Ambari 2.7.4+HDP 3.1.4 For CentOS 7 64位安装

作者：月光蚁族

第一章 Ambari 2.7.4安装配置

--

1. 集群节点相关

1.1. 集群节点信息

wanfeng hdp cluster
192.168.11.10 ambari.wanfeng.com ambari
192.168.11.8 client-v01.wanfeng.com client-v01
192.168.11.2 client-v02.wanfeng.com client-v02
192.168.11.25 master-v01.wanfeng.com master-v01
192.168.11.26 master-v02.wanfeng.com master-v02
192.168.11.23 datanode-v01.wanfeng.com datanode-v01
192.168.11.24 datanode-v02.wanfeng.com datanode-v02
192.168.11.12 datanode-v03.wanfeng.com datanode-v03
192.168.11.13 datanode-v04.wanfeng.com datanode-v04
192.168.11.14 datanode-v05.wanfeng.com datanode-v05

wanfeng freeipa cluster
192.168.11.5 ipa-v01.wanfeng.com ipa-v01
192.168.11.6 ipa-v02.wanfeng.com ipa-v02

注意： datanode-v[04,05].wanfeng.com 两台服务器，在本文档中先不添加到Ambari 管理，后续单独添加。

1.2. 节点说明

1.2.1 ambari.wangfeng.com

主要用于 Ambari管理(ambari-server)
用于存放元数据的MySQL数据库也安装到该节点（元数据主要包括：Ambari, Hive, Ranger, Hue, Azkaban, Superset等）

1.2.2 client-v[01,02].wangfeng.com

两台客户端机器，均安装 Azkaban-exec; Azkaban-web 独立安装(Azkaban-web不支持HA)到 client-v02.wangfeng.com

1.2.3 两台 Master 机器 master-v[01,02].wangfeng.com

基中主要包括：hdfs, yarn, hbase 的高可用相关进程; Hive metadata, Hive server进程 及 Zookeeper进程等

1.2.4 5台 worker 节点 datanode-v[01~05].wangfeng.com

主要是 Datanode, NodeManager, HRegion Server等 进程

1.2.5 ipa-v[01,02].wangfeng.com

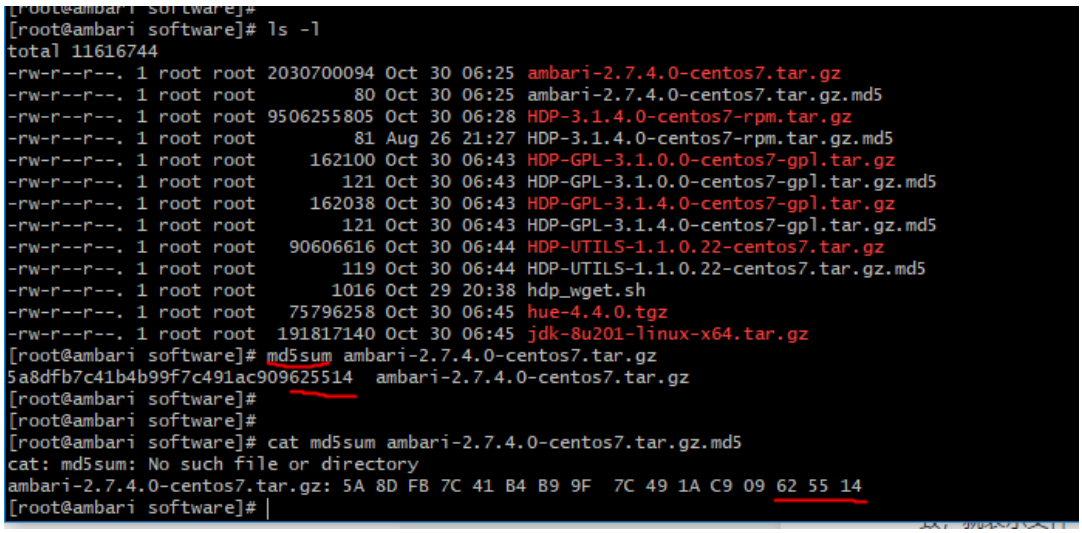
主要用于 FreeIPA“互为主从的两个节点”，FreeIPA是一个集 kerberos, ldap, dns于一体的管理工具，
详细请参考：https://www.freeipa.org/page/Main_Page

2. 资源下载(先执行 yum -y install wget 命令安装 wget)

```
# ambari
wget http://public-repo-1.hortonworks.com/ambari/centos7/2.x/updates/2.7.4.0/ambari-2.7.4.0-centos7.tar.gz
wget http://public-repo-1.hortonworks.com/ambari/centos7/2.x/updates/2.7.4.0/ambari-2.7.4.0-centos7.tar.gz.md5
# hdp
wget http://public-repo-1.hortonworks.com/HDP/centos7/3.x/updates/3.1.4.0/HDP-3.1.4.0-centos7-
```

rpm.tar.gz
wget http://public-repo-1.hortonworks.com/HDP/centos7/3.x/updates/3.1.4.0/HDP-3.1.4.0-centos7-rpm.tar.gz.md5
hdp gpl
wget http://public-repo-1.hortonworks.com/HDP-GPL/centos7/3.x/updates/3.1.4.0/HDP-GPL-3.1.4.0-centos7-gpl.tar.gz
wget http://public-repo-1.hortonworks.com/HDP-GPL/centos7/3.x/updates/3.1.4.0/HDP-GPL-3.1.4.0-centos7-gpl.tar.gz.md5
hdp utils
wget http://public-repo-1.hortonworks.com/HDP-UTILS-1.1.0.22/repos/centos7/HDP-UTILS-1.1.0.22-centos7.tar.gz
wget http://public-repo-1.hortonworks.com/HDP-UTILS-1.1.0.22/repos/centos7/HDP-UTILS-1.1.0.22-centos7.tar.gz.md5

备注1：其中 .md5 文件是校验文件，比如 我想检验 ambari-2.7.4.0-centos7.tar.gz 这个文件下载是否完整，可以执行 md5sum ambari-2.7.4.0-centos7.tar.gz 命令，得到的结果与 .md5文件比较如果一致，就表示文件下载完整，如下图。(建议大家都习惯性的校验文件，这比起你使用一个不正确的安装文件导致报一些莫名其妙的错误去折腾半天要省太多时间。一般最后6位对上了，就基本OK了)



备注2：Ambari & HDP的版本选型，请参考 HDP 版本矩阵图，链接是

https://supportmatrix.hortonworks.com/

备注3: 根据HDP 版本矩阵图可知 HDP 3.1.4 版本只支持 JDK 1.8,请自己下载相关安装包(我下载的是 jdk-8u201-linux-x64.tar.gz 安装包)

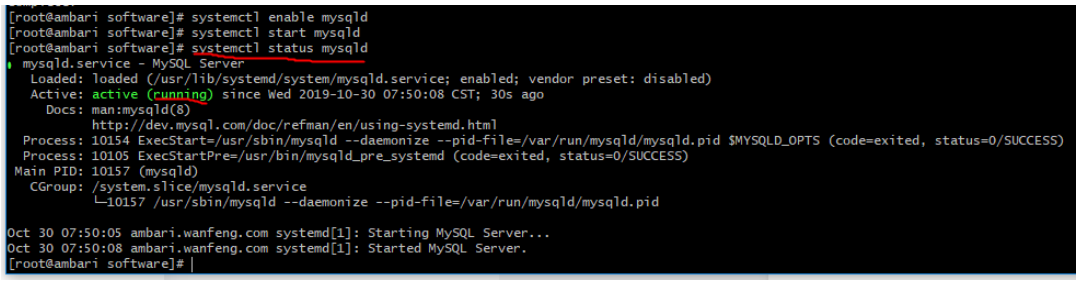
3. MySQL5.7 安装到 ambari 服务器(如果是线上环境，建议MySQL高可用)

参考：https://linuxize.com/post/install-mysql-on-centos-7/

3.1 以 root 用户在 ambari.wanfeng.com 服务器执行如下命令

```
yum -y localinstall https://dev.mysql.com/get/mysql57-community-release-el7-11.noarch.rpm
yum -y install mysql-community-server
systemctl enable mysqld
systemctl start mysqld
systemctl status mysqld
```

执行 systemctl status mysqld 后，可以看到MySQL服务已经是 running 状态，如下图。

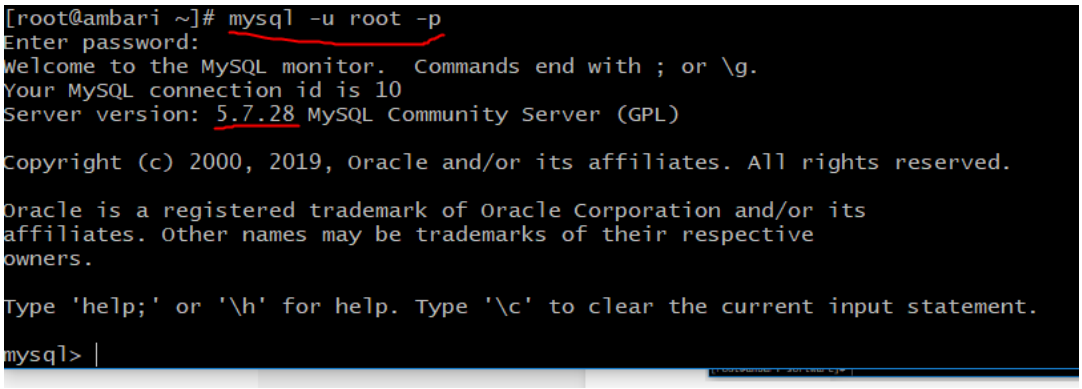


3.2 以 root 用户在 ambari.wanfeng.com 服务器查看并修改 MySQL root 用户密码（ 我的密码：Bee#56915123 ）

```
grep 'temporary password' /var/log/mysqld.log
mysql_secure_installation
```

3.3 以 root 用户ambari.wanfeng.com 服务器 执行 mysql -u root -p

后尝试登陆，如下图



3.4 我们可以通过执行如下语句，降低密码要求

SHOW VARIABLES LIKE 'validate_password%';

set global validate_password_policy=0;

set global validate_password_mixed_case_count=0;

set global validate_password_number_count=3;

set global validate_password_special_char_count=0;

set global validate_password_length=3;

SHOW VARIABLES LIKE 'validate_password%';

4. 创建元数据相关库

4.1 创建 ambari 数据库，用于存放 Ambari 相关元数据。在 ambari.wanfeng.com 服务器，用 mysql -u root -p 登陆MySQL数据库实例，并执行如下命令

drop database if exists ambari;

CREATE DATABASE `ambari` /*!40100 DEFAULT CHARACTER SET utf8 */;

drop user 'ambari'@'ambari.wanfeng.com';

GRANT ALL PRIVILEGES ON ambari.* TO 'ambari'@'ambari.wanfeng.com' IDENTIFIED BY'Ambari_123' WITH GRANT OPTION;

GRANT ALL PRIVILEGES ON ambari.* TO 'ambari'@'192.168.11.18' IDENTIFIED BY'Ambari_123' WITH GRANT OPTION;

flush privileges;

show create database ambari;

4.2 创建 hive 数据库，用于存放 Hive 相关元数据。在 ambari.wanfeng.com 服务器，用 mysql -u root -p 登陆MySQL数据库实例，并执行如下命令

```
drop database if exists hive;
CREATE DATABASE `hive` /*!40100 DEFAULT CHARACTER SET utf8 */;
drop user  'hive'@'master-v01.wanfeng.com';
drop user  'hive'@'master-v02.wanfeng.com';
GRANT ALL PRIVILEGES ON hive.* TO 'hive'@'master-v01.wanfeng.com'
IDENTIFIED BY 'Wfhive_4321' WITH GRANT OPTION;
GRANT ALL PRIVILEGES ON hive.* TO 'hive'@'master-v02.wanfeng.com'
IDENTIFIED BY 'Wfhive_4321' WITH GRANT OPTION;
flush privileges;
show create database hive;
```

4.3 创建 ranger 数据库，用于存放 Ranger 相关元数据。在 ambari.wanfeng.com 服务器，用 mysql -u root -p 登陆MySQL数据库实例，并执行如下命令

```
drop database if exists `ranger`;
use mysql;
create database `ranger`;
delete from mysql.user where User='rangeradmin';
grant all privileges on ranger.* to 'rangeradmin'@'master-v01.wanfeng.com'
identified by 'Wfrangeradmin_8899' WITH GRANT OPTION;
grant all privileges on ranger.* to 'rangeradmin'@'master-v02.wanfeng.com'
identified by 'Wfrangeradmin_8899' WITH GRANT OPTION;
grant all privileges on ranger.* to 'rangeradmin'@'client-v01.wanfeng.com'
identified by 'Wfrangeradmin_8899' WITH GRANT OPTION;
grant all privileges on ranger.* to 'rangeradmin'@'client-v02.wanfeng.com'
identified by 'Wfrangeradmin_8899' WITH GRANT OPTION;
flush privileges;
```

4.4 创建 rangerkms 数据库，用于存放 RangerKMS 相关元数据。在 ambari.wanfeng.com 服务器，用 mysql -u root -p 登陆MySQL数据

库实例，并执行如下命令

```
drop database if exists `rangerkms`;
use mysql;
create database `rangerkms`;
delete from user where User='rangerkms';
grant all privileges on rangerkms.* to 'rangerkms'@'master-v01.wanfeng.com'
identified by 'Rangerkms_123';
grant all privileges on rangerkms.* to 'rangerkms'@'master-v02.wanfeng.com'
identified by 'Rangerkms_123';
flush privileges;
```

4.5 创建 superset 数据库，用于存放 superset 相关元数据。在 ambari.wanfeng.com 服务器，用 mysql -u root -p 登陆MySQL数据库实例，并执行如下命令

```
drop database if exists `superset`;
use mysql;
CREATE DATABASE `superset` /*!40100 DEFAULT CHARACTER SET utf8 */;
grant all privileges on superset.* to 'superset'@'client-v01.wanfeng.com'
identified by 'wfsuperset456';
grant all privileges on superset.* to 'superset'@'client-v02.wanfeng.com'
identified by 'wfsuperset456';
flush privileges;
```

5. 配置 ambari.wanfeng.com 服务器能无密码访问其他服务器，并同步各节点的 /etc/hosts 文件

5.1 以 root 用户在 ambari.wanfeng.com 服务器上执行 ssh-keygen -t rsa **命令，一路回车就OK，如下图**

```
[root@ambari ~]# ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
Created directory '/root/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa.
Your public key has been saved in /root/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:WyHJ3QmNmZzLVaO+W9SnY04aQW6eV7OKdY6tyjcl8As root@ambari.wanfeng.com
The key's randomart image is:
+---[RSA 2048]----+
|          ..* ... |
|        . o*o+.  .|
|         +.o=o  .o|
|          .oo+...*|
|         S .o++=+o|
|          o E++O=.|
|         .   oB*= |
|           . .*o  |
|            oo .  |
+----[SHA256]-----+
[root@ambari ~]#
```

5.2 配置到其他机器的无密码访问，在 ambari.wanfeng.com 服务器执行类似如下命令(操作前先将集群所有节点信息添加到 /etc/hosts 文件)

ssh-copy-id -i ~/.ssh/id_rsa.pub master-v01.wanfeng.com

ssh-copy-id -i ~/.ssh/id_rsa.pub master-v02.wanfeng.com

...

