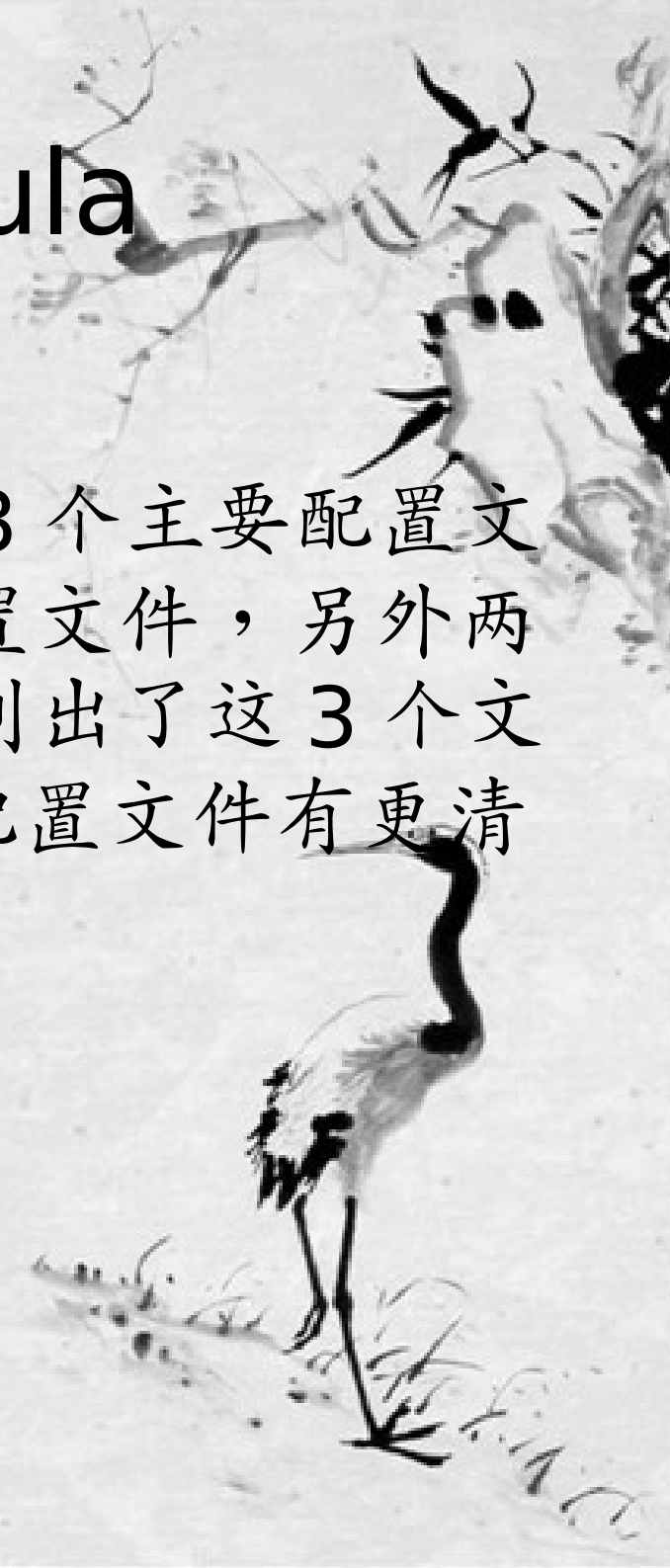
需要注意的是，在 bacula 的整个备份恢复系统中，客户端 FD 和 SD 要保证网络连接畅通，为了保证备份以及恢复数据的速度和效率让客户端 FD 和 SD 处在一个网段中。



