杭研直存测试

阿里云直存参考资料

```
操作步骤参考：
https://help.aliyun.com/document_detail/100624.htm?spm=a2c4g.11186623.2.5.6ab26d13yg3tic#concept-xzh-nzk-2gb
```

查看现网流量监控

```
Cacti查询上联带宽http://10.168.71.59/index.php
jiankong    mxEsJT04$*q7
```

如果系统是bclinux的话，yum一般是不可达，需要重新下载centos的源（详见《BC_Linux源换成Centos源》）

程序运行所需的依赖环境，可直接使用批量安装脚本进行快速安装，使用批量脚本前，需要先对云主机进行互信设置。

互信设置方法

```
1.在master上执行：ssh-keygen 一直回车  如果已经有公钥和私钥  这一步就不要执行了

2.在master上执行：yum -y install expect

3.新建互信脚本huxin.sh，内容如下，然后执行sh huxin.sh  （建议创建的所有云主机的密码设置成一样）

cat huxin.sh

#!/bin/bash

###需要做互信的机器地址，移动云上同一个vpc下，需要填内网IP地址###

SERVERS="36.134.123.60 36.134.123.11"

###需要做互信机器的密码，建议都设置成一样###

PASSWORD=Passw0rd@

auto_ssh_copy_id() {

expect -c "set timeout -1;

spawn ssh-copy-id $1;

expect {

*(yes/no)* {send -- yes\r;exp_continue;}

*assword:* {send -- $2\r;exp_continue;}

eof {exit 0;}

}";

}

ssh_copy_id_to_all() {

for SERVER in $SERVERS

do

auto_ssh_copy_id $SERVER $PASSWORD

done

}

ssh_copy_id_to_all
```

如果执行互信的脚本报错

```
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/root/.ssh/id_rsa.pub"
```

需要在安全组中，创建规则，放开安全组间的权限

9852c9c2-b44a-40c3-84e4-353f34dd3...	所有协议	所有端口	流入	安全组访问	lp_beijing1_sg_01	IPv4	修改 删除
892c3b39-be06-4a60-8b52-2cfc98cc3...	所有协议	所有端口	流入	地址段访问	223.104.148.94/32	IPv4	修改 删除
88349742-761f-407b-b4a8-862bb4fc0...	所有协议	所有端口	流入	地址段访问	223.112.105.132/32	IPv4	修改 删除

master节点安装Python3

```
yum -y install python3
```

安装pip

```
curl https://bootstrap.pypa.io/get-pip.py -o get-pip.py

python3 get-pip.py  #pip关联到Python哪个版本，就使用python2或python3执行安装
注：
    关联pip如果报错

WARNING: Retrying (Retry(total=0, connect=None, read=None, redirect=None, status=None)) after connection broken by
'NewConnectionError('<pip._vendor.urllib3.connection.HTTPSConnection object at 0x7f963cf0e4a8>: Failed to establish a new connection: [Errno
101] Network is unreachable',)': /simple/pip/
ERROR: Could not find a version that satisfies the requirement pip
ERROR: No matching distribution found for pip
```

说明网络有点问题，可以使用国内的镜像源加速，将python3 get-pip.py改成

python3 get-pip.py -i http://pypi.douban.com/simple/ --trusted-host pypi.douban.com

检查pip关联的python版本

```
[root@lp-shanghai-hyzc-a-001 ~]#  pip -V
pip 21.0.1 from /usr/local/lib/python3.6/site-packages/pip (python 3.6)
```

运行环境缺少requests boto3 botocore psutil 四个库文件，使用pip安装

```
pip install requests boto3 botocore psutil
注：
    如果报错和关联pip时报错一样，也使用国内的镜像源加速
```

pip install requests boto3 botocore psutil -i http://pypi.douban.com/simple/ --trusted-host pypi.douban.com

安装gevent模块

```
pip3 install gevent
```

在主节点给所有客户端批量安装，下方脚本的SERVERS去掉了主节点的IP（主节点刚刚安装过了）

```
[root@lp-shanghai-hyzc-a-001 ~]#  cat install_all.sh
#!/bin/bash


SERVERS="192.168.30.2 192.168.30.3 192.168.30.4 192.168.30.5 192.168.30.6 192.168.30.7 192.168.30.9 192.168.30.11 192.168.30.14"


for SERVERS in $SERVERS;
do
#echo $SERVERS;
ssh root@$SERVERS 'yum -y install python3;curl https://bootstrap.pypa.io/get-pip.py -o get-pip.py;python3 get-pip.py;pip install requests boto3
botocore psutil gevent';
done
```