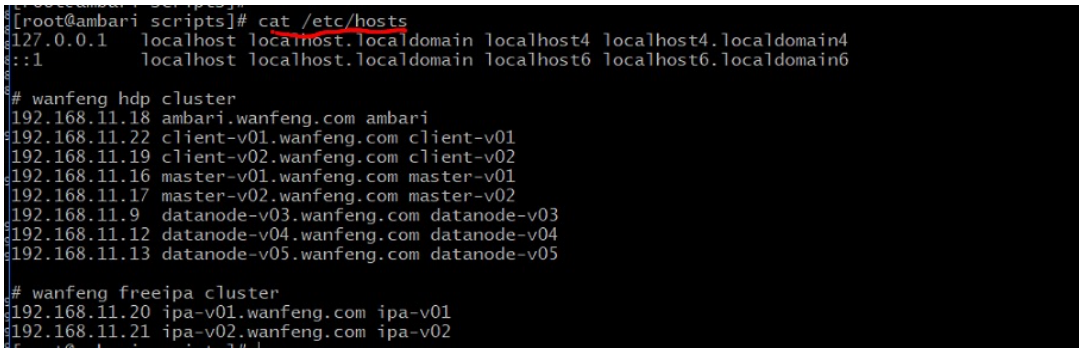
注意1： 所有需要添加到 Ambari 来管理的服务器，均需要配置无密码访问。

执行 ssh-copy-id -i ~/.ssh/id_rsa.pub 命令(类似如下)，执行后，可以用 ssh root@$host 来验证是否可以无密码远程登陆了。具体操作类似如下图。

```
[root@ambari ~]# ssh-copy-id -i ~/.ssh/id_rsa.pub master-v01.wanfeng.com
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/root/.ssh/id_rsa.pub"
The authenticity of host 'master-v01.wanfeng.com (192.168.11.16)' can't be established.
ECDSA key fingerprint is SHA256:WYH4npVIjdgCDOtB4v46qHti2hFcLPVrDk+oVidCv/Q.
ECDSA key fingerprint is MD5:49:b2:1a:25:61:33:df:be:81:c8:e9:5b:da:5b:06:13.
Are you sure you want to continue connecting (yes/no)? yes
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it is to install the new keys
root@master-v01.wanfeng.com's password:

Number of key(s) added: 1

Now try logging into the machine, with:   "ssh 'master-v01.wanfeng.com'"
and check to make sure that only the key(s) you wanted were added.

[root@ambari ~]# ssh master-v01.wanfeng.com
Last login: Tue Oct 29 14:51:02 2019 from 192.168.11.11
[root@master-v01 ~]# exit
logout
Connection to master-v01.wanfeng.com closed.
[root@ambari ~]#
```

注意2: 安全起见，两台 ipa 机器不应该加到 ambari 集群，所以也就不应该配置 ambari.wanfeng.com 服务器无密码访问

5.3 在所有节点添加集群主机信息到 /etc/hosts。（为了方便操作，可以先在 ambari.wanfeng.com 上修改 /etc/hosts 文件，然后同步到其他节点），最后所有节点的 /etc/hosts文件的内容类似如下图



咱们可以先编写两个脚本：一个用于批量同步文件，一个用于批量执行命令。

mkdir ~/scripts

5.1.1 ~/scripts/ssh_to_all_node.sh 文件内容如下(注意：一定要小心调用该脚本，一个误操作，可能就是删服务器走人的节凑哦！）

cat ~/scripts/ssh_to_all_node.sh

```
#!/bin/bash

if [ $# -ne 1 ]
then
  echo "Must one parameter for sshCommand"
  exit 1
else
  cmd=$1
fi

echo $source_file
echo $target_dir

local_host=$(hostname -f)
```

```
echo $local_host
echo "------------------------------------------------------------"
echo "Begin run command: [ ${cmd} ] to AllCusterNode, Please waitting..."
echo ""

for target_host in `cat /etc/hosts |grep 'wanfeng.com'|grep -v 'ipa'|grep -v 'grep'|awk '{print $2}'`
do
  if [ "${target_host}" != "${local_host}" ]; then
    echo "Run Command:  ssh root@${target_host} ${cmd}"
    sleep 1
    ssh root@${target_host} "${cmd}"
  fi
done
```

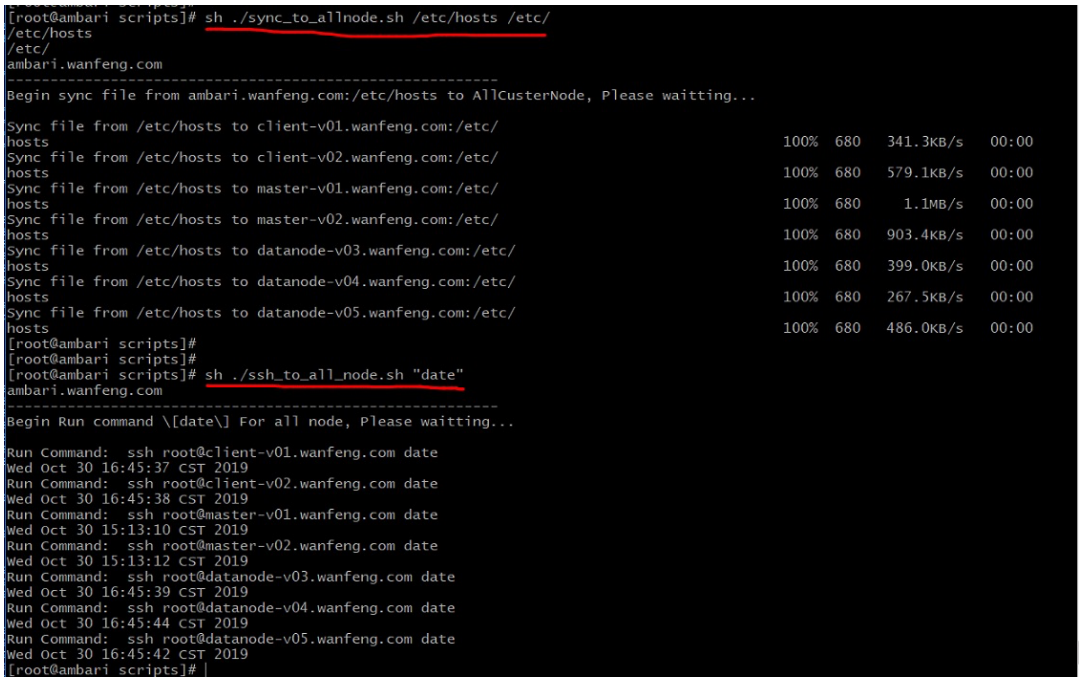
5.1.2 ~/scripts/sync_to_allnode.sh 文件内容如下

cat ~/scripts/sync_to_allnode.sh

```
#!/bin/bash

if [ $# -ne 2 ]
then
   echo "Must two parameter: 1. source file, 2. TargetDir"
   exit 1
else
   source_file=$1
   target_dir=$2
fi

echo $source_file
echo $target_dir

local_host=$(hostname -f)
echo $local_host
```

```
echo "------------------------------------------------------------"
echo "Begin sync file from ${local_host}:${source_file} to AllCusterNode, Please waitting..."
echo ""

for target_host in `cat /etc/hosts |grep 'wanfeng.com'|grep -v 'ipa'|grep -v 'grep'|awk '{print $2}'`
do
  if [ "${target_host}" != "${local_host}" ]; then
    echo "Sync file from ${source_file} to ${target_host}:${target_dir}"
    sleep 1
    scp -r ${source_file} root@${target_host}:${target_dir}/
  fi
done
```

5.1.3 执行 ~/scripts/sync_to_allnode.sh /etc/hosts /etc/ ， 运行效果类似如下图



6. 配置时钟同步(参

考：https://www.cnblogs.com/lei0213/p/8723106.html 不赘述）

yum -y install chrony

systemctl enable chronyd

systemctl start chronyd

7. 配置 Maximum Open Files Requirements

7.1 在 ambari.wanfeng.com 服务器 vi /etc/security/limits.conf 文件，添加如下内容

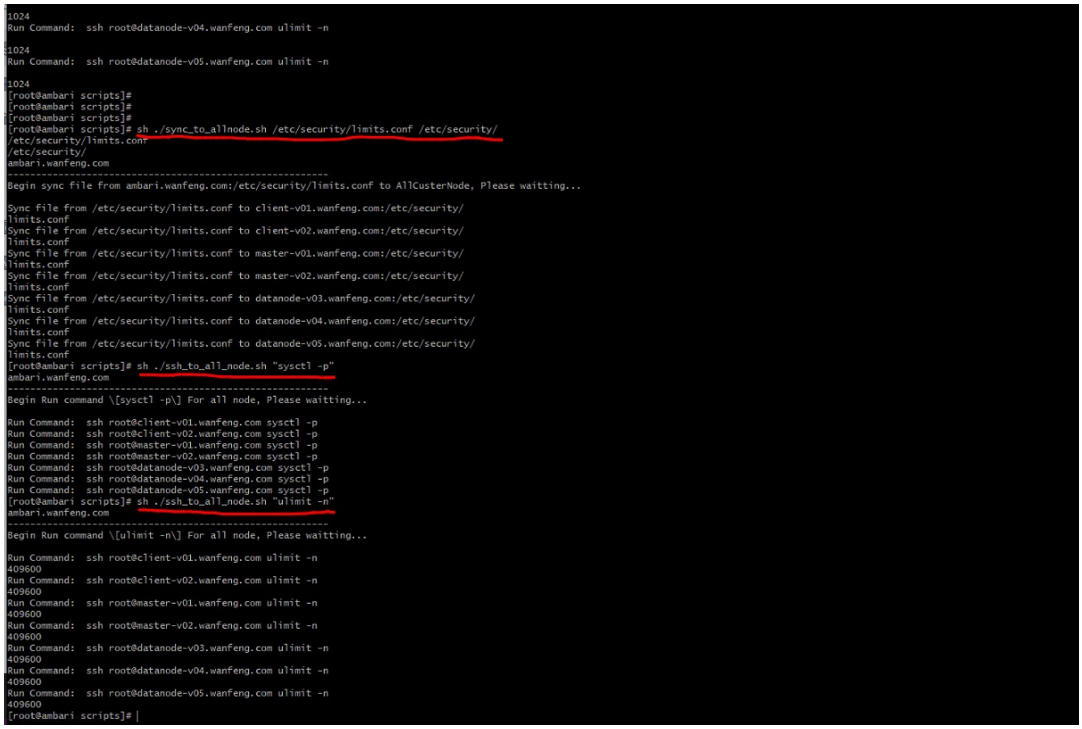
```
*      -      nofile      409600
*      -      nproc       63535
root   -      nofile      409600
root   -      nproc       63535
```

执行 sysctl -p 命令，使上面修改生效（注意：需要新开登陆会话，才能看到效果）

7.2 将 /etc/security/limits.conf 文件同步到其他节点

```
sh ./sync_to_allnode.sh /etc/security/limits.conf /etc/security/
sh ./ssh_to_all_node.sh "sysctl -p"
sh ./ssh_to_all_node.sh "ulimit -n"
```

具体操作效果如下图：



8. 在 ambari.wanfeng.com 服务器配置 yum 本地源

8.1 Disable SELinux and PackageKit and check the umask Value

用 vi /etc/selinux/config 命令编辑 /etc/selinux/config 文件，修改为 SELINUX=disabled

SELINUX=disabled

其他节点批量修改命令如下：

sh ./ssh_to_all_node.sh 'sed -i "s/SELINUX=enforcing/SELINUX=disabled/g" `grep SELINUX=enforcing -rl /etc/selinux/config`'

8.2 安装相关 rpm 包

yum -y install yum-utils createrepo yum-plugin-priorities

cd ~/scripts/;

sh ./ssh_to_all_node.sh "yum -y install yum-utils createrepo yum-plugin-priorities;"

8.2 安装 httpd，以 root 用户在 ambari.wanfeng.com 服务器执行如

下命令

```
yum -y install yum-utils createrepo
yum -y install httpd
mkdir -p /var/www/html/
systemctl enable httpd
systemctl start httpd
systemctl status httpd
```

8.3 解压安装包，以 root 用户在 ambari.wanfeng.com 服务器执行如下命令

```
cd /DATA/disk1/software
tar -xvf ambari-2.7.4.0-centos7.tar.gz
tar -xvf HDP-3.1.4.0-centos7-rpm.tar.gz
tar -xvf HDP-GPL-3.1.4.0-centos7-gpl.tar.gz
tar -xvf HDP-UTILS-1.1.0.22-centos7.tar.gz
tar -xvf jdk-8u191-linux-x64.tar.gz
```

8.4 创建软链接，以 root 用户在 ambari.wanfeng.com 服务器执行如下命令

```
cd /var/www/html/
ln -s /DATA/disk1/software/ambari ./ambari
ln -s /DATA/disk1/software/HDP ./HDP
ln -s /DATA/disk1/software/HDP-GPL ./HDP-GPL
ln -s /DATA/disk1/software/HDP-UTILS ./HDP-UTILS
ls -l ./
```

8.5 配置yum相关的 repo 文件，最后每个文件的内容类似如下

8.5.1 ambari.repo 文件

cat /etc/yum.repos.d/ambari.repo

```
#VERSION_NUMBER=2.7.4.0-118
[ambari-2.7.4.0]
```

```
json.url = http://ambari.wanfeng.com/HDP/hdp_urlinfo.json
name=ambari Version - ambari-2.7.4.0
baseurl=http://ambari.wanfeng.com/ambari/centos7/2.7.4.0-118
gpgcheck=1
gpgkey=http://ambari.wanfeng.com/ambari/centos7/2.7.4.0-118/RPM-GPG-KEY/RPM-GPG-KEY-Jenkins
enabled=1
priority=1
```

