Q：启动MySQL时，systemctl start mysql，出现这样的报错

```
[root@pxc1 /]# journalctl -xe
1月 19 11:02:28 pxc1 dbus-daemon[854]: dbus[854]: [system] Activated service 'org.fedoraproject.Setroubleshootd' failed: Launch helper exit
1月 19 11:02:28 pxc1 dbus[854]: [system] Activating service name='org.fedoraproject.Setroubleshootd' (using servicehelper)
1月 19 11:02:28 pxc1 dbus-daemon[854]: dbus[854]: [system] Activating service name='org.fedoraproject.Setroubleshootd' (using servicehelper
1月 19 11:02:28 pxc1 dbus-daemon[854]: ERROR: policydb version 31 does not match my version range 15-30
1月 19 11:02:28 pxc1 dbus-daemon[854]: ERROR: Unable to open policy //etc/selinux/targeted/policy/policy.31.
1月 19 11:02:28 pxc1 python[9101]: detected unhandled Python exception in '/usr/sbin/setroubleshootd'
1月 19 11:02:28 pxc1 abrt-server[9106]: Not saving repeating crash in '/usr/sbin/setroubleshootd'
1月 19 11:02:28 pxc1 dbus-daemon[854]: Traceback (most recent call last):
1月 19 11:02:28 pxc1 dbus-daemon[854]: File "/usr/sbin/setroubleshootd", line 30, in <module>
1月 19 11:02:28 pxc1 dbus-daemon[854]: from setroubleshoot.util import log_debug
1月 19 11:02:28 pxc1 dbus-daemon[854]: File "/usr/lib64/python2.7/site-packages/setroubleshoot/util.py", line 304, in <module>
1月 19 11:02:28 pxc1 dbus-daemon[854]: from sepolicy import get_all_file_types
1月 19 11:02:28 pxc1 dbus-daemon[854]: File "/usr/lib64/python2.7/site-packages/sepolicy/__init__.py", line 907, in <module>
1月 19 11:02:28 pxc1 dbus-daemon[854]: raise e
1月 19 11:02:28 pxc1 dbus-daemon[854]: ValueError: Failed to read //etc/selinux/targeted/policy/policy.31 policy file
1月 19 11:02:28 pxc1 dbus[854]: [system] Activated service 'org.fedoraproject.Setroubleshootd' failed: Launch helper exited with unknown re
1月 19 11:02:28 pxc1 dbus-daemon[854]: dbus[854]: [system] Activated service 'org.fedoraproject.Setroubleshootd' failed: Launch helper exit
1月 19 11:02:28 pxc1 dbus[854]: [system] Activating service name='org.fedoraproject.Setroubleshootd' (using servicehelper)
1月 19 11:02:28 pxc1 dbus-daemon[854]: dbus[854]: [system] Activating service name='org.fedoraproject.Setroubleshootd' (using servicehelper
1月 19 11:02:28 pxc1 dbus-daemon[854]: ERROR: policydb version 31 does not match my version range 15-30
1月 19 11:02:28 pxc1 dbus-daemon[854]: ERROR: Unable to open policy //etc/selinux/targeted/policy/policy.31.
1月 19 11:02:28 pxc1 python[9108]: detected unhandled Python exception in '/usr/sbin/setroubleshootd'
1月 19 11:02:28 pxc1 abrt-server[9113]: Not saving repeating crash in '/usr/sbin/setroubleshootd'
1月 19 11:02:28 pxc1 dbus-daemon[854]: Traceback (most recent call last):
1月 19 11:02:28 pxc1 dbus-daemon[854]: File "/usr/sbin/setroubleshootd", line 30, in <module>
1月 19 11:02:28 pxc1 dbus-daemon[854]: from setroubleshoot.util import log_debug
1月 19 11:02:28 pxc1 dbus-daemon[854]: File "/usr/lib64/python2.7/site-packages/setroubleshoot/util.py", line 304, in <module>
1月 19 11:02:28 pxc1 dbus-daemon[854]: from sepolicy import get_all_file_types
1月 19 11:02:28 pxc1 dbus-daemon[854]: File "/usr/lib64/python2.7/site-packages/sepolicy/__init__.py", line 907, in <module>
1月 19 11:02:28 pxc1 dbus-daemon[854]: raise e
1月 19 11:02:28 pxc1 dbus-daemon[854]: ValueError: Failed to read //etc/selinux/targeted/policy/policy.31 policy file
1月 19 11:02:28 pxc1 dbus[854]: [system] Activated service 'org.fedoraproject.Setroubleshootd' failed: Launch helper exited with unknown re
```

A：yum provides semanage

yum -y install policycoreutils-python.x86_64

关闭selinux，重启

1、永久修改主机名

hostnamectl set-hostname xxx

重启

2、创建pxc集群的basedir路径

mkdir /opt/pxc

```
[root@pxc1 ~]# mkdir /opt/pxc
```

3、新建专门的linux用户和组

groupadd mysql

创建group mysql

vim /etc/group

会发现最后一行有mysql用户组

adduser -g mysql mysql

创建mysql用户，同时加入mysql用户组，自动创建mysql的homedir为/home/mysql

vim /etc/passwd 可以看到最后一行是mysql用户。

passwd mysql

为mysql用户新设立密码

将mysql用户加入sudo权限

chmod +w /etc/sudoers

vim /etc/sudoers

添加如下行：

```
#
# Adding HOME to env_keep may enable a user to run unrestricted
# commands via sudo.
#
# Defaults   env_keep += "HOME"

Defaults    secure_path = /sbin:/bin:/usr/sbin:/usr/bin

## Next comes the main part: which users can run what software on
## which machines (the sudoers file can be shared between multiple
## systems).
## Syntax:
##
##      user    MACHINE=COMMANDS
##
## The COMMANDS section may have other options added to it.
##
## Allow root to run any commands anywhere
root    ALL=(ALL)       ALL
mysql   ALL=(ALL)       ALL
## Allows members of the 'sys' group to run networking, software,
## service management apps and more.
# %sys ALL = NETWORKING, SOFTWARE, SERVICES, STORAGE, DELEGATING, PROCESSES, LOCATE, DRIV

## Allows people in group wheel to run all commands
%wheel  ALL=(ALL)       ALL

## Same thing without a password
# %wheel        ALL=(ALL)       NOPASSWD: ALL

## Allows members of the users group to mount and unmount the
## cdrom as root
```