云主机安装s3cmd，使用s3cmd单独验证集群环境是否正常

```
pip install s3cmd
```

新建s3cfg文件，内容如下

```
[root@lp-shanghai-hyzc-a-001 ~]#  cat ningbo2.s3cfg
[default]
access_key = JRFEGEI16MM6XPGUW0EB1
secret_key = b4PQtzkgvg1AsJTMu76nf8XRurBBhxWw65y8TeGy1
host_base = eos-ningbo-3.cmecloud.cn
host_bucket = eos-ningbo-3.cmecloud.cn
access_token =
add_encoding_exts =
add_headers =
bucket_location = US
ca_certs_file =
cache_file =
check_ssl_certificate = True
check_ssl_hostname = True
cloudfront_host = cloudfront.amazonaws.com
content_disposition =
content_type =
default_mime_type = binary/octet-stream
delay_updates = False
delete_after = False
delete_after_fetch = False
delete_removed = False
dry_run = False
enable_multipart = True
encrypt = False
expiry_date =
expiry_days =
expiry_prefix =
follow_symlinks = False
force = False
get_continue = False
gpg_command = /usr/bin/gpg
gpg_decrypt = %(gpg_command)s -d --verbose --no-use-agent --batch --yes --passphrase-fd %(passphrase_fd)s -o %(output_file)s %(input_file)s
gpg_encrypt = %(gpg_command)s -c --verbose --no-use-agent --batch --yes --passphrase-fd %(passphrase_fd)s -o %(output_file)s %(input_file)s
gpg_passphrase =
guess_mime_type = True
human_readable_sizes = False
invalidate_default_index_on_cf = False
invalidate_default_index_root_on_cf = True
invalidate_on_cf = False
kms_key =
limit = -1
limitrate = 0
list_md5 = False
log_target_prefix =
long_listing = False
max_delete = -1
mime_type =
multipart_chunk_size_mb = 15
multipart_max_chunks = 10000
preserve_attrs = True
progress_meter = True
proxy_host =
proxy_port = 0
put_continue = False
recursive = False
recv_chunk = 65536
reduced_redundancy = False
requester_pays = False
restore_days = 1
restore_priority = Standard
send_chunk = 65536
server_side_encryption = False
signature_v2 = True
signurl_use_https = False
simpledb_host = sdb.amazonaws.com
skip_existing = False
socket_timeout = 300
stats = False
stop_on_error = False
storage_class =
throttle_max = 100
upload_id =
urlencoding_mode = normal
use_http_expect = False
use_https = False
use_mime_magic = True
verbosity = WARNING
website_endpoint = http://%(bucket)s.s3-website-%(location)s.amazonaws.com/
website_error =
website_index = index.html
```

查询桶信息

```
[root@lp-shanghai-hyzc-a-001 ~]#  s3cmd -c ningbo2.s3cfg  ls
2021-04-06 06:37  s3://ningbo2-test-bucket01
[root@lp-shanghai-hyzc-a-001 ~]#  s3cmd -c ningbo2.s3cfg info s3://ningbo2-test-bucket01
s3://ningbo2-test-bucket01/ (bucket):
   Location:  ningbo3
   Payer:     BucketOwner
   Expiration Rule: none
   Policy:    none
   CORS:      <CORSConfiguration xmlns="http://s3.amazonaws.com/doc/2006-03-01/"><CORSRule><ID>a6fc80c1b58348d494932bf57348acd8</ID>
<AllowedMethod>GET</AllowedMethod><AllowedMethod>PUT</AllowedMethod><AllowedMethod>DELETE</AllowedMethod><AllowedMethod>HEAD</AllowedMethod>
<AllowedMethod>POST</AllowedMethod><AllowedOrigin>*</AllowedOrigin><AllowedHeader>*</AllowedHeader><MaxAgeSeconds>100</MaxAgeSeconds>
<ExposeHeader>ETag</ExposeHeader></CORSRule><CORSRule><ID>c5a1a1968a084404b8396f6bb1a6884d</ID><AllowedMethod>GET</AllowedMethod>
<AllowedMethod>PUT</AllowedMethod><AllowedMethod>DELETE</AllowedMethod><AllowedMethod>HEAD</AllowedMethod><AllowedMethod>POST</AllowedMethod>
<AllowedOrigin>https://console-ningbo-2.cmecloud.cn:*</AllowedOrigin><AllowedHeader>*</AllowedHeader><MaxAgeSeconds>100</MaxAgeSeconds>
<ExposeHeader>ETag</ExposeHeader></CORSRule></CORSConfiguration>
   ACL:       csb_zhuanxiang_luopeng: FULL_CONTROL
```