8.5.2 hdp.repo 文件

cat /etc/yum.repos.d/hdp.repo

```
#VERSION_NUMBER=3.1.4.0-315
[HDP-3.1.4.0]
name=HDP Version - HDP-3.1.4.0
baseurl=http://ambari.wanfeng.com/HDP/centos7/3.1.4.0-315
gpgcheck=1
gpgkey=http://ambari.wanfeng.com/HDP/centos7/3.1.4.0-315/RPM-GPG-KEY/RPM-GPG-KEY-Jenkins
enabled=1
priority=1

[HDP-UTILS-1.1.0.22]
name=HDP-UTILS Version - HDP-UTILS-1.1.0.22
baseurl=http://ambari.wanfeng.com/HDP-UTILS/centos7/1.1.0.22
gpgcheck=1
gpgkey=http://ambari.wanfeng.com/HDP/centos7/3.1.4.0-315/RPM-GPG-KEY/RPM-GPG-KEY-Jenkins
enabled=1
priority=1
```

8.5.3 hdp.gpl.repo 文件

cat /etc/yum.repos.d/hdp.gpl.repo

#VERSION_NUMBER=3.1.4.0-315
[HDP-GPL-3.1.4.0]
name=HDP-GPL Version - HDP-GPL-3.1.4.0
baseurl=http://ambari.wanfeng.com/HDP-GPL/centos7/3.1.4.0-315
gpgcheck=1
gpgkey=http://ambari.wanfeng.com/HDP-GPL/centos7/3.1.4.0-315/RPM-GPG-KEY/RPM-GPG-KEY-Jenkins
enabled=1
priority=1

8.5.4 最后执行如下命令，校验一下。

yum clean all
yum makecache
yum repolist

9. Enable NTP on the Cluster and on the Browser Host

yum install -y ntp;systemctl enable ntpd;
sh ./ssh_to_all_node.sh "yum install -y ntp;systemctl enable ntpd;"

10. Edit the Network Configuration File

ETWORKING=yes
HOSTNAME=<fully.qualified.domain.name>

cd ~/scripts/;
sh ./ssh_to_all_node.sh 'echo "NETWORKING=yes" >> /etc/sysconfig/network;hostname=$(hostname -f);echo "HOSTNAME=${hostname}" >> /etc/sysconfig/network;'
sh ./ssh_to_all_node.sh "cat /etc/sysconfig/network;"

11. Disable iptables

systemctl disable firewalld
service firewalld stop

```
sh ./ssh_to_all_node.sh "systemctl disable firewalld;service firewalld stop;"
```

12. Install MySQL JDBC Driver（只需在 ambari.wanfeng.com 服务器上执行就OK了）

```
yum -y install mysql-connector-java*
```

9. 安装 JDK（只需在 ambari.wanfeng.com 服务器上先安装，然后同步JDK目录到其他服务器）

```
cd /DATA/disk1/software/;
tar -xvf jdk-8u201-linux-x64.tar.gz;
mkdir -p /usr/java/;
mv ./jdk1.8.0_201 /usr/java/;
cd /usr/java/;
ln -s ./jdk1.8.0_201 ./current;
chown -R root:root ./jdk1.8.0_201;
ls -l ./;

cd ~/scripts/;
sh ./ssh_to_all_node.sh "mkdir -p /usr/java/;"
sh ./sync_to_all_node.sh /usr/java/jdk1.8.0_201 /usr/java/;
sh ./ssh_to_all_node.sh "cd /usr/java/;ln -s ./jdk1.8.0_201 ./current;ls -l ./;"

echo "" >> /etc/profile;
echo "#set Java environment" >> /etc/profile;
echo "export JAVA_HOME=/usr/java/latest" >> /etc/profile;
echo "export JRE_HOME=\$JAVA_HOME/jre" >> /etc/profile;
echo "export
CLASSPATH=.:\$JAVA_HOME/lib:\$JRE_HOME/lib:\$CLASSPATH" >>
/etc/profile;
echo "export PATH=\$JAVA_HOME/bin:\$JRE_HOME/bin:\$PATH" >>
/etc/profile;

sh ./sync_to_all_node.sh /etc/profile /etc/;
```

10. 在 ambari.wanfeng.com服务器上安装配置 ambari-server

10.1 安装 ambari-server RPM包，操作如下

yum -y install ambari-server

10.2 配置 ambari-server

ambari-server setup

终端的输出结果类似如下：

[root@ambari scripts]# ambari-server setup
Using python /usr/bin/python
Setup ambari-server
Checking SELinux...
SELinux status is 'disabled'
Customize user account for ambari-server daemon [y/n] (n)? **y**
Enter user account for ambari-server daemon (root):
Adjusting ambari-server permissions and ownership...
Checking firewall status...
Checking JDK...
[1] Oracle JDK 1.8 + Java Cryptography Extension (JCE) Policy Files 8
[2] Custom JDK
==========
==========
Enter choice (1): **2**
WARNING: JDK must be installed on all hosts and JAVA_HOME must be valid on all hosts.
WARNING: JCE Policy files are required for configuring Kerberos security. If you plan to use Kerberos,please make sure JCE Unlimited S trength Jurisdiction Policy Files are valid on all hosts.
Path to JAVA_HOME: /usr/java/current
Validating JDK on Ambari Server...done.

Check JDK version for Ambari Server...
JDK version found: 8
Minimum JDK version is 8 for Ambari. Skipping to setup different JDK for Ambari Server.
Checking GPL software agreement...
GPL License for LZO: https://www.gnu.org/licenses/old-licenses/gpl-2.0.en.html
Enable Ambari Server to download and install GPL Licensed LZO packages [y/n] (n)? **y**
Completing setup...
Configuring database...
Enter advanced database configuration [y/n] (n)? **y**
Configuring database...
==========
==========
Choose one of the following options:
[1] - PostgreSQL (Embedded)
[2] - Oracle
[3] - MySQL / MariaDB
[4] - PostgreSQL
[5] - Microsoft SQL Server (Tech Preview)
[6] - SQL Anywhere
[7] - BDB
==========
==========
Enter choice (1): **3**
Hostname (localhost): ambari.wanfeng.com
Port (3306):
Database name (ambari):
Username (ambari):
Enter Database Password (bigdata):**Ambari_123**
Re-enter password:**Ambari_123**
Configuring ambari database...

Should ambari use existing default jdbc /usr/share/java/mysql-connector-java.jar [y/n] (y)? **y**
Configuring remote database connection properties...
WARNING: Before starting Ambari Server, you must run the following DDL directly from the database shell to create the schema: /var/lib /ambari-server/resources/Ambari-DDL-MySQL-CREATE.sql
Proceed with configuring remote database connection properties [y/n] (y)? **y**
Extracting system views...
ambari-admin-2.7.4.0.118.jar
....
Ambari repo file contains latest json url http://ambari.wanfeng.com/HDP/hdp_urlinfo.json, updating stacks repoinfos with it...
Adjusting ambari-server permissions and ownership...
Ambari Server 'setup' completed successfully.

执行 /ambari-server/resources/Ambari-DDL-MySQL-CREATE.sql 脚本生成 ambari 相关元数据。

mysql -hambari.wanfeng.com -uambari -pAmbari_123 ambari < /var/lib/ambari-server/resources/Ambari-DDL-MySQL-CREATE.sql;

进 MySQL 数据库，验证表数据是否已经生成：

use ambari;
show tables;

10.3 添加 jdbc 驱动到 ambari-server，执行如下命令

ambari-server setup --jdbc-db=mysql --jdbc-driver=/usr/share/java/mysql-connector-java.jar

10.4 启动 ambari-server，执行如下命令

ambari-server start

查看 ambari-server 当前状态的命令如下：

ambari-server status

关闭命令就是

ambari-server stop

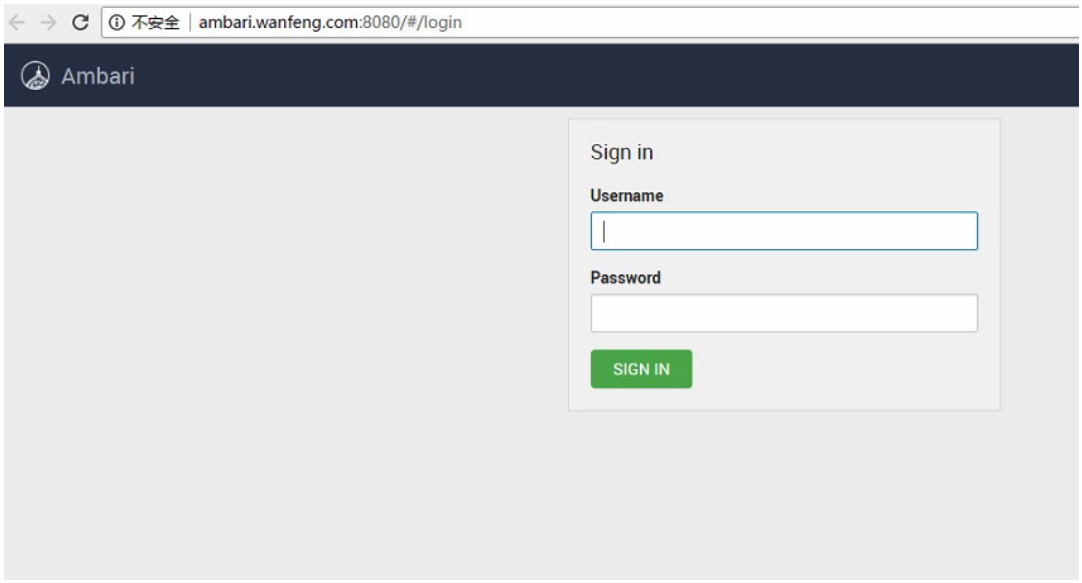
启动成功后，可以执行类似如下命令，去查看日志，有没有报错信息

tail -n 400 /var/log/ambari-server/ambari-server.log

启动成功后，就可以浏览 Ambari 的 web 页面了，链接类似如下(端口默认 8080，也可以通过编辑 /etc/ambari-server/conf/ambari.properties 文件去个性化各参数)

http://ambari.wanfeng.com:8080

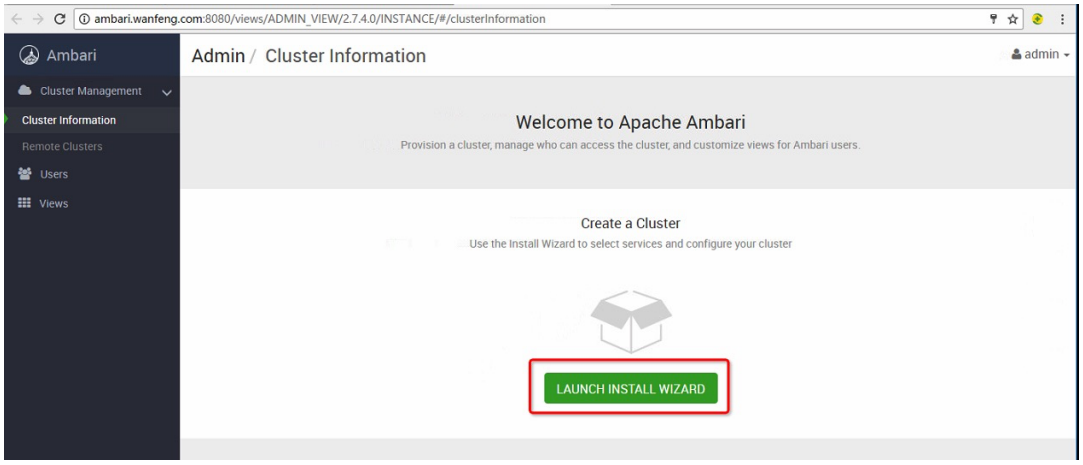
默认的用户名密码均为 **admin**，登陆窗口如下图

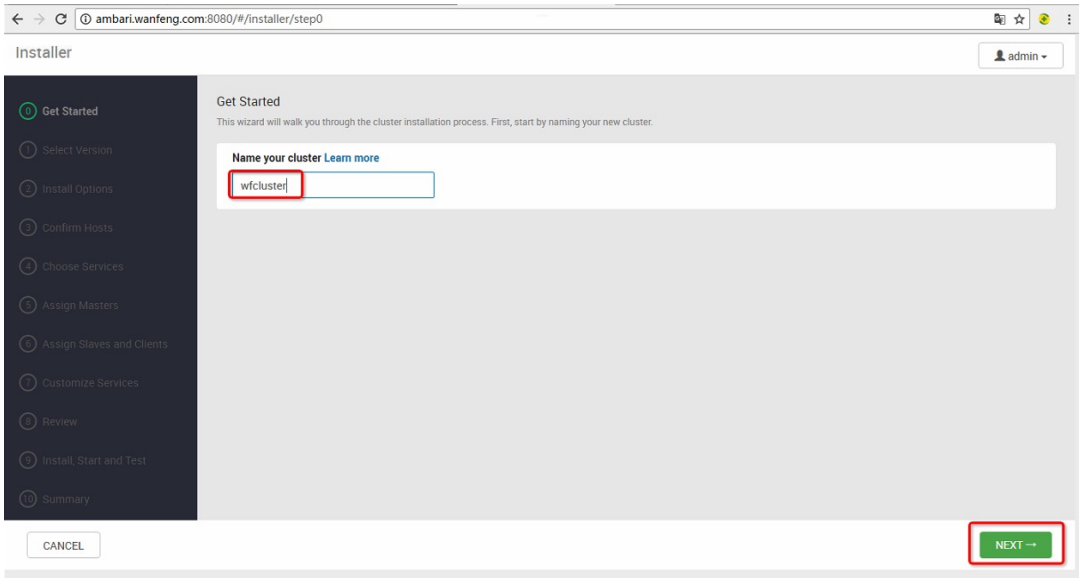


第二章 HDP 3.1.4 高可用集群安装与配置

---

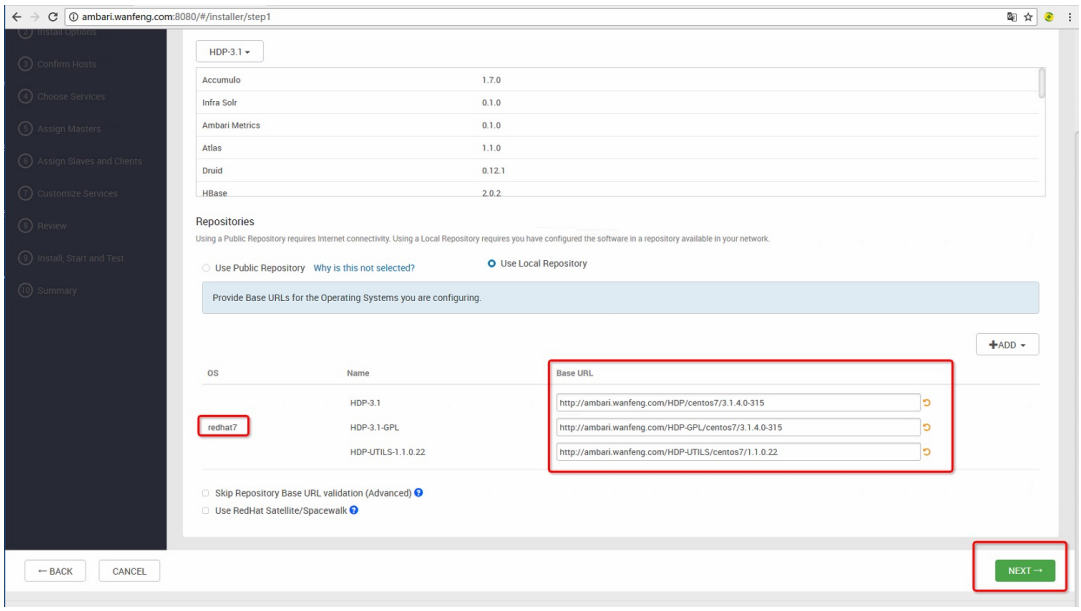
1. 登陆 ambari web 页面，如下图，点击"LAUNCH INSTALL WIZARD"，并给集群命名为 "wfcluster"（万峰集群），并点击"NEXT"进入"Select Version"窗口，如下两图