然后再chmod -w /etc/sudoers

4、创建mysql服务的datadir logdir tmpdir

mkdir /opt/pxc/data

mkdir /opt/pxc/tmp

mkdir /opt/pxc/log

chown -R mysql:mysql /opt/pxc

```
[root@pxc1 ~]# chown -R  mysql:mysql  /opt/pxc
[root@pxc1 ~]# cd /opt/
[root@pxc1 opt]# ll
总用量 0
drwxr-xr-x. 5 mysql mysql 40 1月  19 09:30 pxc
drwxr-xr-x. 2 root  root   6 3月  26 2015 rh
[root@pxc1 opt]#
```

5、卸载对pxc安装有影响的软件或库

检查mysql是否存在

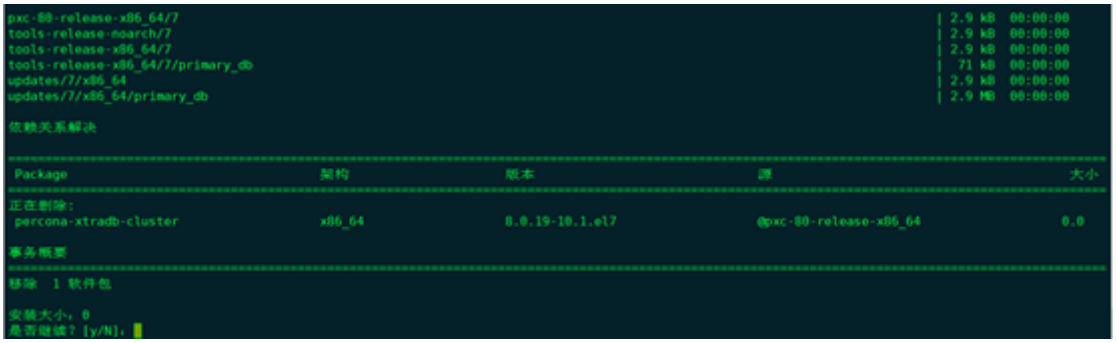
which mysql

```
[root@pxc1 opt]# which mysql
/usr/bin/which: no mysql in (/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/root/bin)
[root@pxc1 opt]#
```

如果存在

如果之前是通过yum install percona-xtradb-cluster 安装的，卸载方法是：

yum remove percona-xtradb-cluster



移除完了percona-xtradb-cluster后which mysql

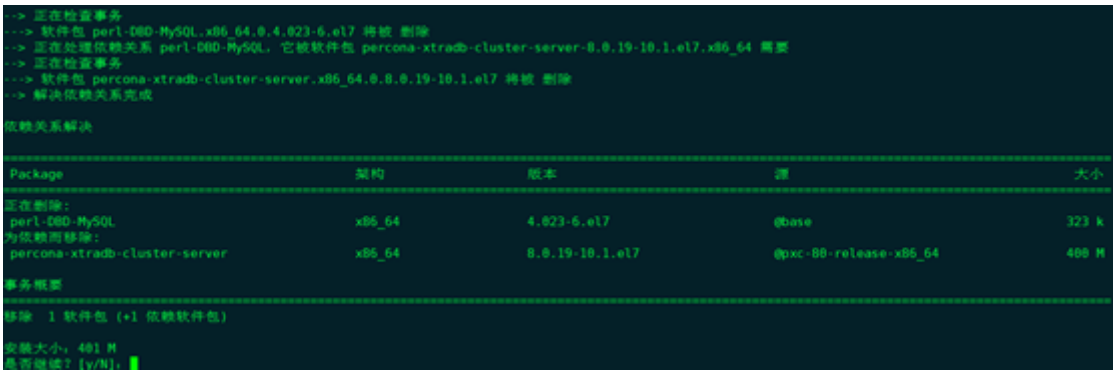
```
[root@cent7c opt]# which mysql
/usr/bin/mysql
```

发现mysql命令还在，继续处理，通过rpm -qa 查询所有带mysql字样的rpm安装包如下：

```
[root@cent7c opt]# rpm -qa | grep -i mysql
perl-DBD-MySQL-4.023-6.el7.x86_64
[root@cent7c opt]# yum remove perl-DBD-MySQL-4.023-6.el7.x86_64
```

然后挨个执行rpm -e --nodeps 或者执行yum remove xxxx移除如下：

```
[root@cent7c opt]# rpm -qa | grep -i mysql | xargs rpm -e --nodeps
```