GNU/Linux-Bacula

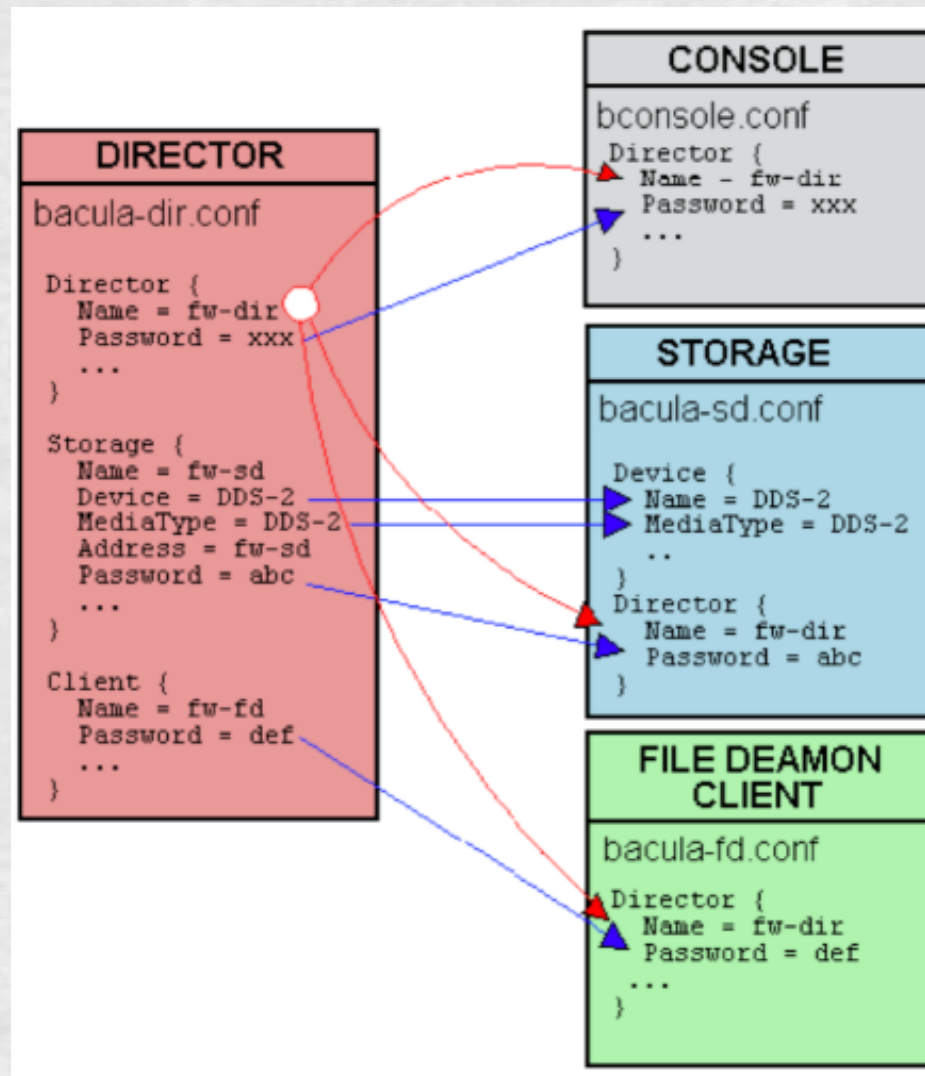
Bacula 各个组成部分的关联性

在 bacula 的 5 个组成部分中，3 个主要配置文件是相互关联的，修改任何一个配置文件，另外两个文件都要进行相应的改动。下图列出了这 3 个文件之间的相互关系。便，对这 3 个配置文件有更清晰的认识。