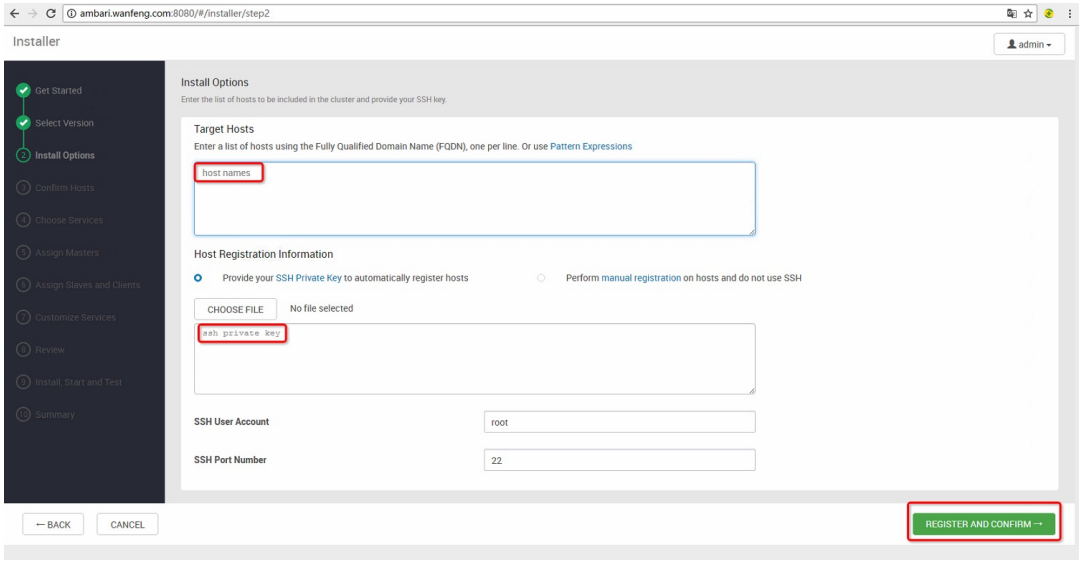




2. 在"Select Version"窗口的 "OS"栏只保留"redhat7"，其他将删除，并配置相关的 Baseurl（相关的 baseurl在 hdp.repo,

hdp.gpl.rep配置文件中均能找到)，并点击 "NEXT"进入"Install Options"窗口，最后类似如下图





3. 在上图中

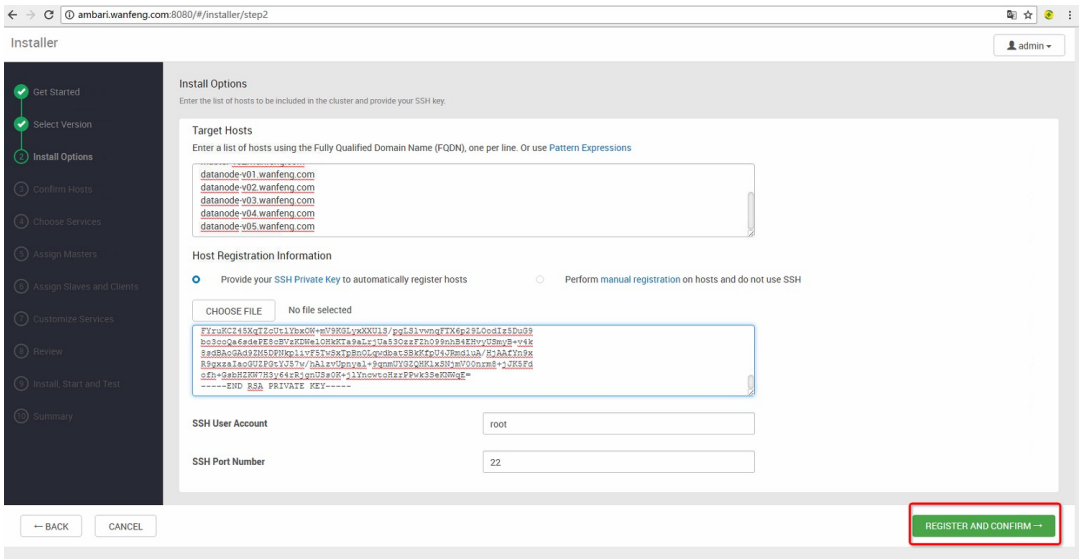
3.1 "host names"文本框中填写集群的节点信息，我填充的内容如下：

client-v01.wanfeng.com

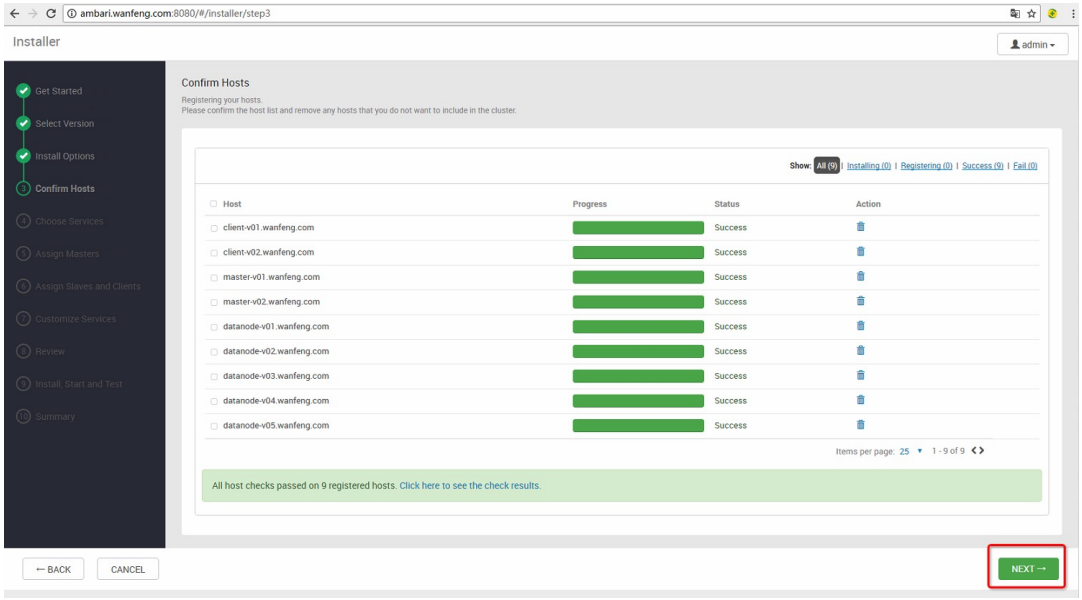
client-v02.wanfeng.com

master-v01.wanfeng.com
master-v02.wanfeng.com
datanode-v01.wanfeng.com
datanode-v02.wanfeng.com
datanode-v03.wanfeng.com
datanode-v04.wanfeng.com
datanode-v05.wanfeng.com

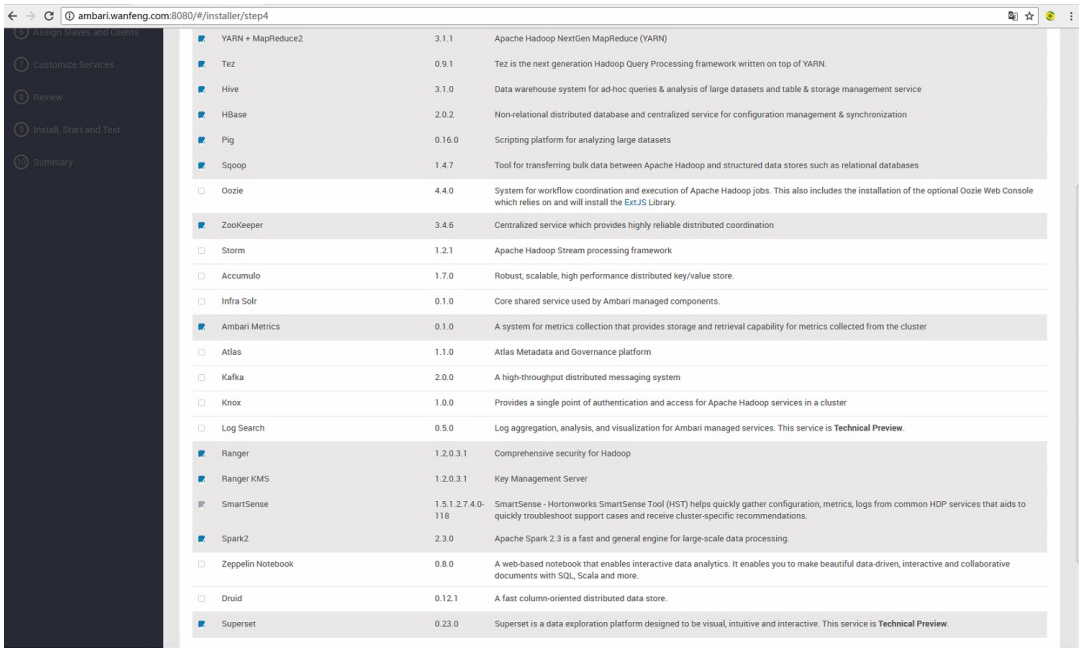
3.2 "ssh private key"文本框中填写ambari.wanfeng.com 服务器的 ~/.ssh/id_rsa 文件的内容。填充后如下图



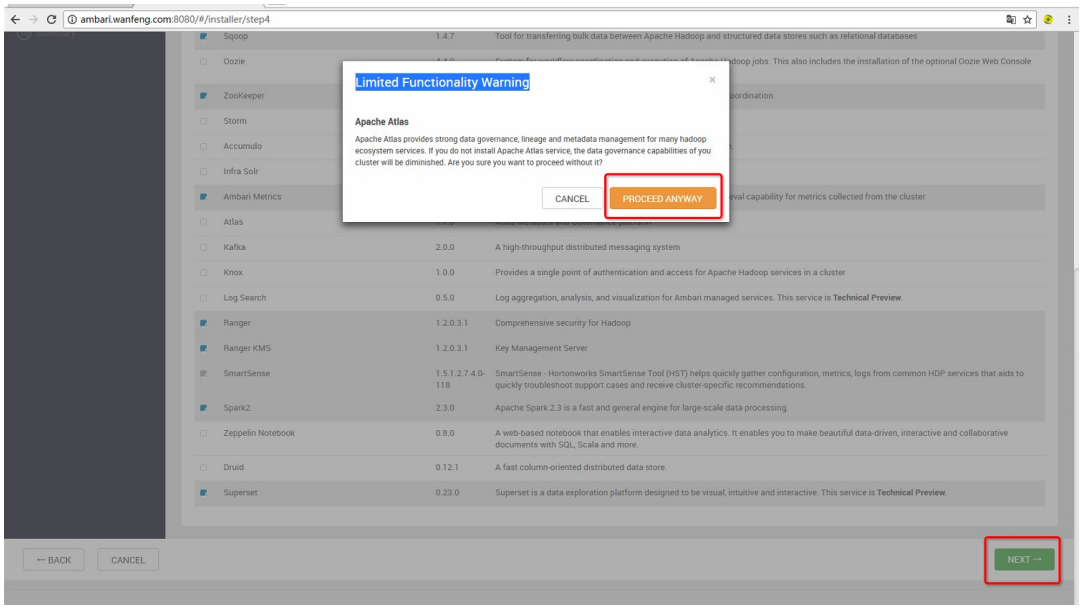
4. 点击上图中的"REGISTER AND CONFIRM"按钮后，进入"Confirm Hosts"窗口(此时需要等待一会儿)，如下图