查看对象是否存在

```
对象存在
[root@lp-shanghai-hyzc-a-001 ~]#  s3cmd -c ningbo2.s3cfg info s3://ningbo2-test-bucket01/shat091758t_a90000
s3://ningbo2-test-bucket01/shat091758t_a90000 (object):
   File size: 1536000
   Last mod:  Fri, 09 Apr 2021 10:17:12 GMT
   MIME type: application/octet-stream
   Storage:   STANDARD
   MD5 sum:   f1083baf289cc95cbb4eaf0c90f9b51d
   SSE:       none
   Policy:    none
   CORS:      <CORSConfiguration xmlns="http://s3.amazonaws.com/doc/2006-03-01/"><CORSRule><ID>a6fc80c1b58348d494932bf57348acd8</ID>
<AllowedMethod>GET</AllowedMethod><AllowedMethod>PUT</AllowedMethod><AllowedMethod>DELETE</AllowedMethod><AllowedMethod>HEAD</AllowedMethod>
<AllowedMethod>POST</AllowedMethod><AllowedOrigin>*</AllowedOrigin><AllowedHeader>*</AllowedHeader><MaxAgeSeconds>100</MaxAgeSeconds>
<ExposeHeader>ETag</ExposeHeader></CORSRule><CORSRule><ID>c5a1a1968a084404b8396f6bb1a6884d</ID><AllowedMethod>GET</AllowedMethod>
<AllowedMethod>PUT</AllowedMethod><AllowedMethod>DELETE</AllowedMethod><AllowedMethod>HEAD</AllowedMethod><AllowedMethod>POST</AllowedMethod>
<AllowedOrigin>https://console-ningbo-2.cmecloud.cn:*</AllowedOrigin><AllowedHeader>*</AllowedHeader><MaxAgeSeconds>100</MaxAgeSeconds>
<ExposeHeader>ETag</ExposeHeader></CORSRule></CORSConfiguration>
   ACL:       csb_zhuanxiang_luopeng: FULL_CONTROL
对象不存在
[root@lp-shanghai-hyzc-a-001 ~]#  s3cmd -c ningbo2.s3cfg info s3://ningbo2-test-bucket01/shat150914t_a1
ERROR: S3 error: 404 (Not Found)
```

在底层创建管理员用户，然后用管理员用户给OP的用户进行授权（底层的管理员用户是让SRE登录集群执行的命令，OP上无法创建）

lpadminuser01为底层创建的，命令为

```
radosgw-admin user create --uid lpadminuser01 --display-name lpadminuser01 --access-key lpadminuser01 --secret-key lpadminuser01 --system
```

如下lp_allow_user.py中，授权的信息为policy中的内容，脚本意思为使用创建的admin账号lpadminuser01，给移动云OP账号UserName="3156253c90574882bfee65965676e7ae"，进行授权，授权内容为policy中的Allow、sts:GetSessionToken，不进行此步骤，则无法调通GetSessionToken接口。

UserName为OP上的账号及子账号菜单中的"账号ID"，执行如下py脚本后会返回200，授权成功

```
D:\工作\进行中ing\杭研直存测试\hy_zhicun\allow_user.py
[root@ceph109 teststs]# cat lp_allow_user.py
from boto3.session import Session
import boto3
access_key = "lpadminuser01"
secret_key = "lpadminuser01"
url = "http://eos-ningbo-3.cmecloud.cn"
session = Session(access_key, secret_key)

config_dict = { 'signature_version' : 's3', 'connect_timeout': 30000, 'read_timeout': 30000}
configuration = boto3.session.Config(**config_dict)
client = boto3.client('iam',
aws_access_key_id=access_key,
aws_secret_access_key=secret_key,
endpoint_url=url,
region_name='',
use_ssl = False,
config = configuration,
)
policy = '''{
  "Version": "2012-10-17",
    "Statement": [{
      "Effect": "Allow",
      "Action": "sts:GetSessionToken",
      "Resource": "*"
    }]
}'''
response = client.put_user_policy(
    UserName="3156253c90574882bfee65965676e7ae",
    PolicyName='admin-sts',
    PolicyDocument= policy
)

response2 = client.get_user_policy(
    UserName='3156253c90574882bfee65965676e7ae',
    PolicyName='admin-sts'
)
print response2["ResponseMetadata"]["HTTPStatusCode"]
```