GNU/Linux-Bacula

Bacula 各个组成部分的关联性



GNU/Linux-Bacula

Bacula 部署结构

在实际应用中，没有必要将 5 个部分分别放在不同的服务器上，他们之间的某些部分是可以分开 / 合并部署的。常见的几种部署结构有：

1. Director 与 SD 以及 Console 合并在一台服务器上，而客户端 FD 在另外一台节点上，当然客户端 FD 可以在一台或者多台节点上

GNU/Linux-Bacula

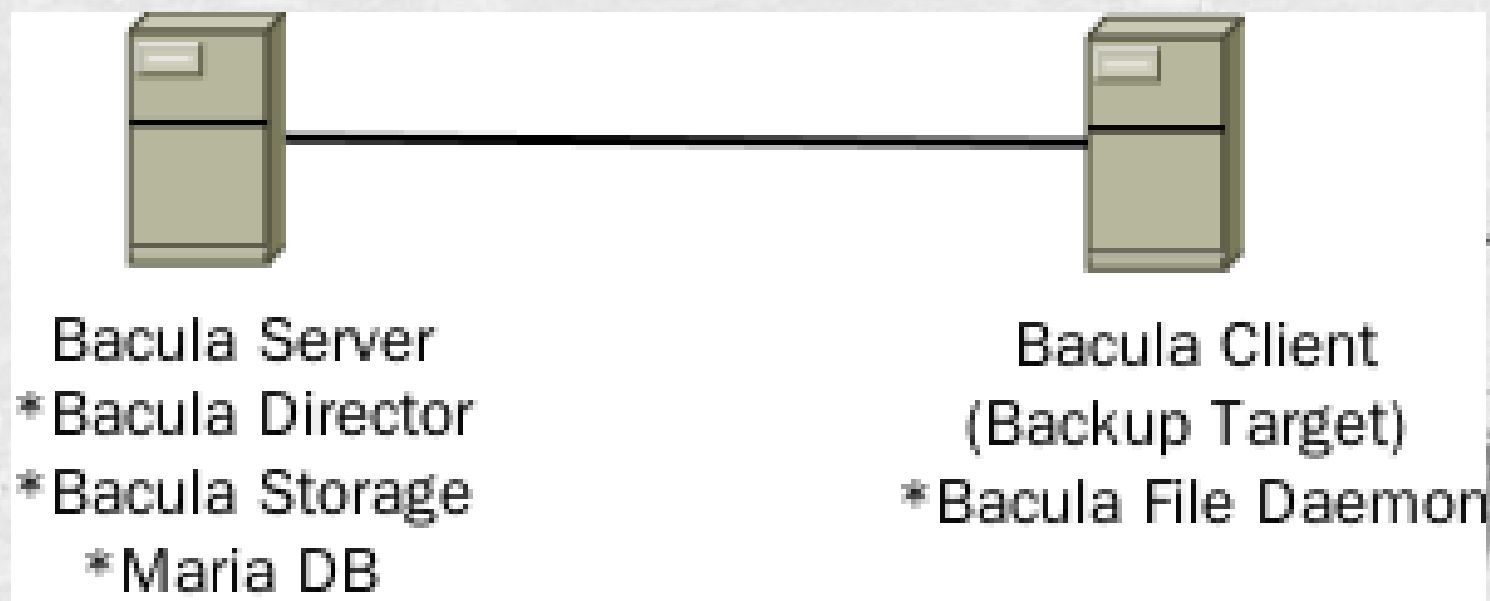
Bacula 部署结构

2.Direcotr 与 Console 在一台服务器上,SD 在一台计算机上,客户端 FD 可以在一台或多台节点上

3.Director 与客户端 FD、SD 以及 Console 都在一台计算机上,也就是服务器自己备份自己,数据保存在本机上。

GNU/Linux-Bacula

Bacula 试验示意图



GNU/Linux-Bacula

二：实现 Bacula

1: 安装并配置 MariaDB

1) 在所有后端安装 MariaDB

```
#yum install mariadb-server
```

```
#vi /etc/my.cnf
```

```
/* 在 [mysqld] 区段最后设定所支持的字符集  
character-set-server=utf8
```



GNU/Linux-Bacula

二：实现 Bacula

2) 在所有后端安装 MariaDB

```
# systemctl start mariadb
```

```
# systemctl enable mariadb
```