最后再次which mysql 发现没有了，全部卸载成功。

有些 CentOS 版本默认捆绑了mariadb-libs，在安装PXC之前需要先将其卸载

yum -y remove mari*

6、防火墙、selinux和端口准备

开启需要的端口

systemctl start firewalld

firewall-cmd --zone=public --add-port=3306/tcp --permanent

firewall-cmd --zone=public --add-port=4444/tcp --permanent

firewall-cmd --zone=public --add-port=4567/tcp --permanent

firewall-cmd --zone=public --add-port=4568/tcp --permanent

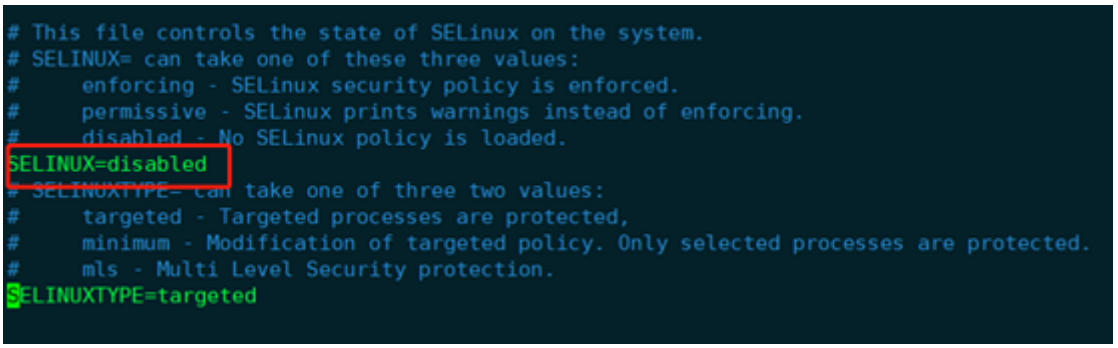
firewall-cmd --reload

查看所有开启的访问端口：firewall-cmd --list-ports

7、 pxc集群部署涉及需要的selinux操作

永久生效需要修改文件/etc/selinux/config

vim /etc/selinux/config



修改文件中SELINUX=disabled,重启即可。

至此，pxc集群安装所需的全部准备工作全部完成。下面开始安装pxc集群。

8、以root用户创建/root/tmp

解压项目交付清单中的pxc8.0_on_centos7_tarball.tar.gz 到/root/tmp目录下：

```
[root@pxc1 tmp]# tar -xvf pxc8.0_on_centos7_tarball.tar.gz
pxc8.0_on_centos7_tarball/
pxc8.0_on_centos7_tarball/percona-xtradb-cluster-8.0.19-10.1.el7.x86_64.rpm
pxc8.0_on_centos7_tarball/percona-xtradb-cluster-client-8.0.19-10.1.el7.x86_64.rpm
pxc8.0_on_centos7_tarball/percona-xtradb-cluster-debuginfo-8.0.19-10.1.el7.x86_64.rpm
pxc8.0_on_centos7_tarball/percona-xtradb-cluster-devel-8.0.19-10.1.el7.x86_64.rpm
pxc8.0_on_centos7_tarball/percona-xtradb-cluster-full-8.0.19-10.1.el7.x86_64.rpm
pxc8.0_on_centos7_tarball/percona-xtradb-cluster-garbd-8.0.19-10.1.el7.x86_64.rpm
pxc8.0_on_centos7_tarball/percona-xtradb-cluster-server-8.0.19-10.1.el7.x86_64.rpm
pxc8.0_on_centos7_tarball/percona-xtradb-cluster-shared-8.0.19-10.1.el7.x86_64.rpm
pxc8.0_on_centos7_tarball/percona-xtradb-cluster-shared-compat-8.0.19-10.1.el7.x86_64.rpm
pxc8.0_on_centos7_tarball/percona-xtradb-cluster-test-8.0.19-10.1.el7.x86_64.rpm
pxc8.0_on_centos7_tarball/qpress-11-1.el7.x86_64.rpm
pxc8.0_on_centos7_tarball/percona-xtrabackup-80-8.0.13-1.el7.x86_64.rpm
pxc8.0_on_centos7_tarball/percona-xtrabackup-80-debuginfo-8.0.13-1.el7.x86_64.rpm
pxc8.0_on_centos7_tarball/percona-xtrabackup-test-80-8.0.13-1.el7.x86_64.rpm
pxc8.0_on_centos7_tarball/proxysql2-2.0.12-1.1.el7.x86_64.rpm
[root@pxc1 tmp]#
```

cd pxc8.0_on_centos7_tarball/

```
[root@pxc1 tmp]# cd pxc8.0_on_centos7_tarball/
```

yum localinstall *.rpm

完成持续约2-3分钟即可完成全部安装。

which mysql 发现/usr/bin/mysql ,mysql --version 可见这个mysql是pxc版本的mysql。

```
完毕!
[root@pxc1 pxc8.0_on_centos7_tarball]# mysql --version
mysql  Ver 8.0.19-10 for Linux on x86_64 (Percona XtraDB Cluster (GPL), Release rel10, Revision 727f180, WSREP version 26.4.3)
[root@pxc1 pxc8.0_on_centos7_tarball]#
```

9、初次启动mysql服务并修改管理员用户(root)密码

修改配置文件

Centos 下Pxc集群中的mysql默认配置文件是/etc/my.cnf。

vim /etc/my.cnf

默认的配置文件如下：

```
# Template my.cnf for PXC
# Edit to your requirements.
[client]
socket=/var/lib/mysql/mysql.sock

[mysqld]
server-id=1
datadir=/var/lib/mysql
socket=/var/lib/mysql/mysql.sock
log-error=/var/log/mysqld.log
pid-file=/var/run/mysqld/mysqld.pid

# Binary log expiration period is 604800 seconds, which equals 7 days
binlog_expire_logs_seconds=604800

######## wsrep ###############
# Path to Galera library
wsrep_provider=/usr/lib64/galera4/libgalera_smm.so

# Cluster connection URL contains IPs of nodes
#If no IP is found, this implies that a new cluster needs to be created,
#in order to do that you need to bootstrap this node
wsrep_cluster_address=gcomm://

# In order for Galera to work correctly binlog format should be ROW
binlog_format=ROW

# Slave thread to use
wsrep_slave_threads=8

wsrep_log_conflicts

# This changes how InnoDB autoincrement locks are managed and is a requirement for Galera
"/etc/my.cnf" 48L, 1280C
```

