

3 MySQL主流版本版本特性与部署安装



1.MySQL单实例环境基础使⽤

1.1.基础使⽤

0.启动

1

#第一种启动方式

mysqld\_safe --defaults-file=/data/mysql3306/etc/my.cnf --user=mysql &

#第二种启动方式

mysqld --defaults-file=/data/mysql3306/etc/my.cnf --user=mysql &

2

3

4

5

1.登⼊

客户端⽅式登⼊

1

mysql -uroot -p -hlocalhost -P 3306

接⼝⽅式登⼊

mysql -uroot -p -S /data/mysql3306/tmp/mysql.sock

1

创建⼀个新⽤户

1

mysql> create user qianlong@'localhost' identified by '123456'; mysql> grant all on \*.\* to qianlong@'localhost' with grant option; mysql -uqianlong -p123456 -S /var/lib/mysql/mysql.sock

2

3

关机

#1.利用mysqladmin管理命令shutdown

mysqladmin shutdown -S /data/mysql3306/tmp/mysql.sock

#2.利用mysql客户端shutdown

mysql -uqianlong -p123456 -S /data/mysql3306/tmp/mysql.sock -e "shutdown"

#3.暴力关机(生产环境不推荐使用) kill -9 mysqlpid

1

2

3

4

5

6

7

8



知识点:mysqld\_safe和mysqld区别

mysqld:是mysql的核⼼程序，⽤于管理mysql的数据库⽂件以及⽤户的请求操作。mysqld可以读取配置⽂件,启动Mysql实例

mysqld\_safe:mysqld的守护进程(守护进程(Daemon)⼀般是为了保护我们的程序/服务的正常运⾏，当程序被关闭、异常退出等时再次启动程序/恢复服务),mysqld挂了会⾃动把 mysqld进程拉起来，相当于⻅识mysqld进程。

⽣产环境中使⽤mysqld\_safe居多

启动顺序 mysql.server—>mysqld\_safe—>mysqld



为了后期的⽅便管理，启动停⽌更⽅便，可以使⽤systemctl进⾏管理

1

#1.建立system管理文件

cat > /usr/lib/systemd/system/mysql3306.service <<EOF [Unit]

Description=MySQL Server After=network.target

[Install]

WantedBy=multi-user.target [Service]

Type=forking TimeoutSec=0 PermissionsStartOnly=true

ExecStart=/usr/local/mysql/bin/mysqld --defaults-file=/data/mysql3307/etc/ my.cnf --daemonize

LimitNOFILE = 65535

Restart=on-failure RestartSec=10 RestartPreventExitStatus=1 PrivateTmp=false

EOF

#2.添加可执行权限

chmod 755 /usr/lib/systemd/system/mysql3306.service

#3.重载 systemctl units systemctl daemon-reload

#4.设置为开机自启动

systemctl enable mysql3306.service

#5.启动与关闭Mysql实例

systemctl start mysql3306.service

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32



1.2.MySQL升降级inplace⽅式(课下实现)

inplace就地升级的⽅式有极⼤⻛险 mysql实例直接版本升级如果有业务容易发⽣未知问题，

⽣产⼀般不会采⽤，后期课程待主从复制学完 讲解企业级Mysql实例在线升级



2.MySQL⽇志简介



2.1.⼆进制⽇志(binary log)

1

#1.作用:

1. 记录DML/DDL语句:引起数据修改或者删除的，记录所有操作,包括发生时间，执行时长等；不记录SELECT
2. 主从复制
3. 恢复数据
4. 审计（audit）

#2.配置：

log\_bin = /xxx/mysql3306/var/mysql-bin #多实例

log\_bin\_index = mysql-bin.index

expire\_logs\_days = 14

max\_binlog\_size = 1G

binlog\_format = ROW

binlog\_cache\_size = 16M

sync\_binlog = 1

binlog\_rows\_query\_log\_events = 1

#3.查看：

mysql> show variables like '%log\_bin%'; #是否开启已有的binlog:

mysql> show binary logs; mysql> show master logs;