3) 初始化 Mariadb

```
# mysql_secure_installation
```



GNU/Linux-Bacula

二：实现 Bacula

设定数据库 root 的密码

Set root password? [Y/n] y

移除匿名账户

Remove anonymous users? [Y/n] y



GNU/Linux-Bacula

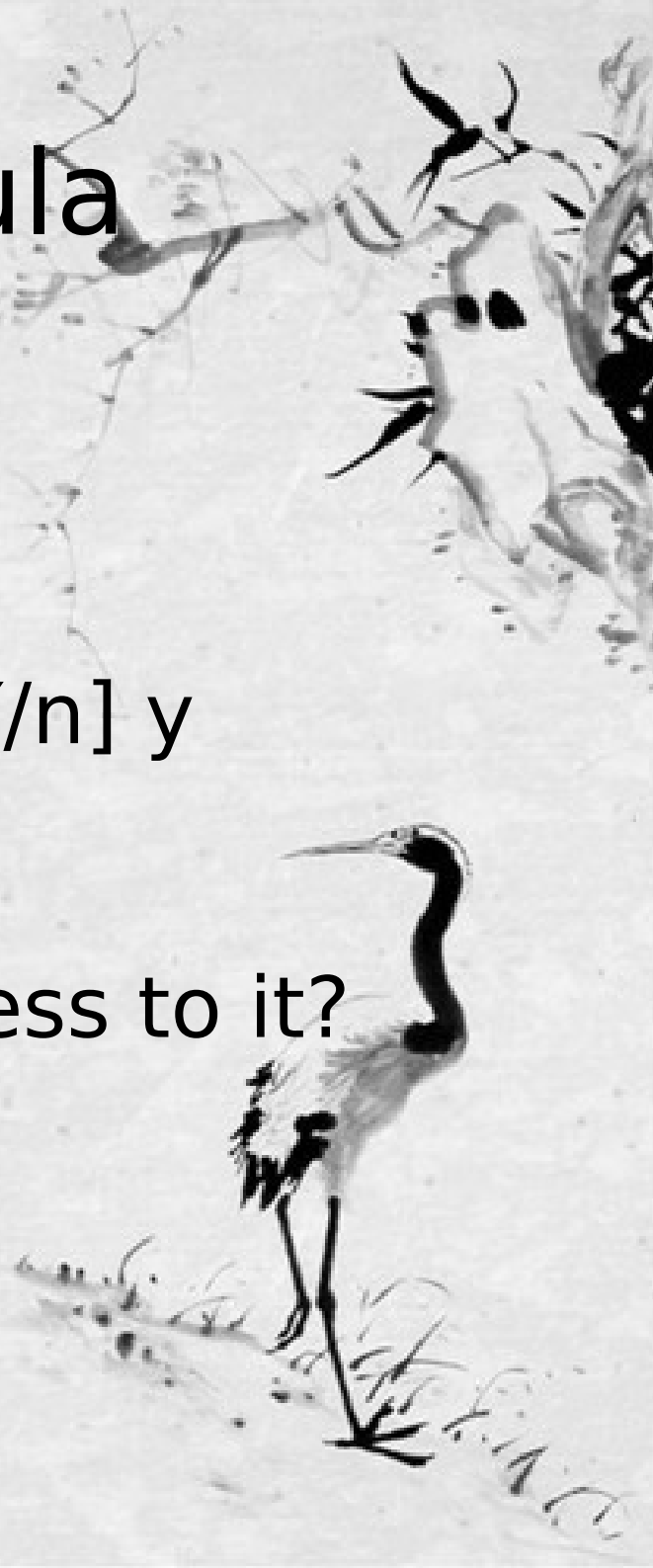
二：实现 Bacula

禁止 root 账户远程登录

Disallow root login remotely? [Y/n] y

移除测试数据库

Remove test database and access to it?
[Y/n] y



GNU/Linux-Bacula

二：实现 Bacula

重新加载权限表

Reload privilege tables now? [Y/n] y



GNU/Linux-Bacula

二：实现 Bacula

3) 登录测试

```
# mysql -u root -p  
> select user,host,password from  
mysql.user;  
> show databases;  
> exit
```



GNU/Linux-Bacula

二：实现 Bacula

2. 安装并配置 Bacula 服务

1) 安装 Bacula 软件包

```
#yum install bacula-director bacula-storage  
bacula-console -y
```

2) 将 Bacula 默认数据库改为 Mariadb(选择 mysql)

```
#alternatives --config libbaccats.so
```

GNU/Linux-Bacula

二：实现 Bacula

2. 安装并配置 Bacula 服务

3) 为 Bacula 添加数据库

```
#/usr/libexec/bacula/grant_mysql_privileges  
-p  
#  
/usr/libexec/bacula/create_mysql_database  
-p  
# /usr/libexec/bacula/make_mysql_tables -p
```