将其修改为：

```
[client]
socket=/opt/pxc/tmp/mysql.sock

[mysqld]
server-id=1
port=8888
lower_case_table_names=1
basedir=/opt/pxc
datadir=/opt/pxc/data
socket=/opt/pxc/tmp/mysql.sock
mysqlx_socket=/opt/pxc/tmp/mysqlx.sock
log-error=/opt/pxc/log/mysqld.log
pid-file=/opt/pxc/tmp/mysqld.pid
default_authentication_plugin=mysql_native_password
```

```
# Template my.cnf for PXC
# Edit to your requirements.
[client]
socket/opt/pxc/tmp/mysql.sock

[mysqld]
server-id=1
port=8888
lower_case_table_names=1
basedir=/opt/pxc
datadir=/opt/pxc/data
socket=/opt/pxc/tmp/mysql.sock
mysqlx_socket=/opt/pxc/tmp/mysqlx.sock
log-error=/opt/pxc/log/mysqld.log
pid-file=/opt/pxc/tmp/mysqld.pid
default_authentication_plugin=mysql_native_password

# Binary log expiration period is 604800 seconds, which equals 7 days
binlog_expire_logs_seconds=604800

######## wsrep ###############
# Path to Galera library
wsrep_provider=/usr/lib64/galera4/libgalera_smm.so

# Cluster connection URL contains IPs of nodes
#If no IP is found, this implies that a new cluster needs to be created,
#in order to do that you need to bootstrap this node
wsrep_cluster_address=gcomm://

# In order for Galera to work correctly binlog format should be ROW
binlog_format=ROW

# Slave thread to use
-- 插入 --
```

上述配置将pxc对外提供服务的默认端口(是3306)改为8888。并创建/opt/pxc下data tmp log3个目录分别作为mysql服务的数据存储目录，tmp目录和log目录。依次将所有目录改为mysql用户和mysql组。

mkdir -p /opt/pxc/data

mkdir -p /opt/pxc/tmp

mkdir -p /opt/pxc/log

同时把8888端口加入防火墙开放端口。

firewall-cmd --zone=public --add-port=8888/tcp --permanent

重新加载防火墙规则

firewall-cmd --reload

查看所有开启的访问端口：firewall-cmd --list-ports

```
[root@pxc1 pxc8.0_on_centos7_tarball]# firewall-cmd --zone=public --add-port=8888/tcp --permanent
success
[root@pxc1 pxc8.0_on_centos7_tarball]# firewall-cmd --reload
success
[root@pxc1 pxc8.0_on_centos7_tarball]# firewall-cmd --list-ports
3306/tcp 8888/tcp 4444/tcp 4568/tcp 4567/tcp
[root@pxc1 pxc8.0_on_centos7_tarball]#
```

接下来启动每个节点上的mysql服务。注意必须切换到mysql用户以sudo systemctl start mysql 命令启动。（当然sudo service mysql start命令启动也可以。）

(sudo systemctl status mysql 查询mysql服务状态，sudo systemctl stop mysql 停止mysql服务)

su mysql

sudo systemctl start mysql

```
[root@pxc1 ~]# su mysql
[mysql@pxc1 root]$ sudo systemctl start mysql
[sudo] password for mysql:
[mysql@pxc1 root]$ sudo systemctl status mysql
● mysql.service - Percona XtraDB Cluster
   Loaded: loaded (/usr/lib/systemd/system/mysql.service; enabled; vendor preset: disabled)
   Active: active (running) since 二 2021-01-19 11:22:35 CST; 11s ago
  Process: 3489 ExecStopPost=/usr/bin/mysql-systemd stop-post (code=exited, status=0/SUCCESS)
  Process: 3456 ExecStop=/usr/bin/mysql-systemd stop (code=exited, status=0/SUCCESS)
  Process: 3726 ExecStartPost=/usr/bin/mysql-systemd start-post $MAINPID (code=exited, status=0/SUCCESS)
  Process: 3724 ExecStartPost=/bin/sh -c systemctl unset-environment _WSREP_START_POSITION (code=exited, status=0/SUCCESS)
  Process: 3618 ExecStartPre=/bin/sh -c VAR=`bash /usr/bin/mysql-systemd galera-recovery`; [ $? -eq 0 ] && systemctl set-environment _WSREP
_START_POSITION=$VAR || exit 1 (code=exited, status=0/SUCCESS)
  Process: 3614 ExecStartPre=/bin/sh -c systemctl unset-environment _WSREP_START_POSITION (code=exited, status=0/SUCCESS)
  Process: 3569 ExecStartPre=/usr/bin/mysql-systemd start-pre (code=exited, status=0/SUCCESS)
 Main PID: 3668 (mysqld)
   Status: "Server is operational"
   CGroup: /system.slice/mysql.service
           └─3668 /usr/sbin/mysqld --wsrep_start_position=911b6f7b-5a03-11eb-abed-3a4a225693b0:1

1月 19 11:22:34 pxc1 systemd[1]: Starting Percona XtraDB Cluster...
1月 19 11:22:35 pxc1 mysql-systemd[3726]: SUCCESS!
1月 19 11:22:35 pxc1 systemd[1]: Started Percona XtraDB Cluster.
[mysql@pxc1 root]$
```

netstat -nap | grep -i mysql

```
[mysql@pxc1 root]$ netstat -nap | grep -i mysql
(Not all processes could be identified, non-owned process info
 will not be shown, you would have to be root to see it all.)
tcp        0      0 0.0.0.0:4567            0.0.0.0:*               LISTEN      3668/mysqld
tcp6       0      0 :::8888                 :::*                    LISTEN      3668/mysqld
tcp6       0      0 :::33060                :::*                    LISTEN      3668/mysqld
unix  2      [ ACC ]     STREAM     LISTENING     39964    3668/mysqld          /opt/pxc/tmp/mysql.sock
unix  2      [ ACC ]     STREAM     LISTENING     35240    3668/mysqld          /opt/pxc/tmp/mysqlx.sock
unix  2      [ ]         DGRAM                    18032    3668/mysqld
[mysql@pxc1 root]$
```