查看是否授权成功

```
[root@HHHT-PSC-P11F1-SPOD3-PM-OS01-BCONEST-ACCESSER-01 ~]# ceph df
RAW STORAGE:
    CLASS     SIZE        AVAIL       USED       RAW USED     %RAW USED
    ssd       314 TiB     314 TiB     40 GiB      460 GiB          0.14
    TOTAL     314 TiB     314 TiB     40 GiB      460 GiB          0.14


POOLS:
    POOL                                   ID      STORED      OBJECTS     USED        %USED     MAX AVAIL
    .rgw.root                               1      25 KiB          52      612 KiB         0        99 TiB
    huhehaote3-zone1.rgw.log                5      99 MiB     111.58k       99 MiB         0        99 TiB
    huhehaote3-zone1.rgw.meta               6     337 KiB         997       12 MiB         0        99 TiB
    huhehaote3-zone1.rgw.control            7         0 B           8          0 B         0        99 TiB
    huhehaote3-zone1.rgw.buckets.non-ec     8      18 KiB          22      279 KiB         0        99 TiB
    huhehaote3-zone1.rgw.buckets.index      9     1.3 GiB      59.76M      1.3 GiB         0        99 TiB
    huhehaote3-zone1.rgw.buckets.data      10         0 B       6.35M          0 B         0        99 TiB

[root@HHHT-PSC-P11F1-SPOD3-PM-OS01-BCONEST-ACCESSER-01 ~]# rados -p huhehaote3-zone1.rgw.meta getxattr cyruser01 user.rgw.user-policy --namespace
users.uid
cyruser01-sts&{
"Version": "2012-10-17",
  "Statement": [{
    "Effect": "Allow",
    "Action": "sts:GetSessionToken",
    "Resource": "*"
  }]
}

备注：huhehaote3-zone1.rgw.meta通过ceph df查看，改成当前资源池的具体信息；cyruser01授权用户的uid
```

测试直存的python代码

```
D:\工作\进行中ing\杭研-呼和浩特直存\s3.py
```

先在单客户端执行python代码

```
测试获取STS
python3 /root/s3-cyr151600.py --access_key=JRFEGEI16MM6XPGUW0EB --secret_key=b4PQtzkgvg1AsJTMu76nf8XRurBBhxWw65y8TeGy --endpoint_url=http://eos-
ningbo-3.cmecloud.cn --bucket=ningbo2-test-bucket01 -t 60 -n 1000 -S -d 3600

测试上传upload/下载download/删除delete文件
python3 s3-cyr151600.py --access_key=JRFEGEI16MM6XPGUW0EB --secret_key=b4PQtzkgvg1AsJTMu76nf8XRurBBhxWw65y8TeGy --endpoint_url=http://eos-ningbo-
3.cmecloud.cn --bucket=ningbo2-test-bucket01 -t 15 -n 16800 -m upload -p suzhou1_$i -s 1.5MB -g 10 -i 5
```

多客户端执行测试。将要执行的python脚本scp到所有的客户端

```
for i in 2 3 4 5 6 7 9 11 14;do scp s3-cyr151600.py root@192.168.30.$i:/root;done
```

在主节点弄个shell脚本，让所有客户端同时执行s3-cyr151600.py脚本

获取STS

```
[root@lp-suzhou-az1-a-001 ~]#  cat getsts.sh
#!/bin/bash

for i in 26 9 23 3 18 2 20 4 7 27;
do
        ssh root@192.168.40.$i "python3 /root/s3-cyr151104.py --access_key=JRFEGEI16MM6XPGUW0EB --
secret_key=b4PQtzkgvg1AsJTMu76nf8XRurBBhxWw65y8TeGy --endpoint_url=http://eos-ningbo-3.cmecloud.cn --bucket=ningbo2-test-bucket01 -t 60 -n 1000 -
S -d 3600 & " &
done

设置crontab，让脚本每隔半小时运行一次（只要在主节点设置就行）
需要先给脚本加上可读可写可执行的权限，否则无法执行
chmod 777 getsts.sh
在主节点执行# crontab -e
编辑"*/30 * * * *  /root/getsts.sh"，:wq保存退出

# crontab -l，可查看定时任务是否设置成功
[root@lp-suzhou-az1-a-001 ~]#  crontab -l
*/30 * * * *  /root/getsts.sh
```