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

2.2.慢⽇志

1

慢日志 记录大于指定时间的SELECT或者未走index 的SQL show variables like '%slow%';

show variables like '%long%';

2

3



2.3.general log

1

作用:开启 general log 将所有到达MySQL Server的SQL语句记录下来功能:排查故障使用

show variables like 'general\_log'; -- 查看日志是否开启

set global general\_log=on; -- 开启日志功能

show variables like 'general\_log\_file'; -- 看看日志文件保存位置

set global general\_log\_file='/class/data/mysql3306/log/mysql.log'; -- 设置日

2

3

4

5

6

7

8

show variables like 'log\_output'; -- 看看日志输出类型 table或file

set global log\_output='table'; -- 设置输出类型为 table 写入mysql.slow\_log set global log\_output='file'; -- 设置输出类型为file

如何关闭:

SET GLOBAL general\_log = 'OFF';

RENAME TABLE mysql.general\_log TO mysql.general\_log2; DELETE FROM mysql.general\_log2;

9

10

11

12

13

14

2.4.错误⽇志

1

功能：记录启动-到关闭发生的 error,waring等信息

作用：定位数据库位置,死锁，异常，

配置：

log\_error\_verbosity

= 2

#1错误信息 2错误信息和告警信息； 3错误信息、告警信息和通知信息。 版本不同默认参数不

同,8.0没有此参数

log\_error

= /xxx/mysql3306/log/mysql.err #多实例

2

3

4

5

6

7

8



3.企业级MySQL⾃动化部署流程

⾯对⼏千台数据库服务器，⾯对每天⼏套到⼗⼏套MySQL集群的申请，如何实现快速交付集群的呢？



3.1.MySQL部署基础

⾸先看MySQL的单实例普通部署，⼤约如下⼏步下载软件

创建⽬录创建账号

编辑配置⽂件初始化

开启实例

创建库、账号等

那么⾃动化实现本地可以快速部署实例，搭建集群，需要如何实现呢？需要考虑如下：



【如何快速安装MySQL？】：

1. 依托公司的YUM源:

a. ⽣成⾃定义的RPM包，直接执⾏yum 安装，不能频繁变更RPM包。所以需要能⽀持⼀定灵活性的更改，就不能通过RPM实现。

1. MySQL部署⾥⾯参数变更

 单独做⼀个可以直接拉起的MySQL空实例，这样就既能⽀持快速部署，⼜能⽀持⼀定的灵活变更，调整参数我们就重新打个可直接开启的包就⾏了。





【MySQL依赖的环境】：

MySQL依赖:编写脚本,分配MySQL服务器⻆⾊调⽤执⾏,保证服务器的环境相同，执⾏⾼效

 安装包

 账号

 服务器参数配置等



【部署MySQL集群 与 周边环境】：

⽣产使⽤的MySQL⼏乎都是集群模式，即主从复制，1主2从这样，这就需要⼤家编写部署集群的脚本了，同时可能还要申请VIP，域名，备份，监控，⾼可⽤等，所以就需要按部就班的执⾏相关脚本了。