可见8888端口在对外提供mysql查询服务。

vim /opt/pxc/log/mysqld.log 可以看到mysql服务启动日志如下：

```
2021-01-19T03:22:35.248867Z 0 [Warning] [MY-013245] [Server] The SSL library function CRYPTO_set_mem_functions failed. This is typically ca
used by the SSL library already being used. As a result the SSL memory allocation will not be instrumented.
2021-01-19T03:22:35.476323Z 3 [Note] [MY-000000] [WSREP] wsrep_init_schema_and_SR (nil)
2021-01-19T03:22:35.481686Z 3 [System] [MY-000000] [WSREP] PXC upgrade completed successfully
2021-01-19T03:22:35.581712Z 0 [Warning] [MY-010068] [Server] CA certificate ca.pem is self signed.
2021-01-19T03:22:35.590232Z 0 [Warning] [MY-010324] [Server] 'db' entry 'percona_schema mysql.pxc.sst.role@localhost' had database in mixed
 case that has been forced to lowercase because lower_case_table_names is set. It will not be possible to remove this privilege using REVOK
E.
2021-01-19T03:22:35.597025Z 0 [Note] [MY-000000] [WSREP] Initialized wsrep sidno 2
2021-01-19T03:22:35.597041Z 0 [Note] [MY-000000] [Galera] Server initialized
2021-01-19T03:22:35.597047Z 0 [Note] [MY-000000] [WSREP] Server status change initializing -> initialized
2021-01-19T03:22:35.597054Z 0 [Note] [MY-000000] [WSREP] wsrep_notify_cmd is not defined, skipping notification.
2021-01-19T03:22:35.597135Z 2 [Note] [MY-000000] [Galera] Bootstrapping a new cluster, setting initial position to 00000000-0000-0000-0000-
000000000000:-1
2021-01-19T03:22:35.602150Z 9 [Note] [MY-000000] [WSREP] Starting applier thread 9
2021-01-19T03:22:35.602603Z 11 [Note] [MY-000000] [WSREP] Starting applier thread 11
2021-01-19T03:22:35.602829Z 13 [Note] [MY-000000] [WSREP] Starting applier thread 13
2021-01-19T03:22:35.602828Z 10 [Note] [MY-000000] [WSREP] Starting applier thread 10
2021-01-19T03:22:35.603061Z 0 [System] [MY-010931] [Server] /usr/sbin/mysqld: ready for connections. Version: '8.0.19-10'  socket: '/opt/px
c/tmp/mysql.sock'  port: 8888  Percona XtraDB Cluster (GPL), Release rel10, Revision 727f180, WSREP version 26.4.3.
2021-01-19T03:22:35.603103Z 14 [Note] [MY-000000] [WSREP] Starting applier thread 14
2021-01-19T03:22:35.603169Z 15 [Note] [MY-000000] [WSREP] Starting applier thread 15
2021-01-19T03:22:35.603282Z 16 [Note] [MY-000000] [WSREP] Starting applier thread 16
2021-01-19T03:22:35.604186Z 8 [Note] [MY-000000] [WSREP] Recovered cluster id 911b6f7b-5a03-11eb-abed-3a4a225693b0
2021-01-19T03:22:35.604711Z 2 [Note] [MY-000000] [WSREP] Server status change initialized -> joined
2021-01-19T03:22:35.604718Z 2 [Note] [MY-000000] [WSREP] wsrep_notify_cmd is not defined, skipping notification.
2021-01-19T03:22:35.604729Z 2 [Note] [MY-000000] [WSREP] wsrep_notify_cmd is not defined, skipping notification.
2021-01-19T03:22:35.608498Z 2 [Note] [MY-000000] [Galera] Server pxc-cluster-node-1 synced with group
2021-01-19T03:22:35.608513Z 2 [Note] [MY-000000] [WSREP] Server status change joined -> synced
2021-01-19T03:22:35.608517Z 2 [Note] [MY-000000] [WSREP] Synchronized with group, ready for connections
2021-01-19T03:22:35.608520Z 2 [Note] [MY-000000] [WSREP] wsrep_notify_cmd is not defined, skipping notification.
2021-01-19T03:22:35.740872Z 0 [System] [MY-011323] [Server] X Plugin ready for connections. Socket: '/opt/pxc/tmp/mysqlx.sock' bind-address
: '::' port: 33060
```

接下来很重要的一个工作是首次登陆，输入密码，修改root密码，并且创建root@%账户使得root可以从任意host连接mysql服务。

执行 grep password /opt/pxc/log/mysqld.log 从日志中查询mysql首次临时密码

```
[mysql@pxc1 root]$ grep password /opt/pxc/log/mysqld.log
2021-01-19T03:00:09.438921Z 5 [Note] [MY-010454] [Server] A temporary password is generated for root@localhost: Cza7o.e!Hjlj
```