GNU/Linux-Bacula

二：实现 Bacula

2. 安装并配置 Bacula 服务

4) 设定 Bacula 用户账户密码为 password

```
# mysql -u root -p
```

```
> set password for
```

```
bacula@'%'=password('password');
```

```
> set password for
```

```
bacula@'localhost'=password('password');
```

GNU/Linux-Bacula

二：实现 Bacula

2. 安装并配置 Bacula 服务

5) 验证 Nacula 数据库及 Bacula 账户

```
> select user,host,password from  
mysql.user;  
> show databases;  
> show tables from bacula;  
> exit
```



GNU/Linux-Bacula

二：实现 Bacula

2. 安装并配置 Bacula 服务

6) 配置 Bacula Director

```
# vi /etc/bacula/bacula-dir.conf
```

```
Director {
```

```
    Name = bacula-dir <- 名称
```

```
    DIRport = 9101 <- 使用的端口
```

```
    QueryFile = "/etc/bacula/query.sql"
```



GNU/Linux-Bacula

二：实现 Bacula

2. 安装并配置 Bacula 服务

WorkingDirectory = "/var/spool/bacula"

PidDirectory = "/var/run"

Maximum Concurrent Jobs = 1 ← 定义并发
工作数量

Password = "password" ← 设定密码 (与数据库
中定义的一致)

Messages = Daemon

}

GNU/Linux-Bacula

二：实现 Bacula

2. 安装并配置 Bacula 服务

FileSet { ← 定义数据备份

Name = "Full Set" ← 数据备份的名字

Include {

Options {

signature = MD5 ← 数字签署方式为 MD5

GNU/Linux-Bacula

二：实现 Bacula

2. 安装并配置 Bacula 服务

```
/* 于 94 行下面添加压缩方法为 GZIP
    Compression = GZIP
}
// 添加 110 行, 定义所需要备份的客户端的目录
File = /home
}
```


GNU/Linux-Bacula

二：实现 Bacula

2. 安装并配置 Bacula 服务

Client { ← 定义需要备份的客户端 (FD)

 Name = bacula-fd <- 定义名字

 /*158 行定义客户端的 FQDN/IP 地址

 Address = node01.server.world

 FDPort = 9102 ← FD 所监听的端口

 Catalog = MyCatalog ← 使用哪个数据库存储
信息

GNU/Linux-Bacula

二：实现 Bacula

2. 安装并配置 Bacula 服务

/* 设定 161 行，密码为 password(与数据库一致)

Password = "password"

/* 指定保存在数据库的记录多久循环一次 (仅影响数据库记录，但不影响备份文件)

File Retention = 30 days

GNU/Linux-Bacula

二：实现 Bacula

2. 安装并配置 Bacula 服务

/* 指定 Job 的保持周期，此值应大于 File Retention 值

Job Retention = 6 months

/* 当到达指定的保持周期时，是否自动删除数据库中的记录

AutoPrune = yes

}



GNU/Linux-Bacula

二：实现 Bacula

2. 安装并配置 Bacula 服务

Storage { ← 存储定义

 Name = File ← 名字

 /* 定义存储的 FQDN/IP 地址

 Address = bsrv.niliu.edu

 SDPort = 9103 ← SD 所监听的端口



GNU/Linux-Bacula

二：实现 Bacula

2. 安装并配置 Bacula 服务

/* 设定所需密码

Password = "password"

/* 定义数据备份的存储介质为 FileStorage, 此 Device 值必须与 bacula-sd.conf 文件中 Device 区段的 Name 项目名称一致

Device = FileStorage

GNU/Linux-Bacula

二：实现 Bacula

2. 安装并配置 Bacula 服务

```
/* 存储介质类型，必须与 bacula-sd.conf 文件  
中 Device 区段的 Media Type 值一致  
Media Type = File  
}
```



GNU/Linux-Bacula

A traditional Chinese ink wash painting is visible in the background on the right side of the slide. It depicts a dragon at the top right, looking towards the left, and a crane at the bottom right, standing and facing left. The painting is done in a minimalist, expressive style with varying line weights and some washes.