3.2.⾃动化部署流程



1.物理机混合部署的流程与架构



环境分类：

打包机器

平台管理机器 MySQL部署机器

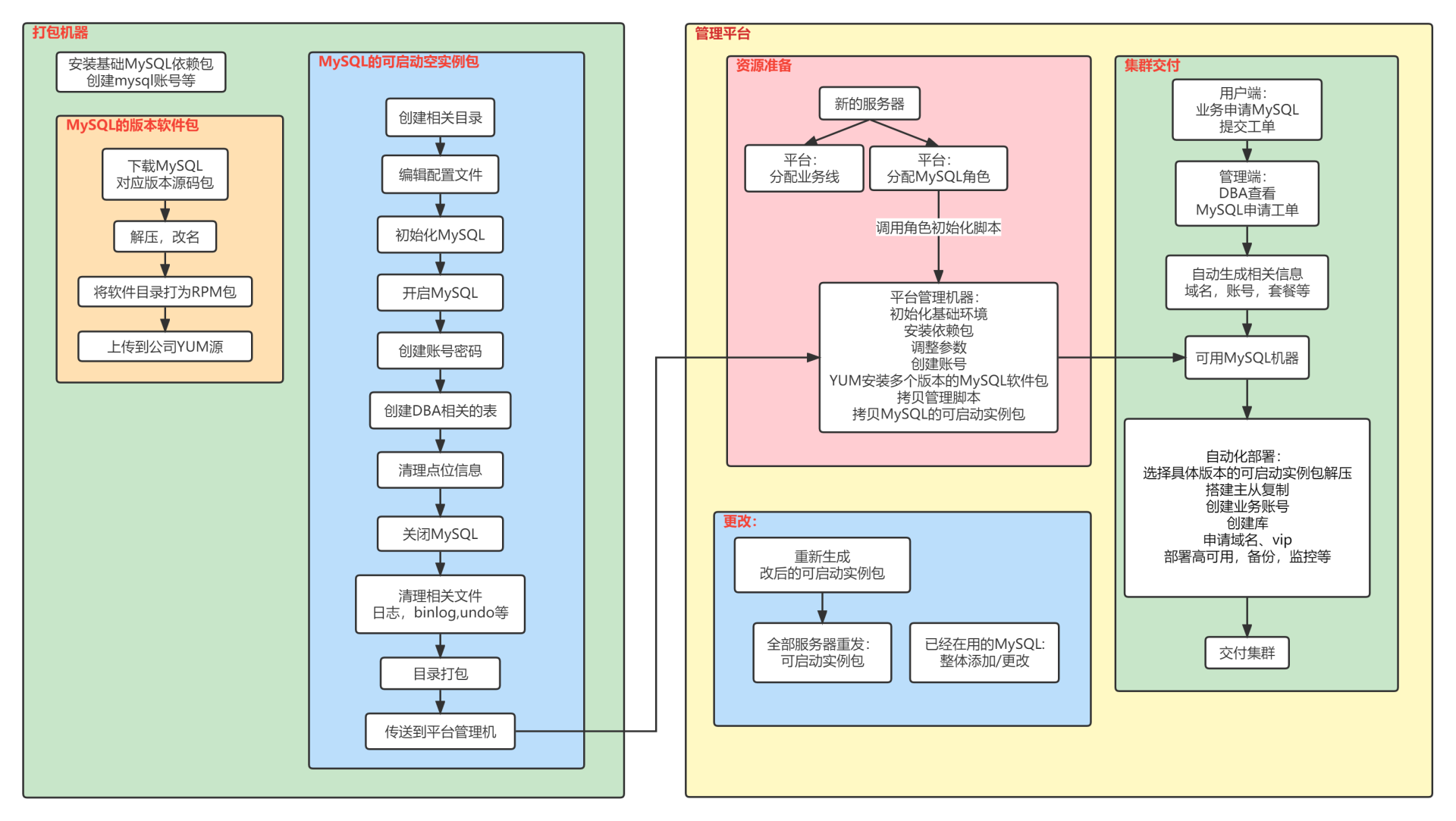


软件分类：

软件版本RPM包可启动的空实例包



【流程图如下】：





2.云化部署的流程与架构



环境分类：