接着执行：mysql -u root -p 以上面临时密码连接mysql

然后修改root@localhost密码：

ALTER USER 'root'@'localhost' IDENTIFIED BY '新密码';

FLUSH PRIVILEGES;

```
mysql> ALTER USER 'root'@'localhost' IDENTIFIED BY 'c
Query OK, 0 rows affected (0.01 sec)

mysql> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0.02 sec)
```

修改了root@localhost的密码后 exit退出后重新以新密码登陆试试。

此外很重要的一个操作是：创建root@%用户使得root用户可以从任意host连接mysql服务。

CREATE USER 'root'@'%' IDENTIFIED BY '自己密码' ;

GRANT ALL ON *.* TO 'root'@'%';

FLUSH PRIVILEGES;

```
mysql> CREATE USER 'root'@'%' IDENTIFIED BY 'c
Query OK, 0 rows affected (0.01 sec)

mysql> GRANT ALL ON *.* TO 'root'@'%';
Query OK, 0 rows affected (0.03 sec)

mysql> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0.02 sec)
```

另外两个服务器上重复上述设置操作完成即可。

至此，pxc集群命令工具安装完成。此时每一个mysql服务都是一个单独孤立的实例，并不形成集群。

10、部署pxc集群

部署pxc集群基于前节3个节点上mysql 命令都可以正常启动连接了。接下来如下操作：

停止所有节点上的mysql服务

在三个节点上依次执行命令：sudo systemctl stop mysql

修改第一个节点上/etc/my.cf

sudo vim /etc/my.cnf

```
# Template my.cnf for PXC
# Edit to your requirements.
[client]
socket=/opt/pxc/tmp/mysql.sock

[mysqld]
server-id=1
port=8888
lower_case_table_names=1
basedir=/opt/pxc
datadir=/opt/pxc/data
socket=/opt/pxc/tmp/mysql.sock
mysqlx_socket=/opt/pxc/tmp/mysqlx.sock
log-error=/opt/pxc/log/mysqld.log
pid-file=/opt/pxc/tmp/mysqld.pid
default_authentication_plugin=mysql_native_password

#pxc-encrypt-cluster-traffic=ON
# Binary log expiration period is 604800 seconds, which equals 7 days
binlog_expire_logs_seconds=604800

######## wsrep ###############
# Path to Galera library
wsrep_provider=/usr/lib64/galera4/libgalera_smm.so

# Cluster connection URL contains IPs of nodes
#If no IP is found, this implies that a new cluster needs to be created,
#in order to do that you need to bootstrap this node
wsrep_cluster_address=gcomm://192.168.0.103,192.168.0.104,192.168.0.106

# In order for Galera to work correctly binlog format should be ROW
binlog_format=ROW
```

```
######## wsrep ###############
# Path to Galera library
wsrep_provider=/usr/lib64/galera4/libgalera_smm.so

# Cluster connection URL contains IPs of nodes
#If no IP is found, this implies that a new cluster needs to be created,
#in order to do that you need to bootstrap this node
wsrep_cluster_address=gcomm://192.168.0.103,192.168.0.104,192.168.0.106

# In order for Galera to work correctly binlog format should be ROW
binlog_format=ROW

# Slave thread to use
wsrep_slave_threads=8

wsrep_log_conflicts

# This changes how InnoDB autoincrement locks are managed and is a requirement for Galera
innodb_autoinc_lock_mode=2

# Node IP address
wsrep_node_address=192.168.0.106
# Cluster name
wsrep_cluster_name=pxc-cluster

#If wsrep_node_name is not specified,  then system hostname will be used
wsrep_node_name=pxc-cluster-node-1

#pxc_strict_mode allowed values: DISABLED,PERMISSIVE,ENFORCING,MASTER
pxc_strict_mode=ENFORCING

# SST method
wsrep_sst_method=xtrabackup-v2
```

重要的修改如上2图所示，简单说明如下：

server-id是pxc实例在pxc集群中的唯一id,必须是数字，不能重复，不同pxc集群节点依次设置1,2,3....

wsrep_provider维持固定设置就可以。

wsrep_cluster_address是pxc集群地址，至少为集群所有节点的1个值。但强烈建议把所有集群节点ip都写上。

wsrep_node_address当前pxc节点的ip地址。

wsrep_cluster_name是pxc集群名，所有节点的这个值必须一致。

wsrep_node_name当前pxc节点的名称，不同节点必须不一致。

重点要说明的是：pxc-encrypt-cluster-traffic的值有ON和OFF两个可能的取值。其默认值是ON，即如果文件中不配置，相当于pxc-encrypt-cluster-traffic=ON。目前文件中改配置项是注释状态，即表明pxc-encrypt-cluster-traffic=ON。 该配置项的含义是 pxc集群所有节点之间通讯的加密与否，OFF表示不加密，ON表示加密，使用的是TSL验证。官方也是强烈推荐用ON以TLS加密认证通讯。Mysq程序第一次启动时会在datadir即：/opt/pxc/data目录下自动生成相关的TLS认证相关文件，文件均已.pem后缀结尾。

设置了pxc-encrypt-cluster-traffic=ON后，PXC集群节点间会通过这些认证文件进行相互间数据同步的通讯。唯一需要注意的是，所有节点的这些认证文件必须一致。这非

常重要。

按照以上同样的方法设置其他2个节点的/etc/my.cf文件。

同步TLS认证文件

在第一个节点上添加其他节点IP

sudo vim /etc/hosts

```
127.0.0.1   localhost localhost.localdomain localhost4 localhost4.localdomain4
::1         localhost localhost.localdomain localhost6 localhost6.localdomain6
192.168.0.103   pxc2
192.168.0.104   pxc3
~
~
```