二：实现 Bacula

2. 安装并配置 Bacula 服务

```
Catalog { ← 设定日志与数据库信息
    Name = MyCatalog
    # Uncomment the following line if you
    want the dbi driver
    # dbdriver = "dbi:sqlite3"; dbaddress =
    127.0.0.1; dbport =
```

GNU/Linux-Bacula

二：实现 Bacula

2. 安装并配置 Bacula 服务

```
/*236 行，设定 bacula 数据库账户的密码  
dbname = "bacula"; dbuser = "bacula";  
dbpassword = "password"  
}
```


GNU/Linux-Bacula

二：实现 Bacula

2. 安装并配置 Bacula 服务

Pool { ← 设定 Job 任务的池属性信息

Name = Default

Pool Type = Backup

Recycle = yes ← 是否重复使用

AutoPrune = yes ← 是否自动清除过期文件

/* 定义备份文件的保留时间

Volume Retention = 180days



GNU/Linux-Bacula

二：实现 Bacula

2. 安装并配置 Bacula 服务

/* 设定最多保存多少个备份文件

Maximum Volumes = 7

/* 设定每次执行备份任务创建几个备份文件

Maximum Volume Jobs = 1

/* 设定备份文件的命名格式

Label Format = Vol-

}



GNU/Linux-Bacula

二：实现 Bacula

2. 安装并配置 Bacula 服务

```
/* 注释 312-316 行
#Console {
#Name = bacula-mon
#Password =
"@@MON_DIR_PASSWORD@@"
#CommandACL = status, .status
# }
```



GNU/Linux-Bacula

二：实现 Bacula

2. 安装并配置 Bacula 服务

```
# vi /etc/bacula/bconsole.conf
```

```
Director {
```

```
    Name = bacula-dir
```

```
    DIRport = 9101
```

```
    /* 设定 Director 的 FQDN/IP 地址
```

```
    address = bsrv.niliu.edu
```

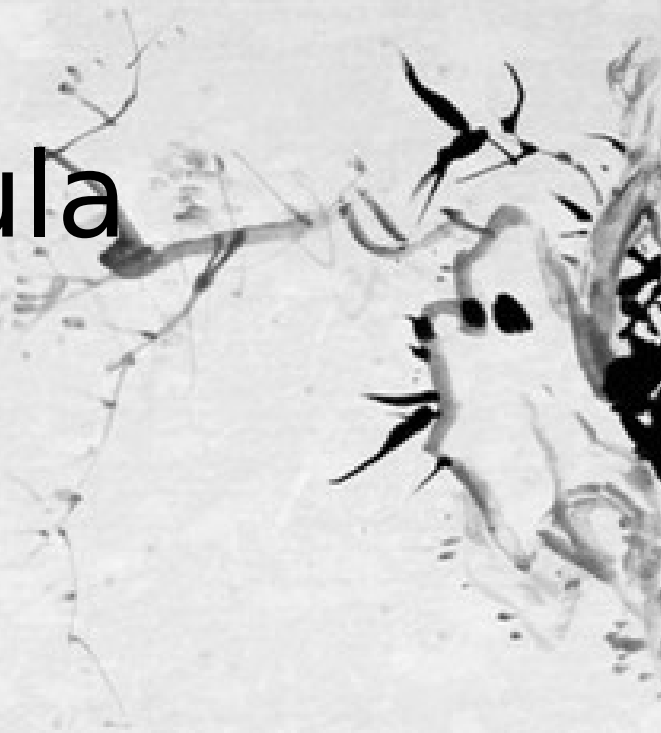


GNU/Linux-Bacula

二：实现 Bacula

2. 安装并配置 Bacula 服务

```
/* 指定 Director 的 password  
Password = "password"  
}
```



GNU/Linux-Bacula

二：实现 Bacula

2. 安装并配置 Bacula 服务

```
# vi
```

```
/usr/libexec/bacula/make_catalog_backup.pl
```

```
/*109 行添加 mysqldump 的相关参数
```

```
exec("HOME='$wd' mysqldump -f -u  
$args{db_user} -p $args{db_password}  
--opt $args{db_name} > '$wd/  
$args{db_name}.sql'");
```