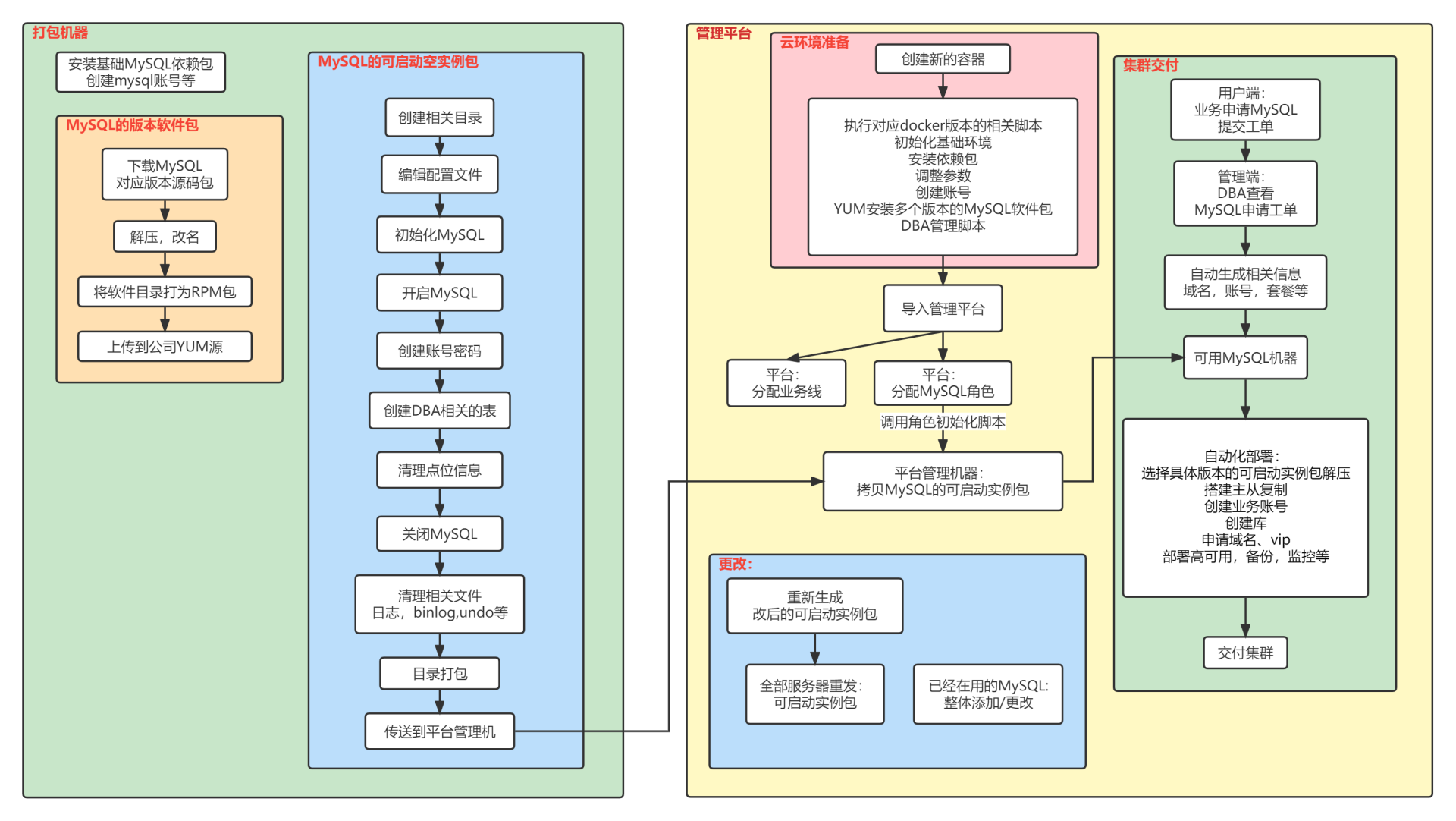
打包机器云环境

平台管理机器 MySQL部署机器



软件分类：

软件版本RPM包可启动的空实例包





3.3.具体打包流程

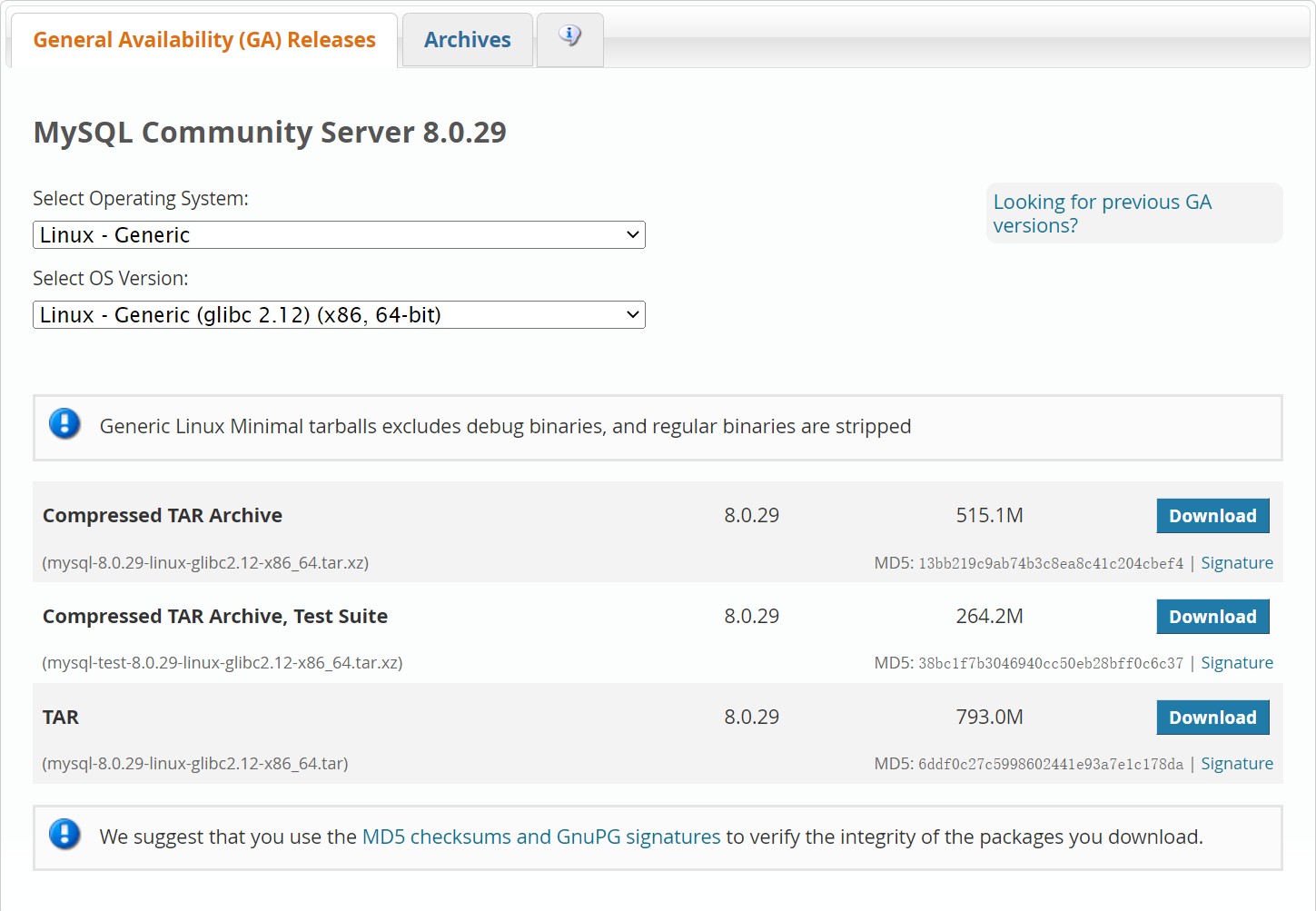


1.软件版本RPM包



【下载版本】：

选择想要的版本，进⾏下载



【打RPM包】：

1

#1.安装fpm

##安装ruby语言

yum -y install ruby-devel gcc make rpm-build rubygems

curl -sSL https://github.com/rvm/rvm/tarball/stable -o rvm-stable.tar.gz tar -xzvf rvm-stable.tar.gz

