Falkon server and swift in the same machine

Falkon worker in every node

计算流程是这样的，提交swift任务给swift服务器，swift服务器通过falkon进行任务调度分发。

我们现在的实现是：swift封装了一个服务，向opennebula提出资源申请（对应你这就应该是向ec2提出资源申请），然后opennebula会启动一个包含一个falkon server和多个 falkon worker的集群。这里我们有一个脚本在完成falkon server启动后，会将falkon server的ip地址通知给swift和所有的falkon worker，worker拿到这个ip地址后，将主动注册到server上，swift则会根据这个ip生成调度连接，把swift的调度提交到falkon server上，完成任务的分发。

所以这整个过程只需要1个swift服务器，和多个falkon集群。

swift服务可以选择提交到不同的falkon集群。

另外需要注意一个问题就是，swift上面的swift脚本要使用的数据目录必须网络共享的，每个falkon worker 必须能够访问到。

镜像文件中的配置：jdk，ssh，opennebula的初始化shell脚本，falkon

Montage地址：<http://montage.ipac.caltech.edu/>

## Falkon安装和设置

Falkon的安装参考Falkon\_User\_Guide\_v2.pdf

### Falkon环境变量的设置

$sudo nano /etc/profile

export FALKON\_HOME=/home/cloudian/falkon.r174

export FALKON\_ROOT=/home/cloudian/falkon.r174

修改falkon.env文件，替换FALKON\_HOME和FALKON\_ROOT

### Falkon代码修改

修改源码，使worker注册后，service通过ip地址找到worker

$cd ${FALKON\_HOME}/worker/org/globus/GenericPortal/clients/GPService\_instance

$vi workerRun.java

“ / ”搜索getMachNamePort

注释掉machName = java.net.InetAddress.getLocalHost().getCanonicalHostName();

打开machName = java.net.InetAddress.getLocalHost().getHostAddress();

再对Falkon源码进行编译

$falkon-build.sh

修改swift的sites.xml中的IP，swift.properties中的ip；修改falkon server中的hosts的IP。

Falkon worker中的falkon代码修改$FALKON\_HOME/worker/org/globus/GenericPortal/clients/FactoryService\_GP/ClientCreate 修改将

|  |
| --- |
| public static void writeTimetofile(String content) throws IOException{  FileOutputStream fileOS = new FileOutputStream("/home/youfuli-swift/nfs/workercreate.txt",true);  OutputStreamWriter osWriter=new OutputStreamWriter(fileOS,"utf-8");  osWriter.write(content);  osWriter.close();  fileOS.close();  } |

|  |
| --- |
| try{  writeTimetofile("the create time="+startTime+"ms \n");  }  catch(Exception e){} |

|  |
| --- |
| writeTimetofile("the end time="+endTime+"ms \n");  writeTimetofile("the worker create time="+(endTime-startTime)+"ms \n"); |

添入其中，以计算worker create时间。

$FALKON\_HOME/worker/org/globus/GenericPortal/clients/GPService\_instance/WorkerRun.java

|  |
| --- |
| //changed by youfuli to get the time  public static void writeTimetofile(String content) throws IOException{  FileOutputStream fileOS = new FileOutputStream("/home/youfuli-swift/nfs/workerregister.txt",true);  OutputStreamWriter osWriter=new OutputStreamWriter(fileOS,"utf-8");  osWriter.write(content);  osWriter.close();  fileOS.close();  } |

也是添加类似的。计算register和de-register. 需要先找到相应的函数。

在falkon worker中找到jdk/jre/lib/security目录下的java.security文件  
把securerandom.source=file:/dev/urandom改成securerandom.source=file:/dev/./urandom 这样可以提高worker的注册速度

在/home/Ubuntu/.profile文件中加

|  |
| --- |
| #we use the following shell to change the ip in host, which allocated by amazon.  CLOUDSOLUTIONIPADDRESS=`/sbin/ifconfig eth0 | grep 'inet addr' | cut -d ':' -f 2 |cut -d ' ' -f 1`  AddressCount=`grep -c 'internal' /etc/hosts`  if [ $AddressCount -eq 2 ];then  sudo sed -i '10d' /etc/hosts  fi  sudo sed -i 's/127.0.1.1/'$CLOUDSOLUTIONIPADDRESS'/g' /etc/hosts |

修改Amazon EC分配的