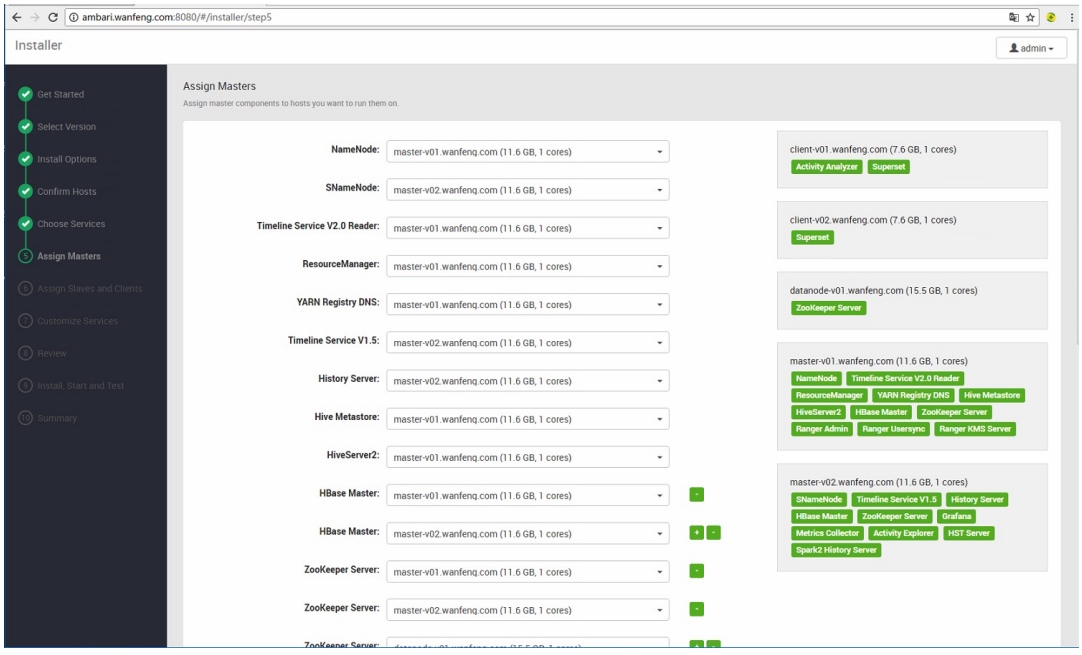
5. 点击上图中的"NEXT"按钮后，进入"Choose File System"窗口，选择我们需要安装的组件，如下图

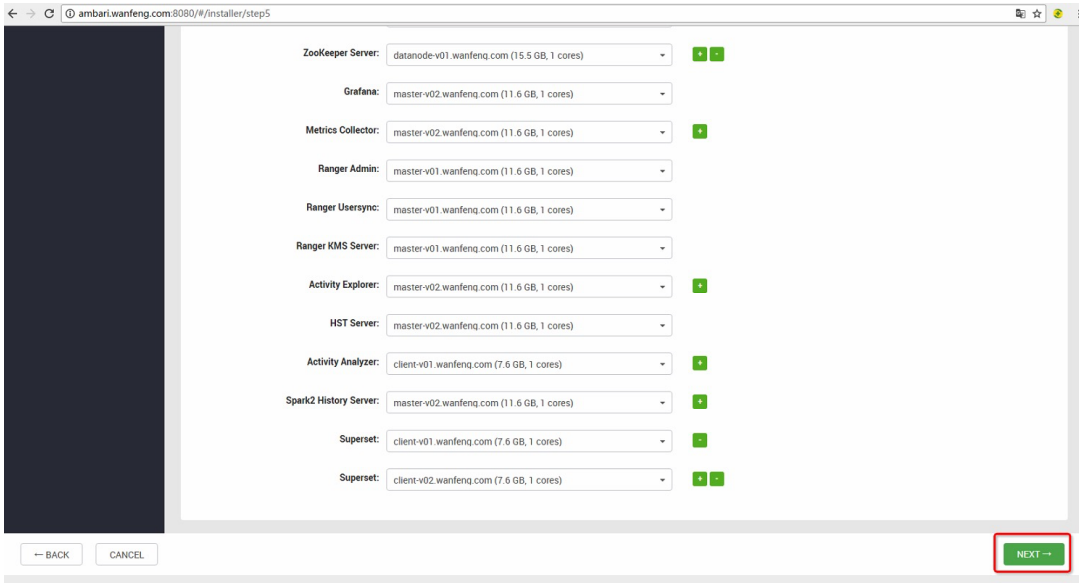


6. 点击上图中的"NEXT"按钮后，弹出两个"Limited Functionality Warning"警告窗口(一个是 "Apache Atlas"，一个是"Infra Solr")，均点击"PROCEED ANYWAY"进入"Assign Masters"窗口

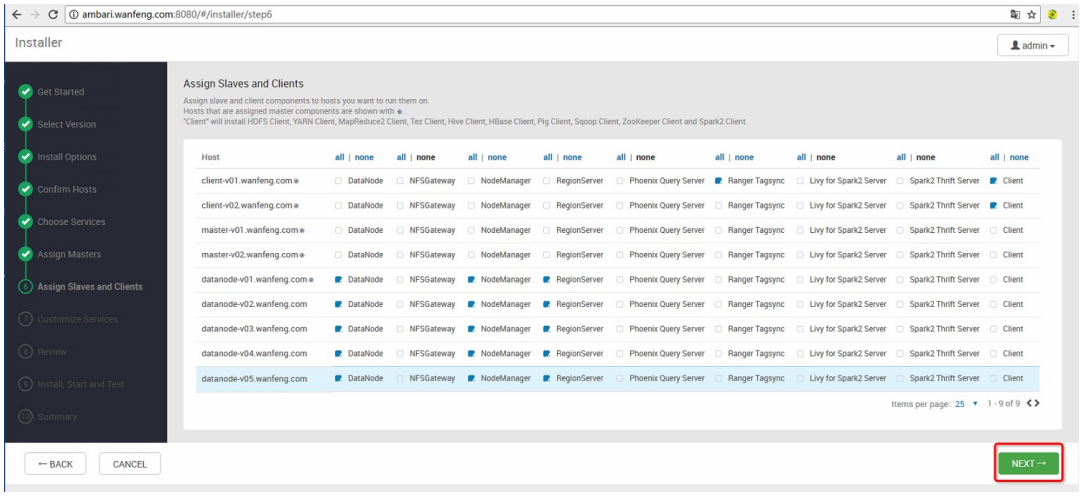


7. 在"Assign Masters"窗口，调整各组件对应的服务器，如下两图(由于一个图已经无法完整显示，故截成两图，均属于Assign Masters窗口)

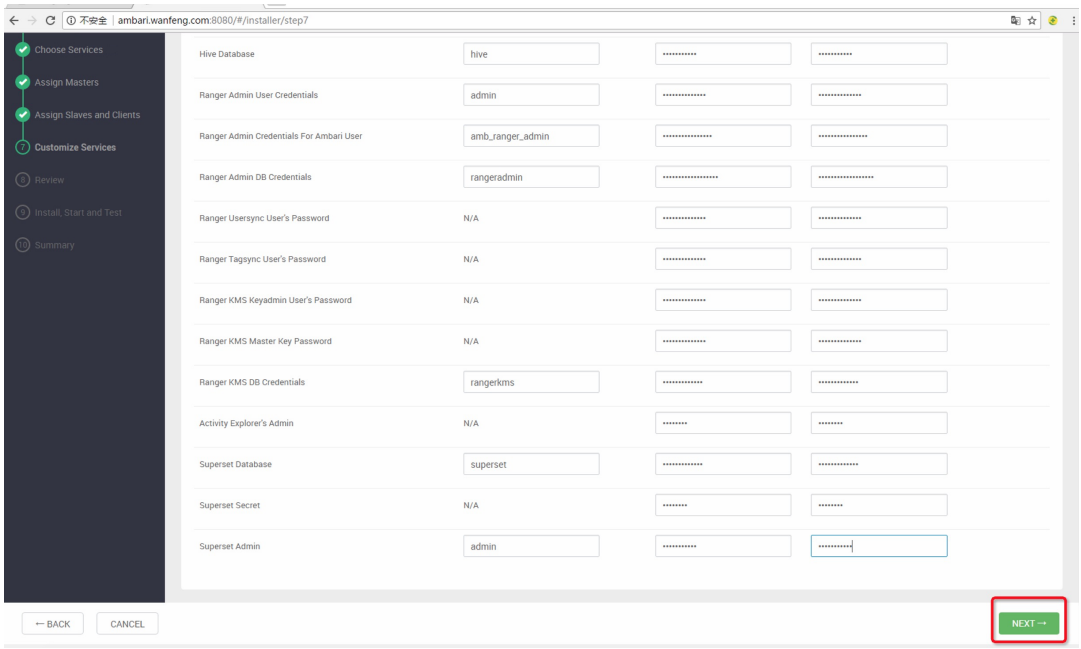




8. 在上图点击"NEXT"按钮进入"Assign Slaves and Clients"窗口，在每个节点选择调整适当的组件后，点击"NEXT"按钮，进入"Customize Services"窗口。



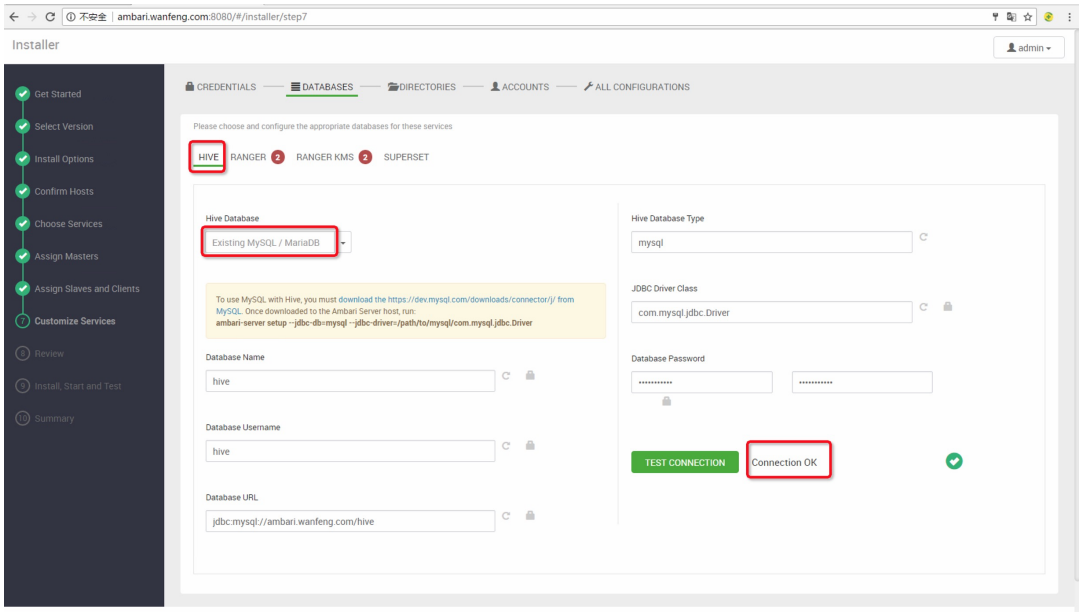
9. 在"Customize Services"窗口，配置每个组件的帐号信息，如下图



将上图中各组件的用户及密码信息保存到 Excel（这是重要的集群信息文件），我的用户及密码信息如下图。

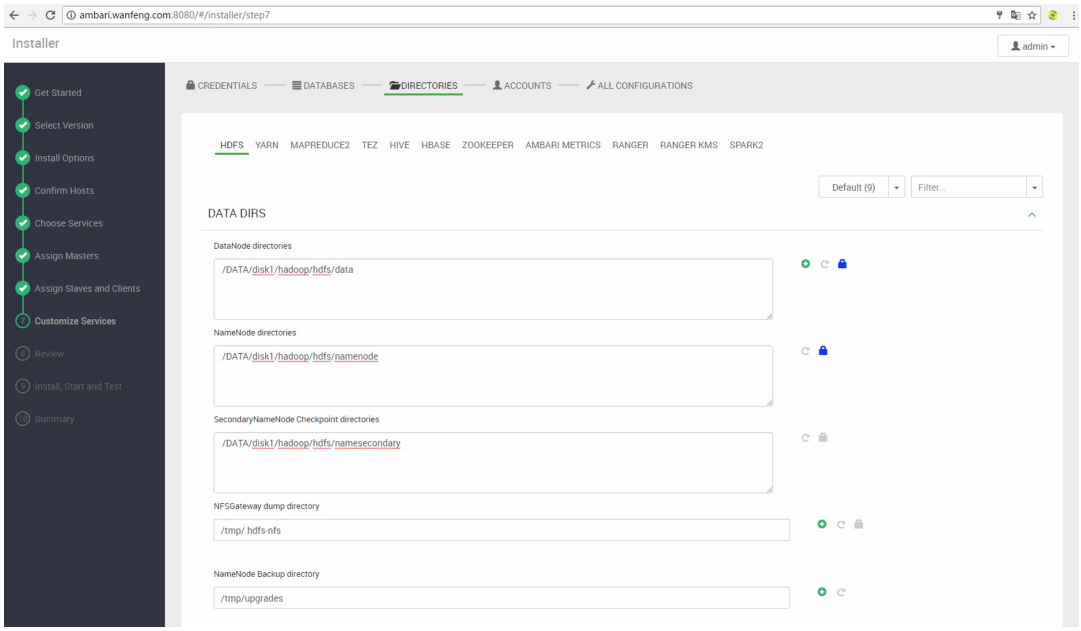
	Username	Password	Confirm Password	
Grafana Admin	admin	wfgfadmin133	wfgfadmin133	
Hive Database	hive	Wfhive_4321	Wfhive_4321	
Ranger Admin User Credentials	admin	rangeradmin888	rangeradmin888	
Ranger Admin Credentials For Ambari User	amb_ranger_admin	……	……	用默认的
Ranger Admin DB Credentials	rangeradmin	Wfrangeradmin_8899	Wfrangeradmin_8899	
Ranger Usersync User's Password	N/A	wfrangeruup987	wfrangeruup987	
Ranger Tagsync User's Password	N/A	wfrangertup155	wfrangertup155	
Ranger KMS Keyadmin User's Password	N/A	wfrangerkms654	wfrangerkms654	
Ranger KMS Master Key Password	N/A	wfrangerkkp123	wfrangerkkp123	
Ranger KMS DB Credentials	rangerkms	Rangerkms_123	Rangerkms_123	
Activity Explorer's Admin	N/A	wfaea445	wfaea445	
Superset Database	superset	wfsuperset456	wfsuperset456	
Superset Secret	N/A	wfss9901	wfss9901	
Superset Admin	admin	wfadmin9854	wfadmin9854	