拷贝第一个节点上datadir即/opt/pxc/data目录下的所有*.pem文件到所有其他节点的datadir

即/opt/pxc/data目录下。

rsync -avP /opt/pxc/data/*.pem mysql@pxc2:/opt/pxc/data/

```
[mysql@pxc1 root]$ rsync -avP /opt/pxc/data/*.pem mysql@pxc2:/opt/pxc/data/
The authenticity of host 'pxc2 (192.168.0.103)' can't be established.
ECDSA key fingerprint is f9:c8:e9:8a:06:25:43:bd:66:f0:92:c9:cd:86:61:1e.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'pxc2,192.168.0.103' (ECDSA) to the list of known hosts.
mysql@pxc2's password:
sending incremental file list
ca-key.pem
        1680 100%    0.00kB/s    0:00:00 (xfer#1, to-check=7/8)
ca.pem
        1120 100%    1.07MB/s    0:00:00 (xfer#2, to-check=6/8)
client-cert.pem
        1120 100%  546.88kB/s    0:00:00 (xfer#3, to-check=5/8)
client-key.pem
        1680 100%  410.16kB/s    0:00:00 (xfer#4, to-check=4/8)
private_key.pem
        1676 100%  409.18kB/s    0:00:00 (xfer#5, to-check=3/8)
public_key.pem
         452 100%  110.35kB/s    0:00:00 (xfer#6, to-check=2/8)
server-cert.pem
        1120 100%  273.44kB/s    0:00:00 (xfer#7, to-check=1/8)
server-key.pem
        1676 100%  409.18kB/s    0:00:00 (xfer#8, to-check=0/8)

sent 11049 bytes  received 278 bytes  462.33 bytes/sec
total size is 10524  speedup is 0.93
```

rsync -avP /opt/pxc/data/*.pem mysql@pxc3:/opt/pxc/data/

```
[mysql@pxc1 root]$ rsync -avP /opt/pxc/data/*.pem mysql@pxc3:/opt/pxc/data/
The authenticity of host 'pxc3 (192.168.0.104)' can't be established.
ECDSA key fingerprint is f9:c8:e9:8a:06:25:43:bd:66:f0:92:c9:cd:86:61:1e.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'pxc3,192.168.0.104' (ECDSA) to the list of known hosts.
mysql@pxc3's password:
sending incremental file list
ca-key.pem
        1680 100%    0.00kB/s    0:00:00 (xfer#1, to-check=7/8)
ca.pem
        1120 100%    1.07MB/s    0:00:00 (xfer#2, to-check=6/8)
client-cert.pem
        1120 100%    1.07MB/s    0:00:00 (xfer#3, to-check=5/8)
client-key.pem
        1680 100%    1.60MB/s    0:00:00 (xfer#4, to-check=4/8)
private_key.pem
        1676 100%    1.60MB/s    0:00:00 (xfer#5, to-check=3/8)
public_key.pem
         452 100%  441.41kB/s    0:00:00 (xfer#6, to-check=2/8)
server-cert.pem
        1120 100%    1.07MB/s    0:00:00 (xfer#7, to-check=1/8)
server-key.pem
        1676 100%    1.60MB/s    0:00:00 (xfer#8, to-check=0/8)

sent 11049 bytes  received 278 bytes  1742.62 bytes/sec
total size is 10524  speedup is 0.93
[mysql@pxc1 root]$
```

引导启动pxc集群第一个节点

此处是pxc1为集群第一个节点。在pxc1上执行：

sudo systemctl start mysql@bootstrap.service

netstat -nap | grep -i mysql

```
[mysql@pxc1 root]$ netstat -nap | grep -i mysql
(Not all processes could be identified, non-owned process info
 will not be shown, you would have to be root to see it all.)
tcp        0      0 0.0.0.0:4567            0.0.0.0:*               LISTEN      6369/mysqld
tcp6       0      0 :::8888                 :::*                    LISTEN      6369/mysqld
tcp6       0      0 :::33060                :::*                    LISTEN      6369/mysqld
unix  2      [ ACC ]     STREAM     LISTENING     59602    6369/mysqld          /opt/pxc/tmp/mysql.sock
unix  2      [ ACC ]     STREAM     LISTENING     64837    6369/mysqld          /opt/pxc/tmp/mysqlx.sock
unix  2      [ ]         DGRAM                    63636    6369/mysqld
[mysql@pxc1 root]$
```

可以看到很多mysqld服务被启动，众多端口在监听。

接着mysql登录该服务，执行show status LIKE 'wsrep%';

```
| wsrep_cluster_capabilities       |
| wsrep_cluster_conf_id            | 1
| wsrep_cluster_size               | 1
| wsrep_cluster_state_uuid         | 911b6f7b-5a03-11eb-abed-3a4a225693b0
| wsrep_cluster_status             | Primary
| wsrep_connected                  | ON
| wsrep_local_bf_aborts            | 0
| wsrep_local_index                | 0
| wsrep_provider_capabilities      | :MULTI_MASTER:CERTIFICATION:PARALLEL_APPLYING:TRX_REPLAY:ISOLATION:PAUSE:CAUSAL_READS:INCREMENTAL_WRIT
ESET:UNORDERED:PREORDERED:STREAMING:NBO: |
| wsrep_provider_name              | Galera
| wsrep_provider_vendor            | Codership Oy <info@codership.com>
| wsrep_provider_version           | 4.3(r752664d)
| wsrep_ready                      | ON
| wsrep_thread_count               | 9
+----------------------------------+-------------------------------------------------------------------------------------------------------------------
-----------------------------------------+
75 rows in set (1.22 sec)

mysql>
```

