

# 样例环境

测试环境一共5台机器分别是三台consul server两台consul agent，innosql安装在两台consul agent上 consul server的ip地址分别为192.168.2.71,192.168.2.72,192.168.2.72 consul agent的ip地址为192.168.2.61,192.168.2.62

## 1.consul

consul的二进制文件,版本号:Consul v0.5.0rc1-66-g954aec6.rc1  
(954aec66231b79c161a4122b023fbcad13047f79) consul server和consul agent 使用相同的二进制文件,使用不同的配置文件

## 2.启动consul server集群

1. 启动第一台consul server 编写consul server配置文件，文件路径为/etc/consul.d/config.json 文件内容如下:

```
{
  "bootstrap": true,
  "server": true,
  "datacenter": "dc1",
  "data_dir": "/tmp/consul",
  "node_name": <hostname>, # 根据hostname填写变量
  "addresses": {
    "http": <ip>, # 根据主机ip地址填写变量
    "rpc": <ip> # 根据主机ip地址填写变量
  }
}
```

启动consul server 命令如下: 将consul二进制文件添加到/usr/bin/

```
consul agent -config-dir /etc/consul.d/
```

1. 启动第二台consul server 编写consul server配置文件，文件路径为/etc/consul.d/config.json 文件内容如下:

```
{
  "bootstrap": false,
  "server": true,
  "datacenter": "dc1",
  "data_dir": "/tmp/consul",
  "node_name": <hostname>, # 根据hostname填写变量
  "addresses": {
    "http": <ip>, # 根据主机ip地址填写变量
    "rpc": <ip> # 根据主机ip地址填写变量
  },
  "start_join": [
    <ip> # 根据第一台consul server的ip地址填写变量
  ]
}
```

启动consul server 命令如下: consul二进制文件添加到/usr/bin/

```
consul agent -config-dir /etc/consul.d/
```

1. 启动第三台consul server 编写consul server配置文件，文件路径为/etc/consul.d/config.json 文件内容如下:

```
{
  "bootstrap": false,
  "server": true,
  "datacenter": "dc1",
  "data_dir": "/tmp/consul",
  "node_name": <hostname>, # 根据hostname填写变量
  "addresses": {
    "http": <ip>, # 根据主机ip地址填写变量
    "rpc": <ip> # 根据主机ip地址填写变量
  },
  "start_join": [
    <ip> # 根据第一台consul server的ip地址填写变量
  ]
}
```

```
"datacenter": "dc1",
"data_dir": "/tmp/consul",
"node_name": <hostname>, # 根据hostname填写变量
"addresses": {
  "http": <ip>, # 根据主机ip地址填写变量
  "rpc": <ip> # 根据主机ip地址填写变量
},
"start_join": [
  <ip>, # 根据第一台consul server的ip地址填写变量
  <ip> # 根据第二台consul server的ip地址填写变量
]
}
```

启动consul server 命令如下: 将consul二进制文件添加到/usr/bin/

```
consul agent -config-dir /etc/consul.d/
```

1. 切换第一台consul server启动模式 将第一台consul server kill掉 执行以下命令

```
pkill consul
```

重新启动第一台consul server 编写consul server配置文件, 文件路径为/etc/consul.d/config.json 文件内容如下:

```
{
  "bootstrap": false,
  "server": true,
  "datacenter": "dc1",
  "data_dir": "/tmp/consul",
  "node_name": <hostname>, # 根据hostname填写变量
  "addresses": {
    "http": <ip>, # 根据主机ip地址填写变量
    "rpc": <ip> # 根据主机ip地址填写变量
  },
  "start_join": [
    <ip>, # 根据第二台consul server的ip地址填写变量
    <ip> # 根据第三台consul server的ip地址填写变量
  ]
}
```

启动consul server 命令如下:

```
consul agent -config-dir /etc/consul.d/
```

## 启动consul agent 并加入集群

1. 启动第一台consul agent 编写consul agent配置文件, 文件路径为/etc/consul.d/config.json 文件内容如下:

```
{
  "bootstrap": false,
  "server": false,
  "datacenter": "dc1",
  "data_dir": "/tmp/consul",
  "node_name": <hostname>, # 根据hostname填写变量
  "addresses": {
    "http": <ip>, # 根据主机ip地址填写变量
    "rpc": <ip> # 根据主机ip地址填写变量
  },
  "start_join": [
    <ip>, # 根据第一台consul server的ip地址填写变量
    <ip>, # 根据第二台consul server的ip地址填写变量
    <ip> # 根据第三台consul server的ip地址填写变量
  ]
}
```

启动consul agent 命令如下: 将consul二进制文件添加到/usr/bin/

```
consul agent -config-dir /etc/consul.d/
```

1. 启动第二台consul agent 编写consul agent配置文件，文件路径为/etc/consul.d/config.json 文件内容如下：

```
{
  "bootstrap": false,
  "server": false,
  "datacenter": "dc1",
  "data_dir": "/tmp/consul",
  "node_name": "<hostname>", # 根据hostname填写变量
  "addresses": {
    "http": "<ip>", # 根据主机ip地址填写变量
    "rpc": "<ip>" # 根据主机ip地址填写变量
  },
  "start_join": [
    "<ip>", # 根据第一台consul server的ip地址填写变量
    "<ip>", # 根据第二台consul server的ip地址填写变量
    "<ip>" # 根据第三台consul server的ip地址填写变量
  ]
}
```