GNU/Linux-Bacula

二：实现 Bacula

2. 安装并配置 Bacula 服务

```
# systemctl start bacula-dir
```

```
# systemctl enable bacula-dir
```



GNU/Linux-Bacula

二：实现 Bacula

2. 安装并配置 Bacula 服务

7) 配置 Bacula 存储

```
# vi /etc/bacula/bacula-sd.conf
```

```
Director {
```

```
    Name = bacula-dir
```

```
    /* 设定 26 行，设置 Storage 所需的口令密码
```

```
    Password = "password"
```

```
}
```



GNU/Linux-Bacula

二：实现 Bacula

2. 安装并配置 Bacula 服务

```
/* 注释 33-37 行
#Director {
#Name = bacula-mon
#Password =
"@@MON_SD_PASSWORD@@"
#Monitor = yes
# }
```



GNU/Linux-Bacula

二：实现 Bacula

2. 安装并配置 Bacula 服务

7) 配置 Bacula 存储

```
# systemctl start bacula-sd
```

```
# systemctl enable bacula-sd
```



GNU/Linux-Bacula

二：实现 Bacula

3. 安装并配置 Bacula 客户端

1) 安装 Bacula 客户端软件

```
# yum -y install bacula-client bacula-console
```

2) 配置客户端软件

```
# vi /etc/bacula/bacula-fd.conf
```



GNU/Linux-Bacula

二：实现 Bacula

3. 安装并配置 Bacula 客户端

```
Director {  
    Name = bacula-dir  
    /*15 行：指定连接 Director 的口令密码  
    Password = "password"  
}
```



GNU/Linux-Bacula

二：实现 Bacula

3. 安装并配置 Bacula 客户端

```
/* 注释 22-26 行
#Director {
#Name = bacula-mon
#Password =
"@@MON_FD_PASSWORD@@"
#Monitor = yes
#}
```



GNU/Linux-Bacula

二：实现 Bacula

3. 安装并配置 Bacula 客户端

```
# vi /etc/bacula/bconsole.conf
Director {
    Name = bacula-dir
    DIRport = 9101
    /* 设定服务器的 FQDN/IP
    address = bsrv.niliu.edu
```



GNU/Linux-Bacula

二：实现 Bacula

3. 安装并配置 Bacula 客户端

```
/* 指定 Director 所需的 password  
Password = "password"  
}
```

```
# systemctl start bacula-fd  
# systemctl enable bacula-fd
```



GNU/Linux-Bacula

三：实现 Bacula 备份

1. 登陆 Bacula 服务器

//* 进入 bacula 控制台

#bconsole

* ← 是 bacula 命令提示符

//* 创建一个备份卷

*label



GNU/Linux-Bacula

三：实现 Bacula 备份

//* 输入卷的名字

Enter new Volume name: Vol-20160304

Select the Pool (1-3): 2

//* 执行备份命令

*run



GNU/Linux-Bacula

三：实现 Bacula 备份

/* 选择需要备份的资源

Select Job resource (1-3): 1

/* 开始备份

OK to run? (yes/mod/no): yes

/* 查看备份信息

*messages



GNU/Linux-Bacula

三：实现 Bacula 备份

//* 退出 Bacula
***exit**

本地查看备份文件
#ls /tmp
.....

Vol-20150721



GNU/Linux-Bacula

四：恢复 Bacula 备份

```
#bconsole
```

```
//* 执行恢复命令
```

```
*restore
```

```
//* 选择恢复选项
```

```
Select item: (1-13): 5
```



GNU/Linux-Bacula

四：恢复 Bacula 备份

/* 进入 bconsole 命令行

\$ls ← 查看备份信息

/* 标记所要恢复的备份

\$mark home

/* 查看被标记的备份

\$lsmark



GNU/Linux-Bacula

四：恢复 Bacula 备份

/* 开始恢复

\$done

OK to run? (yes/mod/no): yes

/* 确认恢复信息

*messages

*exit



GNU/Linux-Bacula

四：恢复 Bacula 备份

//* 确认备份已经恢复到本地

```
# ls -laR /tmp/bacula-restores
```