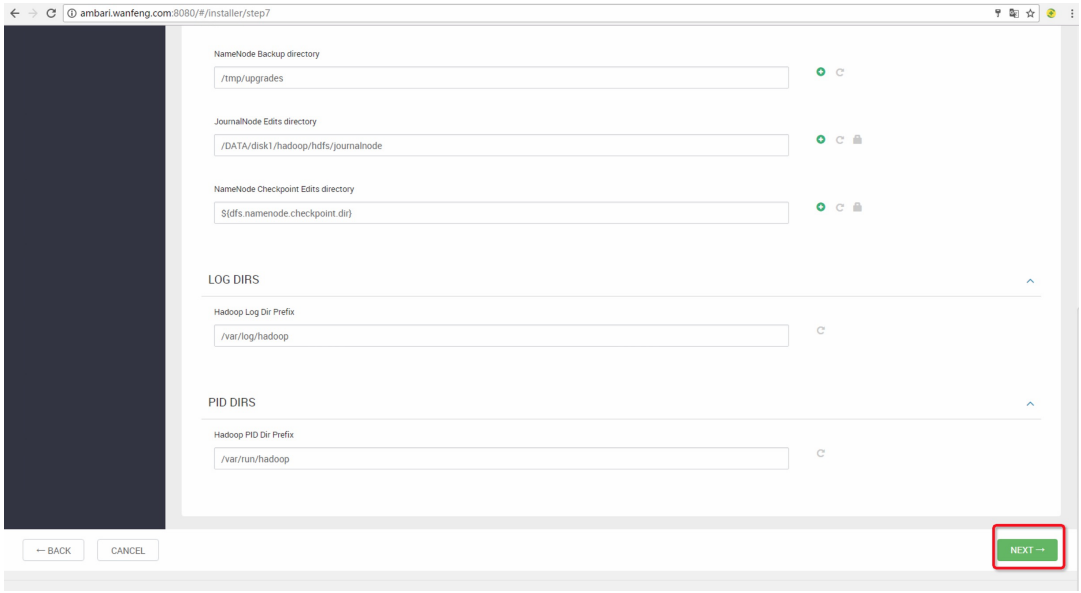
10. 第9步图中点击"NEXT"后进入"Customize Services"窗口，其中我的Hive配置如下图(RANGER, RANGERKMS, SUPERSET等配置信息类似，不再赘述。



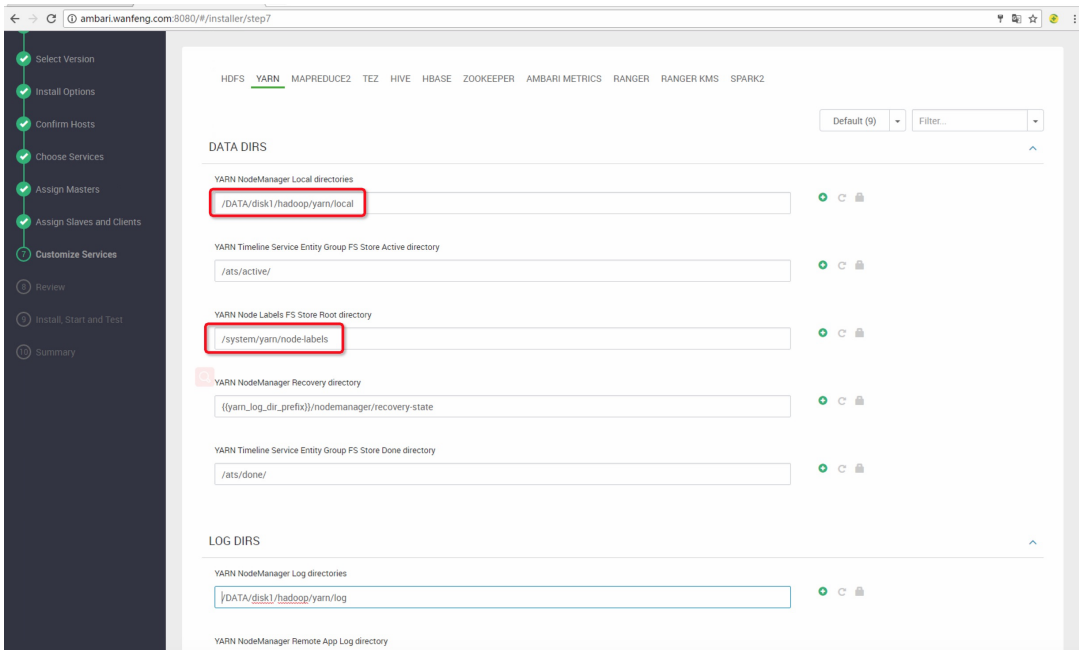
11. 第10步图中点击"NEXT"后进入"DATA DIRS"窗口

11.1 其中我的 HDFS 配置如下两图（两图均为同一窗口）

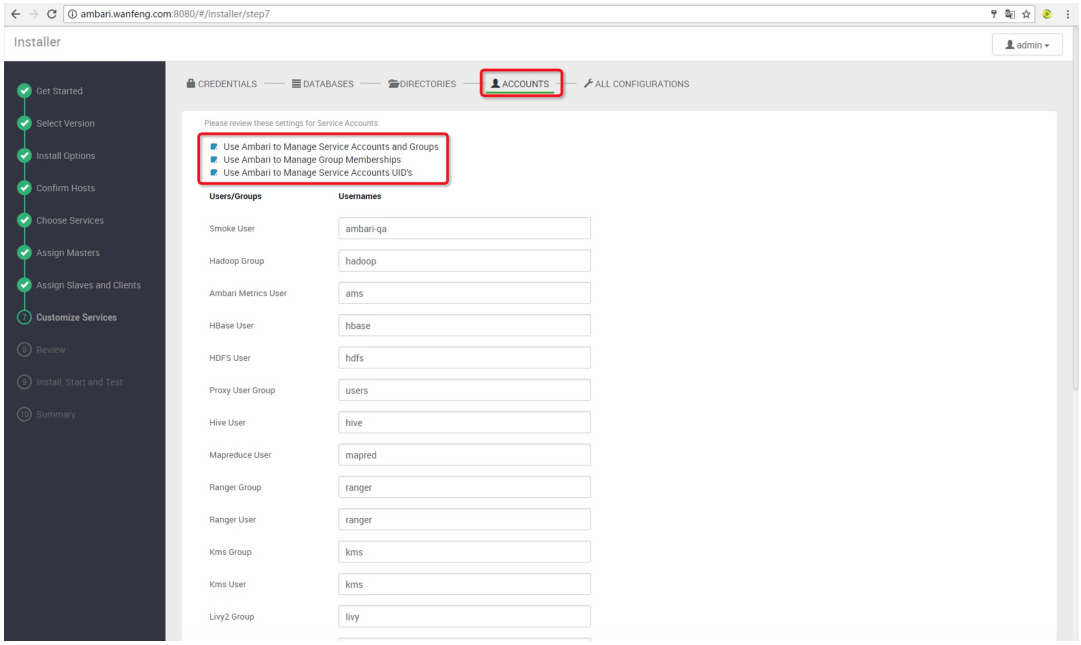




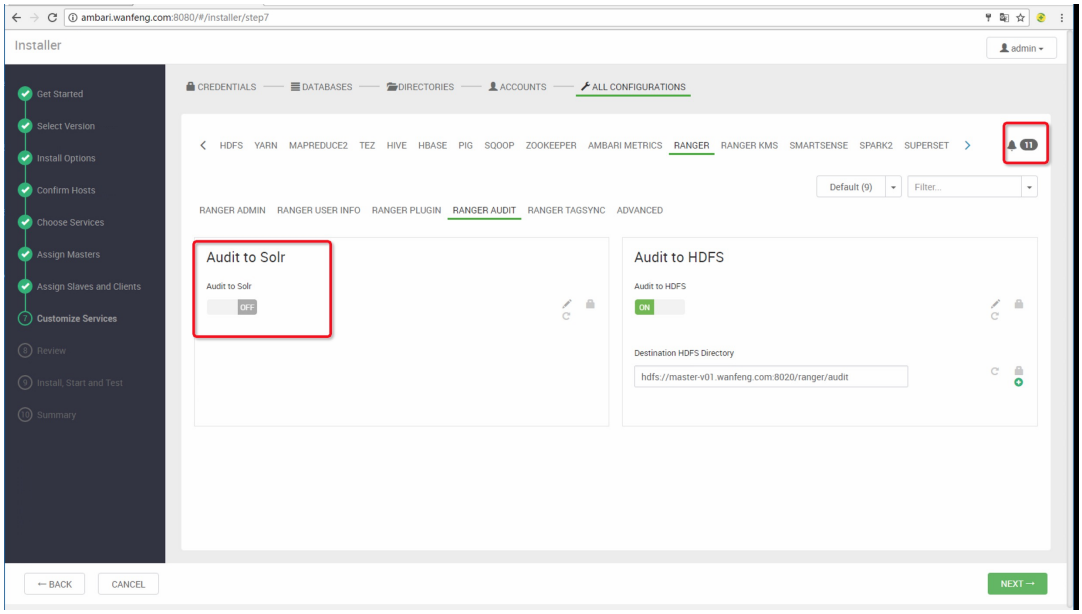
11.2 其中我的 YARN 配置如下图（红框表示我作过修改，由于测试集群，每个节点磁盘只有一块，所以配置相对简单）



12. 点击"NEXT"按钮，进入"ACCOUNTS"配置窗口，一般默认配置就OK，如下图



13. 在上图中，点击"NEXT"进入"ALL CONFIGURATIONS"，发现有配置报警，就修正一下。如下图

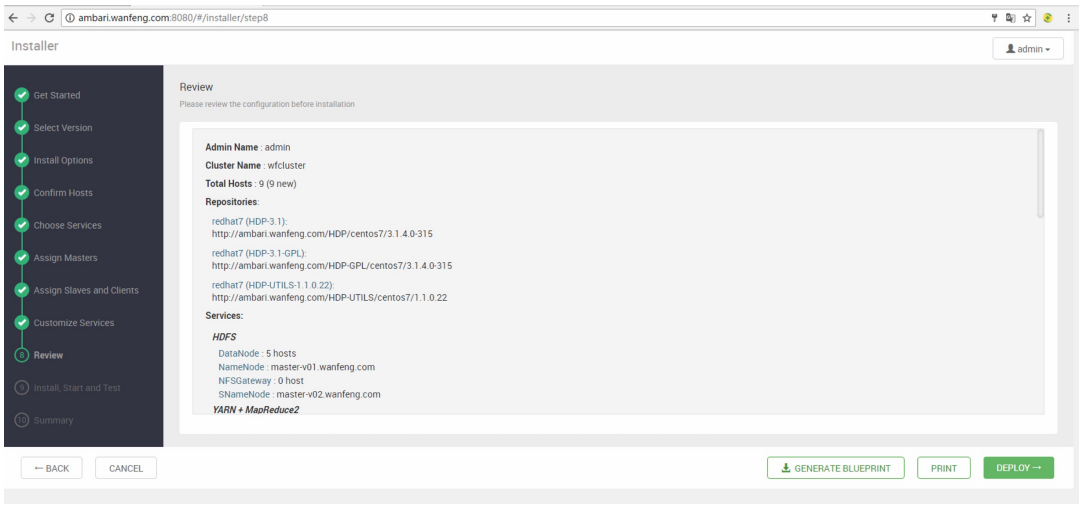


注意： 将"RANGER"栏目中的"Audit to Solr"关闭，没有红色报警的话(当然建议所有报警好好看一下)，应该就可以点击"NEXT"了。

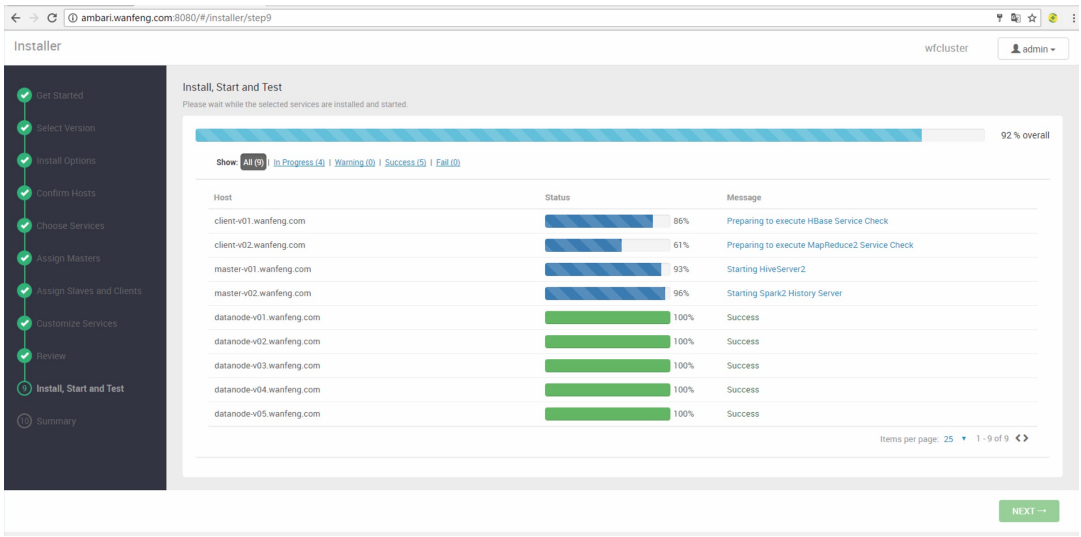
当点击"NEXT"时提示你 CHECK DB的话，就再 TEST CONNECTOIONS 就可以