启动consul agent 命令如下: 将consul二进制文件添加到/usr/bin/

```
consul agent -config-dir /etc/consul.d/
```

## 在所有consul agent上启动innosql服务,配置主从关系

### 在所有consul agent 上注册服务

- 1.准备服务状态检查脚本,检查脚本内容如下：

```
[root@consul-agent1 ~]# cat mysqlcheck.sh
#!/bin/bash
name=$1
dbname=`tr 'A-Z' 'a-z' <<<"$name"`
#basedir="/bpgchina/${dbname}/MYBASE"
datadir="/var/lib/mysql"
bindir="/bin"

MYSQL=$bindir/mysql
MYSQL_USER=<username> #填写innosql登陆用户名
MYSQL_PASSWORD=<password> #填写innosql登陆密码
CHECK_TIME=3
MYSQL_OK=1
function check_mysql_health () {

    $MYSQL -u $MYSQL_USER -p$MYSQL_PASSWORD -S $datadir/mysql.sock -e "show status;" > /dev/null 2>&1

    if [ $? -eq 0 ]; then
        MYSQL_OK=0;
    else
        MYSQL_OK=1;
    fi

    return $MYSQL_OK;
}

while [ $CHECK_TIME -ne 0 ]
do
    let "CHECK_TIME -= 1";
    check_mysql_health;
    if [ $MYSQL_OK = 0 ]; then
        CHECK_TIME=0;
        exit 0
    fi
done
```

```
fi
if [ $MYSQL_OK -eq 1 ] && [ $CHECK_TIME -eq 0 ]; then
    exit 2;
fi
sleep 1;
done
```

## 2.使用consul api接口向consul agent注册innosql服务

```
curl -X PUT -d '{"Name":"innosql","Tags":["master"],"Port":3306,"Check":{"Script":"/root/mysqlcheck.sh","Interval":"10s"}}'
```

## 3.使用consul api向consul service插入service/innosql/leader键(用于应用客户端查询innosql服务连接串)

```
curl -X PUT http://192.168.2.71:8500/v1/kv/service/innosql/leader
```

# 启动服务故障自动切换机制

## 1.修改innosql master上bootstrap/conf目录下面的app.conf,路径/root/bootstrap文件内容如下:

```
appname = bootstrap
runmode = dev
loglevel = 7
hostname = consul-agent1
ip = 192.168.2.61
port = 3306
username = root
password = 111111
datacenter =
token =
service_ip = 192.168.2.71
service_port = 8500
servicename = innosql
```

将上面配置文件中以下字段修改为相应的配置 hostname为consul agent主机名(也就是innosql master的主机名) ip 为innosql 的ip地址 port为innosql 的端口号 username为innosql用户名 password为innosql的密码 service\_ip 为consul server集群中任意一台ip

## 2.在innosql master节点运行bootstrap二进制文件 3.修改mha-handlers/conf目录下面的app.conf,路径/root/mha-handlers文件内容如下:

```
appname = mha-handlers
runmode = dev
loglevel = 7
hostname = consul-agent1
ip = 192.168.2.61
port = 3306
username = root
password = 111111
datacenter =
token =
service_ip = 192.168.2.71
service_port = 8500
servicename = innosql
```

将上面配置文件中以下字段修改为相应的配置 hostname为consul agent主机名(也就是innosql master的主机名) ip 为innosql 的ip地址 port为innosql 的端口号 username为innosql用户名 password为innosql的密码 service\_ip 为consul server集群中任意一台ip 4.编写watch.json配置文件,文件路径/etc/consul.d/watch.json,文件内容如下:

```
[root@consul-agent1 ~]# cat watch.json
{
  "watches": [
    {
      "type": "key",
```

```
"key": "service/innosql/leader",
"handler": "/root/mha-handlers/mha-hacondlers"
}
]
}
```

4.在consul agent上reload配置,命令如下

```
consul reload -rpc-addr=<ip>:8400 #填写consul agent 配置文件里面的rpc地址
```

5.修改innosql slave 上mha-handlers/conf目录下面的app.conf,路径/root/mha-handlers文件内容如下:

```
appname = mha-handlers
runmode = dev
loglevel = 7
hostname = consul-agent1
ip = 192.168.2.61
port = 3306
username = root
password = 111111
datacenter =
token =
service_ip = 192.168.2.71
service_port = 8500
servicename = innosql
```

将上面配置文件中以下字段修改为相应的配置 hostname为consul agent主机名(也就是innosql slave的主机名)  
ip 为innosql slaveip地址 port为innosql slave端口号 username为innosql slave用户名 password为innosql slave  
密码 service\_ip为consul server集群中任意一台ip 6..编写watch.json配置文件,文件路  
径/etc/consul.d/watch.json,文件内容如下:

```
[root@consul-agent2 ~]# cat watch.json
{
  "watches": [
    {
      "type": "key",
      "key": "service/innosql/leader",
      "handler": "/root/mha-handlers/mha-handlers"
    }
  ]
}
```

4.在consul agent上reload配置,命令如下

```
consul reload -rpc-addr=<ip>:8400 #填写consul agent 配置文件里面的rpc地址
```

### 环境搭建完成