其中wsrep_cluster_size 为1，表明现在pxc集群中只有1个节点。

wsrep_local_state_comment 为Synced表明pxc集群节点已经完成通过。后续每往集群中增加一个节点，都强烈建议在上一个节点这个值为Synced之后再操作，1个接1个节点加入，不要一次性加入所有节点。

依次加入其它节点到pxc集群

先在pxc2上执行：sudo systemctl start mysql

（注意：只有第一个节点需要引导启动，命令是：sudo systemctl start mysql@bootstrap.service，后续加入其它节点的命令都是：sudo systemctl start mysql）

注意以这种方式依次将其它节点都加入第一个节点所在的pxc集群，后续所有节点的数据都会被清空，然后将节点1引导节点的数据同步过来。

netstat -nap | grep -i mysql

```
[mysql@pxc2 /]$ sudo systemctl start mysql
[sudo] password for mysql:
[mysql@pxc2 /]$ netstat -nap | grep -i mysql
(Not all processes could be identified, non-owned process info
 will not be shown, you would have to be root to see it all.)
tcp        0      0 0.0.0.0:4567            0.0.0.0:*               LISTEN      5936/mysqld
tcp        0      0 192.168.0.103:48104     192.168.0.106:4567      ESTABLISHED 5936/mysqld
tcp6       0      0 :::8888                 :::*                    LISTEN      5936/mysqld
tcp6       0      0 :::33060                :::*                    LISTEN      5936/mysqld
unix  2      [ ACC ]     STREAM     LISTENING     57553    5936/mysqld          /opt/pxc/tmp/mysql.sock
unix  2      [ ACC ]     STREAM     LISTENING     57554    5936/mysqld          /opt/pxc/tmp/mysqlx.sock
unix  2      [ ]         DGRAM                    40495    5936/mysqld
[mysql@pxc2 /]$
```

再登录第二个节点的mysql服务，mysql用户名和密码直接被同步成了引导节点的用户名和密码。继续执行：show status LIKE 'wsrep%';

```
| wsrep_cluster_capabilities      |
| wsrep_cluster_conf_id           | 2
| wsrep_cluster_size              | 2
| wsrep_cluster_state_uuid        | 911b6f7b-5a03-11eb-abed-3a4a225693b0
| wsrep_cluster_status            | Primary
| wsrep_connected                 | ON
| wsrep_local_bf_aborts           | 0
| wsrep_local_index               | 0
| wsrep_provider_capabilities     | :MULTI_MASTER:CERTIFICATION:PARALLEL_APPLYING:TRX_REPLAY:ISOLATION:PAUSE:CAUSAL_READS:INCREMENTAL_WRIT
ESET:UNORDERED:PREORDERED:STREAMING:NBO: |
| wsrep_provider_name             | Galera
| wsrep_provider_vendor           | Codership Oy <info@codership.com>
| wsrep_provider_version          | 4.3(r752664d)
| wsrep_ready                     | ON
| wsrep_thread_count              | 9
+---------------------------------+-----------------------------------------------------------------------------------------------------
----------------------------------------+
75 rows in set (0.07 sec)

mysql>
```

```
| wsrep_cert_deps_distance        | 0
| wsrep_apply_oooe                | 0
| wsrep_apply_oool                | 0
| wsrep_apply_window              | 0
| wsrep_commit_oooe               | 0
| wsrep_commit_oool               | 0
| wsrep_commit_window             | 0
| wsrep_local_state               | 4
| wsrep_local_state_comment       | Synced
| wsrep_cert_index_size           | 0
| wsrep_cert_bucket_count         | 1
| wsrep_gcache_pool_size          | 4112
| wsrep_causal_reads              | 0
| wsrep_cert_interval             | 0
| wsrep_open_transactions         | 0
| wsrep_open_connections          | 0
| wsrep_ist_receive_status        |
```

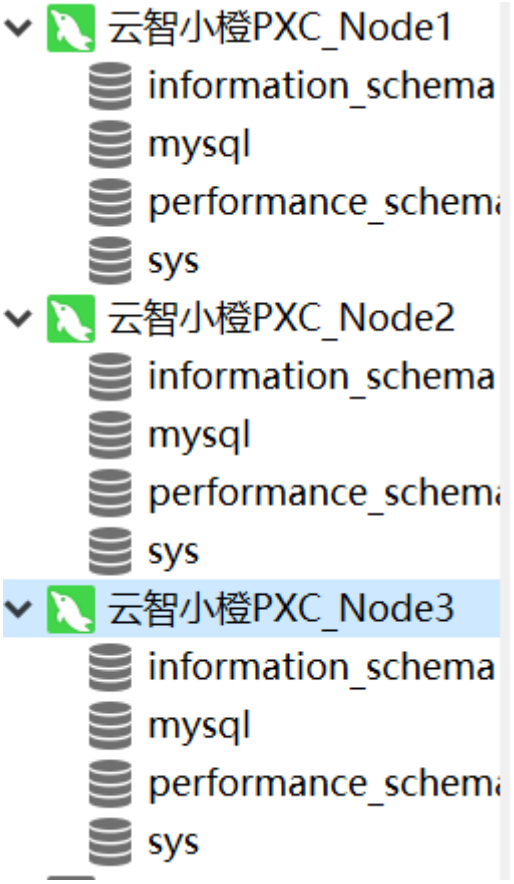
发现此时pxc集群数量2，Synced 表明数据完成了同步。

接下来同样方法启动第3个节点。

至此，pxc集群部署启动完毕。

11、验证PXC集群数据复制正确性

使用Navicat Premium连接数据库



在节点1创建数据库

在节点2创建表

在节点3添加数据

验证是否三个节点结果一致