了。

14. 点击"NEXT"后，如下图，终于可以点击"DEPLOY"开始安装了

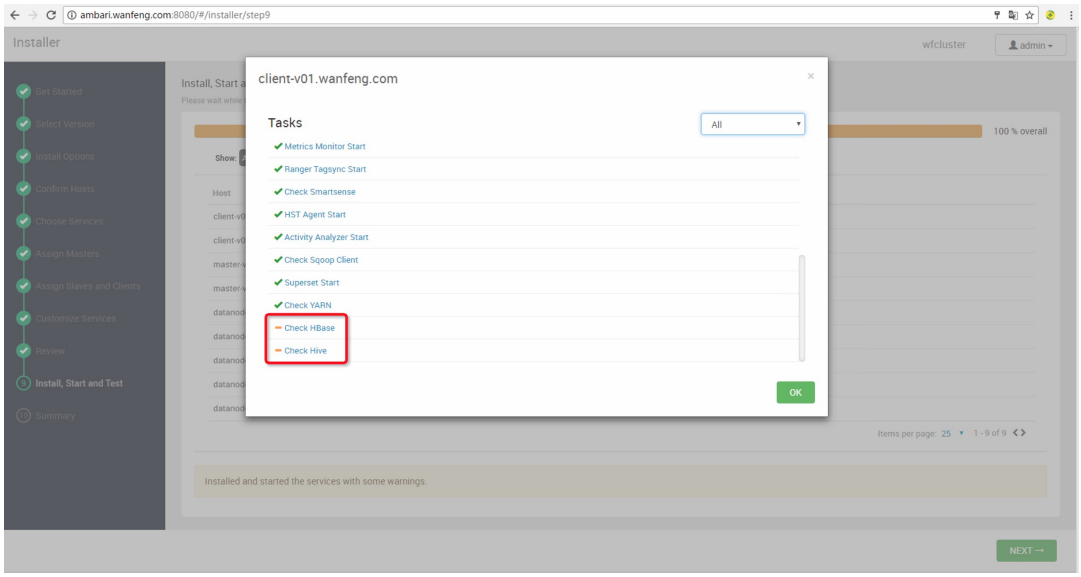


15. 最后安装界面类似如下图，如果安装过程中有报错，请仔细查日志（我是用的虚拟机，有时候可能由于网络异常导致，可以点击"RETRY"按钮重试失败的安装）。

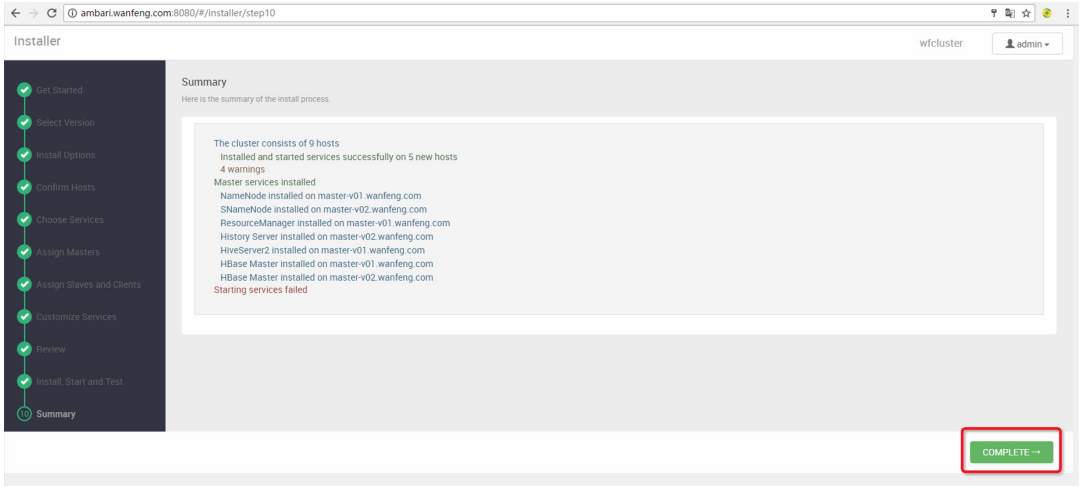


16. 如下图，如果安装完成后，有部分节点有"Warnings encontered"信息，可以点击查看详情，
一些 Check 相关报警可以忽略，此时也看不到"RETRY"按钮

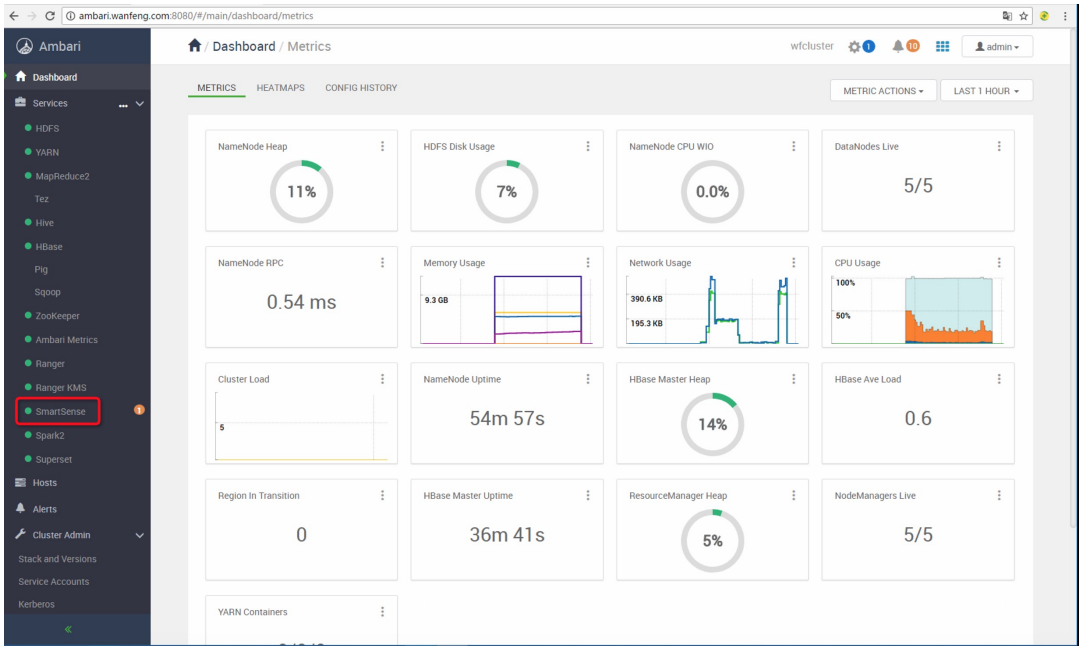
了，只能点击"NEXT"。



17. 如下图，在"Summary"窗口点击"COMPLETE"按钮完成安装(有异常的组件结束后再去单独解决)

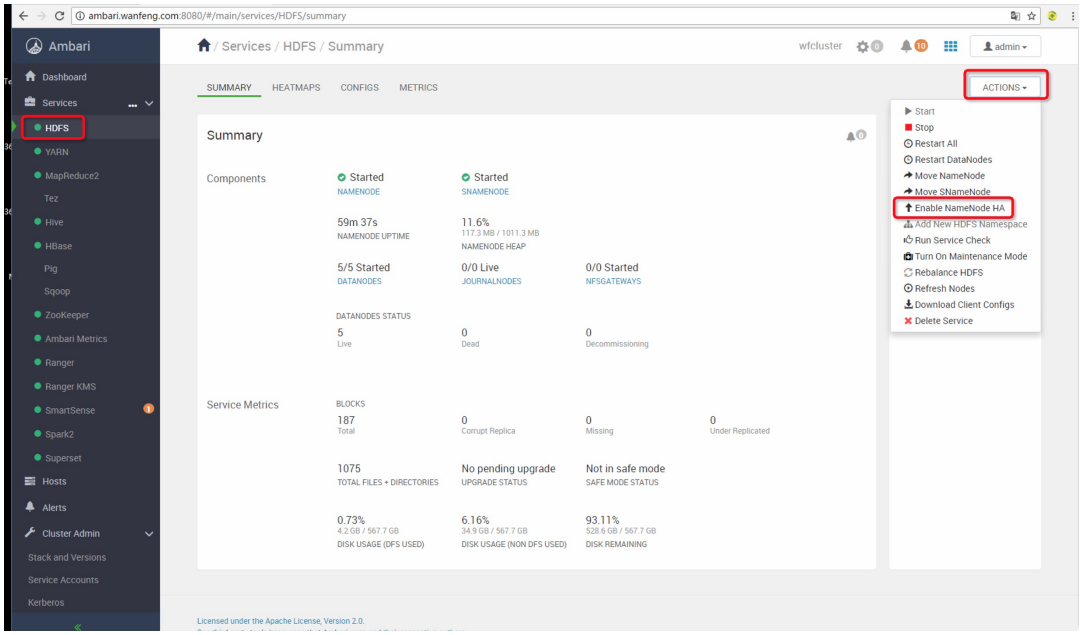


18. 安装完成后的界面如下

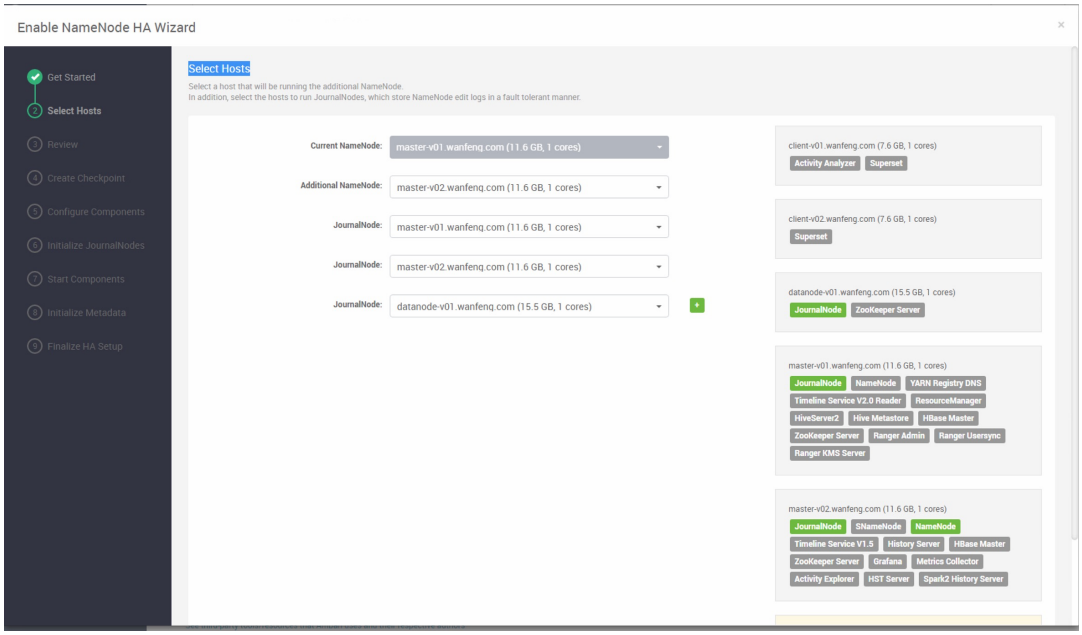


注意： "SmartSense"是Hortonworks 的一个收费组件(主要用于分析各组件的运行情况后，给出合理调优参数的建议等)，如果老异常，我们可以将其删除。

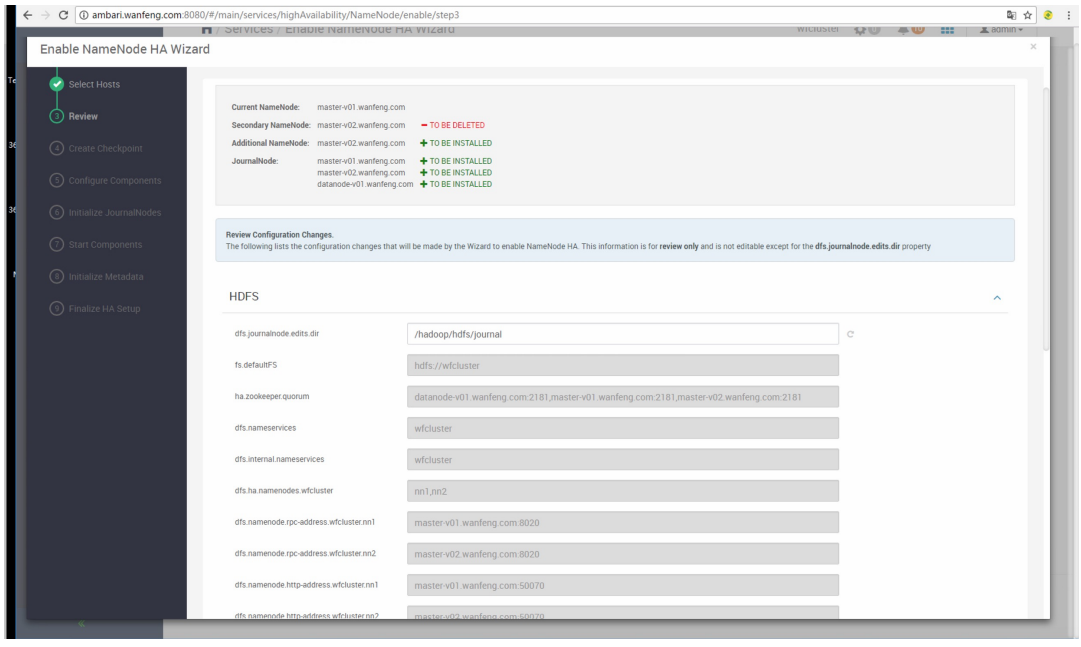
19. HDFS高可用配置，如下两图，在图一我们依次点击"HDFS"->"ACTIONS"->"Enable NameNode HA"，然后进入图二"Enable NameNode HA Wizard"窗口，我们将"Nameservice ID"设置为"wfcluster"，然后点击"NEXT"。



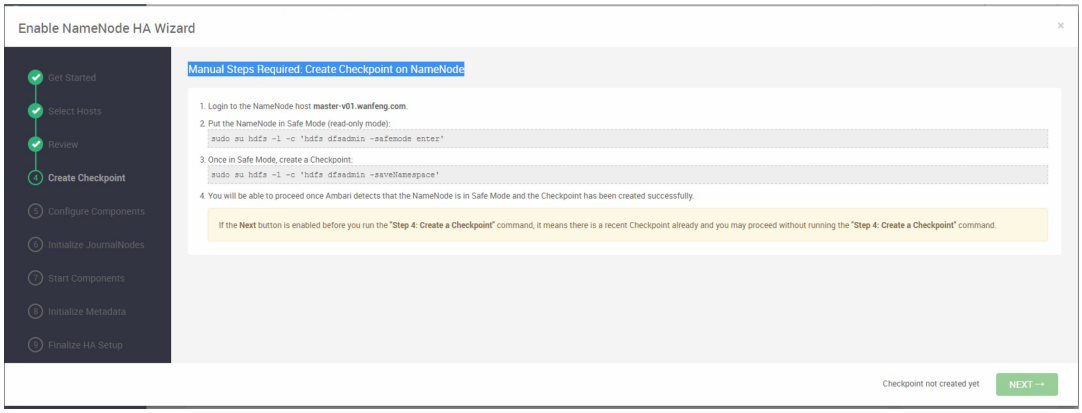
19.1. 在上一步点击"NEXT"后进入"Select Hosts"窗口，如下图，调整各组件安装的服务器信息，点击"NEXT"继续