write文件

```
[root@lp-shanghai-hyzc-a-001 ~]#  cat put.sh
#!/bin/bash

for i in 2 3 4 5 6 7 9 11 14 18;
do
    ssh root@192.168.30.$i "python3 s3-cyr151600.py --access_key=JRFEGEI16MM6XPGUW0EB --secret_key=b4PQtzkgvg1AsJTMu76nf8XRurBBhxWw65y8TeGy --
endpoint_url=http://eos-ningbo-3.cmecloud.cn --bucket=ningbo2-test-bucket01 -t 15 -n 16800 -m upload -p shanghai1_$i -s 1.5MB -g 10 -i 5 & " &
done
备注：
1、每个客户端的对象前缀需要保持不同，所以前缀定为shanghai1_$i，让i获取每个客户端的内网ip地址的最后一位，这样的话，命令就需要写在双引号中，不能写在单引号中。
2、实际运行中发现：客户端跑一段时间后，带宽会降低到400M以下或者脚本已经运行结束，这是需要重新执行测试命令，并且需要更改下对象前缀名，需要将put.sh进行以下调整，并写一个守护
进程，并让守护进程定时运行，一旦检测到当前客户端的带宽低于400，或当前客户端的脚本进程不在，就自动拉起进程

    [root@lp-shanghai-hyzc-a-001 ~]#  cat put_161510-da.sh
    #!/bin/bash

    for i in 2 3 4 5 6 7 9 11 14 18;
    do
            ssh root@192.168.30.$i "echo 1 > /root/jilu.txt; python3 s3-cyr151600.py --access_key=JRFEGEI16MM6XPGUW0EB --
    secret_key=b4PQtzkgvg1AsJTMu76nf8XRurBBhxWw65y8TeGy --endpoint_url=http://eos-ningbo-3.cmecloud.cn --bucket=ningbo2-test-bucket01 -t 15 -n
    16800 -m upload -p shanghai1_$i -s 1.5MB -g 10 -i 5 & " &
    done
守护进程

    [root@lp-suzhou-az1-a-001 ~]#  cat check.py
    #!/usr/bin/python
    # coding:utf-8

    import os
    import datetime
    import time
    import subprocess
    import socket

    # record test args
    test_num = "/root/jilu.txt"

    # test python script
    test_script = "s3-cyr151600.py"

    #get the last ip num
    def get_ipaddr():
        hostname = socket.gethostname()
        ipaddr = socket.gethostbyname(hostname)
        ip = ipaddr.strip('.')[-1]
        return ip

    #get bandwidth
    def get_dstat():
        bandwidths = []
        for i in range(3):
            cmd = "dstat -n 1 3| awk '{print $2}'|tail -n 1"
            bandwidth = subprocess.Popen(cmd, shell=True, stdout=subprocess.PIPE).communicate()[0].strip('\n')
            bandwidths.append(bandwidth)
        bandwidths_sum = 0
        for bandwidth in bandwidths:
            if bandwidth.endswith('M') or bandwidth.endswith('m'):
                bandwidths_sum += int(bandwidth[:-1])
        bandwidths_ave = int(bandwidths_sum/3)
        return bandwidths_ave

    def check_process():
        p1 = subprocess.Popen(['ps', 'aux'], stdout=subprocess.PIPE)
        p2 = subprocess.Popen(['grep', test_script], stdin=p1.stdout, stdout=subprocess.PIPE)
        p3 = subprocess.Popen(['grep', '-v', 'grep'], stdin=p2.stdout, stdout=subprocess.PIPE)
        p4 = subprocess.Popen(['grep', '-v', 'ssh'], stdin=p3.stdout, stdout=subprocess.PIPE)
        out = p4.communicate()[0]
        ip = get_ipaddr()
        bandwidths_ave = get_dstat()
    #    print "bandwidths_ave: " + str(bandwidths_ave)
    #    print "out: " + str(out)

        if not out or bandwidths_ave <= 400:
            subprocess.Popen("ps aux|grep %s|grep -v ssh|grep -v grep|awk '{print $2}'|while read line; do kill -9 $line; done" %test_script,
    shell=True)
            pid = subprocess.Popen(["cat", test_num], stdout=subprocess.PIPE).communicate()[0].strip("\n")
            pid2 = int(pid) + 1
            subprocess.Popen(
                "python3 %s --access_key=JRFEGEI16MM6XPGUW0EB --secret_key=b4PQtzkgvg1AsJTMu76nf8XRurBBhxWw65y8TeGy --endpoint_url=http://eos-
    ningbo-3.cmecloud.cn --bucket=ningbo2-test-bucket01 -t 15 -n 16800 -m upload -p suzhou%d_%d -s 1.5MB -g 10 -i 5  & " % (
                test_script, int(pid), int(ip)), shell=True)
            with open(test_num, 'w') as fp:
                fp.write(str(pid2))

    if __name__ == "__main__":
        check_process()

守护进程脚本需要设置在每一个客户端上（因为每个客户端的带宽变化和运行时间不一样），设置每一分钟，执行一次（具体设置方法见《crontab定时任务》）

    [root@lp-suzhou-az1-a-001 ~]#  crontab -l
    */1 * * * *  python check.py >/dev/null
```