cd rvm-rvm-cc69ed9

./install --auto-dotfiles

source /usr/local/rvm/scripts/rvm rvm install 2.6.3

rvm use 2.6.3 --default

##安装fpm软件

gem sources --add <http://mirrors.aliyun.com/rubygems/> gem sources --remove https://rubygems.org/

gem sources -l

gem install fpm -v 1.4.0 fpm --help

#2.打包

cd /data/app

tar xf mysql-8.0.24-linux-glibc2.12-x86\_64.tar.xz

cp mysql-8.0.24-linux-glibc2.12-x86\_64 ./mysql8024 -rp

fpm -s dir -t rpm -n mysql8024 --epoch 1 -v 1.0.0 -C /data/app/mysql8024

--verbose --prefix /data/app/mysql8024

#3.上传到公司的YUM源

ls mysql8024-1.0.0-1.x86\_64.rpm ##上传到yum仓库后 直接安装即可

yum install mysql8024

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28



2.可启动的空实例包

ssh 打包机器：



【创建mysql⽤户】：

1

/usr/sbin/useradd mysql -s /sbin/nologin -g mysql



【安装依赖包】：

1

yum -y install make gcc-c++ cmake bison-devel ncurses-devel libaio libaio- devel net-tools

【创建⽬录】：

1

mkdir /data/mysql3306/{var,log,tmp,etc} chown mysql.mysql /data/mysql3306 -R

2



【修改配置⽂件】：

1

cd /data/mysql666/etc vim my.cnf

2

3

【初始化】：

1

mysqld --initialize-insecure --user=mysql --basedir=/usr/local/mysql --da tadir=/data/mysql3306/data

【启动MySQL】：

1

mysqld\_safe --defaults-file=/data/mysql3306/etc/my.cnf &



【账号密码】：

1

登录mysql 更改root密码

alter user 'qianlong'@'localhost' IDENTIFIED BY '123456';

酌情创建其他所需账号

2

3

4

【清理】：

1

reset master;

【关闭实例】：

1

mysqladmin shutdown -S /data/mysql3306/tmp/mysql.sock

[配置⽂件]:

1

port = 999999

server\_id = 888888

用于后面自动化的脚本进行替换，此处按照情况定义即可，与脚本对应就行

2

3

【删除⽂件】：

1

cd /data/mysql3306/data rm -fr binlog\*

rm -rf auto.cnf rm -f undo\_00\*

2

3

4

5

rm -f ib\_logfile\*

【清空⽇志】：

1

cd /data/mysql3306/log echo ""> mysql.err echo ""> mysql-slow.log

2

3

【打包】：

1

chown mysql.mysql /data/mysql3306 -R cd /data/mysql3306/

tar -czvf MySQL8024.tar.gz etc log tmp data

传到平台管理机器即可

2

3

4

【再次部署(⼿动)】:

1

mv MySQL8024.tar.gz /data/app/

4、⾃动化执⾏



4.1、服务器初始化脚本

对于⼀个新的服务器，⼤家可以酌情初始化相关环境。

分配MySQL⻆⾊后，对此机器进⾏相关环境初始化，例如：创建mysql账号

按照mysql的依赖包创建⽬录

部署DBA的管理脚本注册监控等等

拷⻉可启动的mysql初始化包,即上⾯的MySQL8029.tar.gz



4.2、部署MySQL

部署MySQL主要为如下部分



【安装软件包】：

yum -y install mysql8029

(对应上⾯打的RPM包，类似：mysql8029-1.0.0.rpm)

【配置实例】：

需要开发脚本，更改配置⽂件的端⼝、server\_id等开启实例



【搭建与交付集群】：

开发脚本，搭建主从复制集群交付集群