19.2. 在上一步点击"NEXT"后进入"Review"窗口，如下图，核对没问题后，点击"NEXT"继续



19.3. 在上一步点击"NEXT"进入"Manual Steps Required: Create Checkpoint on NameNode"窗口，如下图



注意：如上图，如果没有根据提示执行相关操作，"NEXT"按钮是无法点击的。

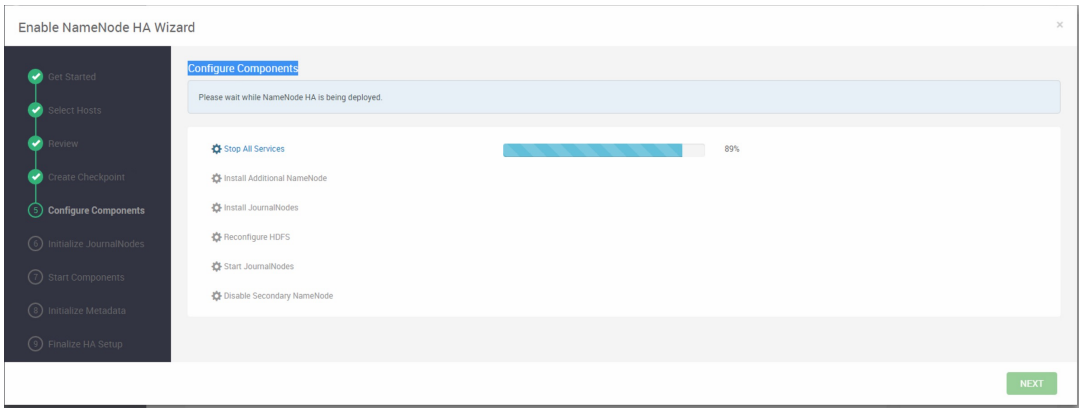
我们先根据提示，进入 master-v01.wanfeng.com 服务器依次执行如下操作(我的操作如下图)

sudo su hdfs -l -c 'hdfs dfsadmin -safemode enter'

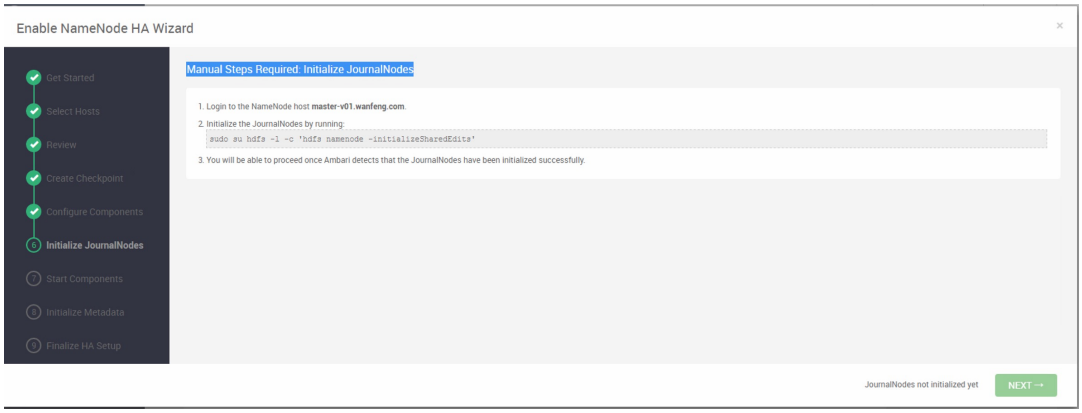
sudo su hdfs -l -c 'hdfs dfsadmin -saveNamespace'

```
[root@master-v01 ~]# sudo su hdfs -l -c 'hdfs dfsadmin -safemode enter'
Safe mode is ON
[root@master-v01 ~]# sudo su hdfs -l -c 'hdfs dfsadmin -saveNamespace'
Save namespace successful
[root@master-v01 ~]#
```

根据提示执行相关操作后，我们就可以在"Manual Steps Required: Create Checkpoint on NameNode"窗口点击"NEXT"进入"Configure Components"窗口，如下图(需要等待一定时间)



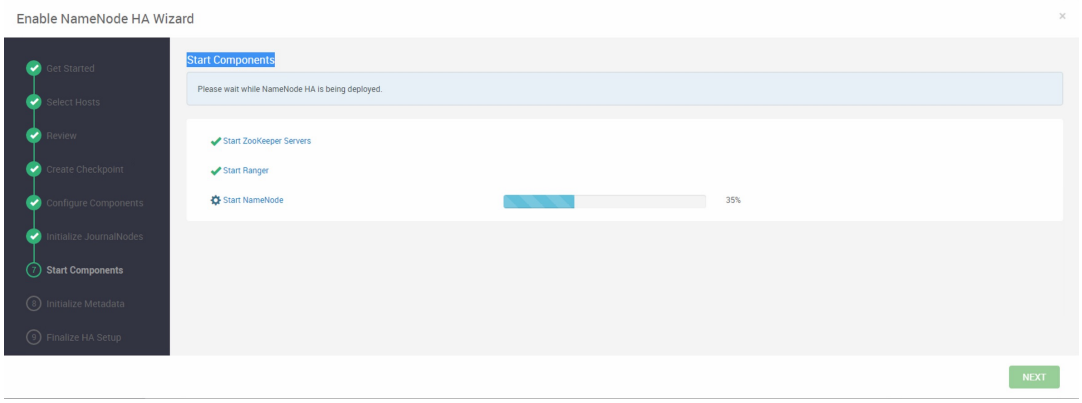
19.4. "Configure Components"窗口执行成功后点击"NEXT"，进入"Manual Steps Required: Initialize JournalNodes"窗口，如下图



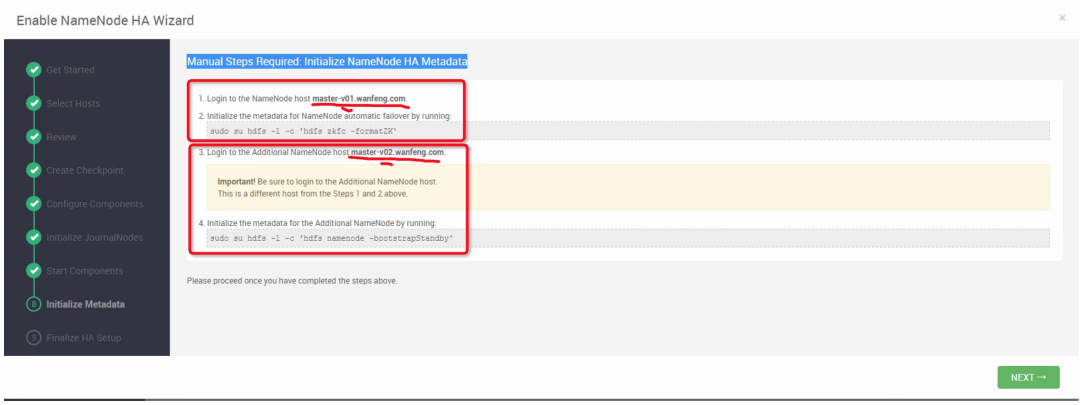
根据提示，进入 master-v01.wanfeng.com 服务器，执行如下操作

sudo su hdfs -l -c 'hdfs namenode -initializeSharedEdits'

操作完成，确认相关日志输出没问题，此时点击"NEXT"进入"Start Components"窗口，如下图



稍等片刻，待操作完成即可点击"NEXT"，进入"Manual Steps Required: Initialize NameNode HA Metadata"窗口，如下图



19.5. 如上图，在"Manual Steps Required: Initialize NameNode HA Metadata"窗口，我们根据提示依次执行：

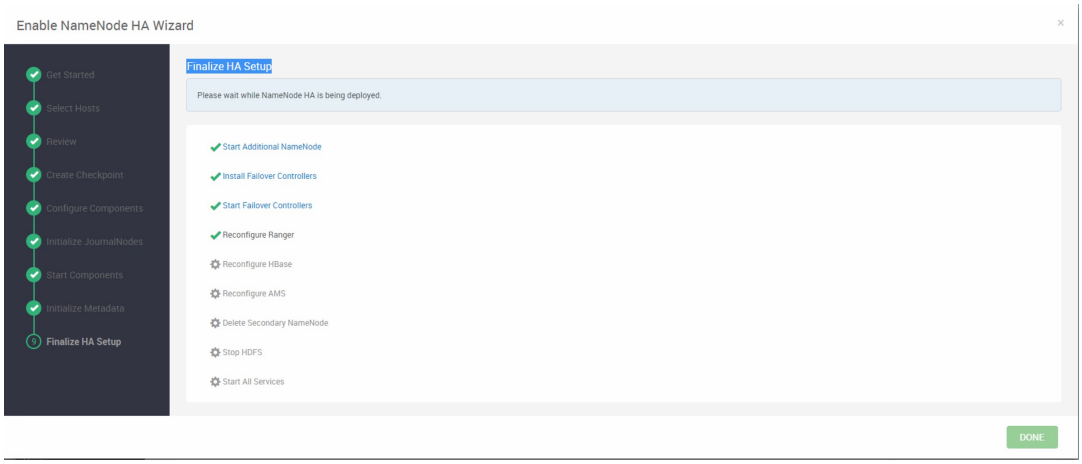
19.5.1 进 master-v01.wanfeng.com 服务器执行如下命令

sudo su hdfs -l -c 'hdfs zkfc -formatZK'

19.5.2 进 master-v02.wanfeng.com 服务器执行如下命令(!!千万注意是 master-v02 哦!!)

sudo su hdfs -l -c 'hdfs namenode -bootstrapStandby'

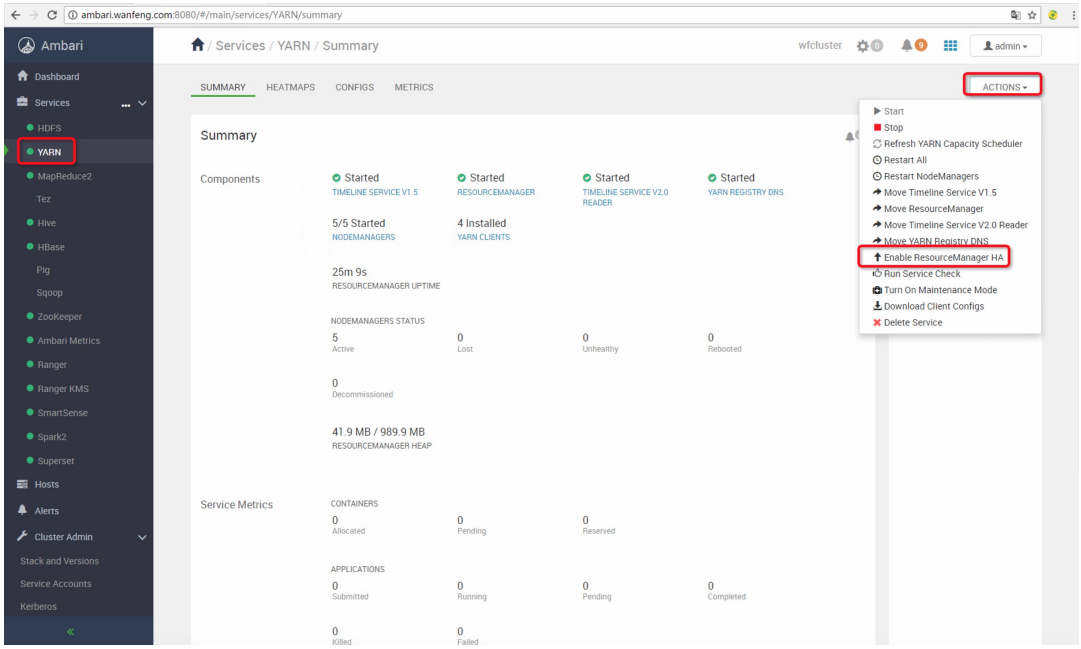
执行完上两步操作后，点击"NEXT"，确认窗口点击"OK"即可进入"Finalize HA Setup"窗口，如下图

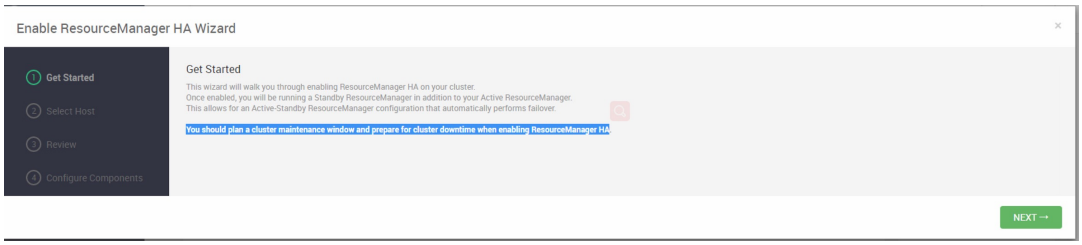


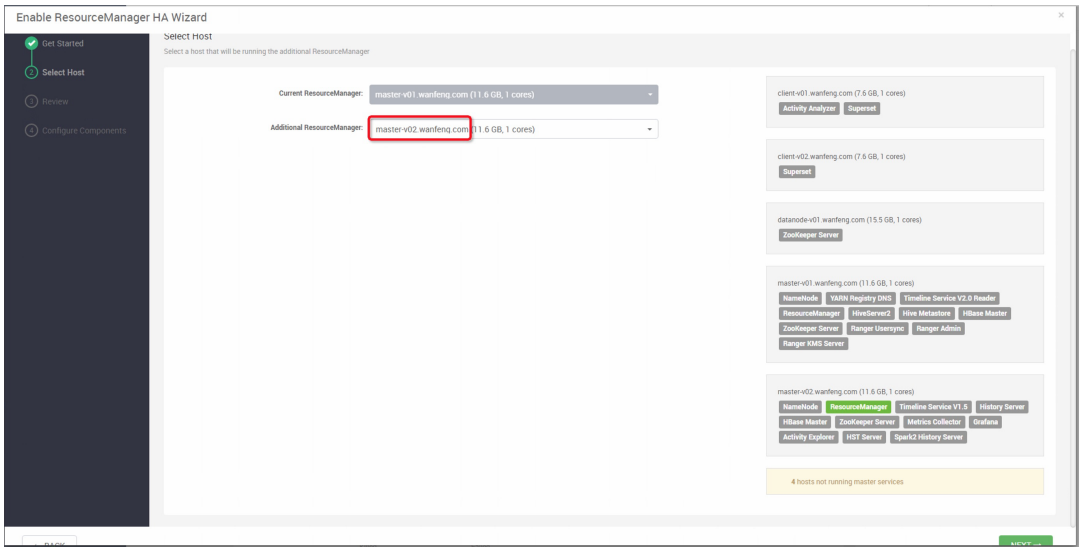
稍等一会儿，点击"NEXT"完成操作。

20. YARN高可用配置

20.1 如下三图，在图一我们依次点击"YARN"->"ACTIONS"->"Enable ResourceManager HA"，然后进入图二"Get Started"窗口，提示你集群有一定的窗口维护期，点击"NEXT"进入图三"Select Host"窗口。

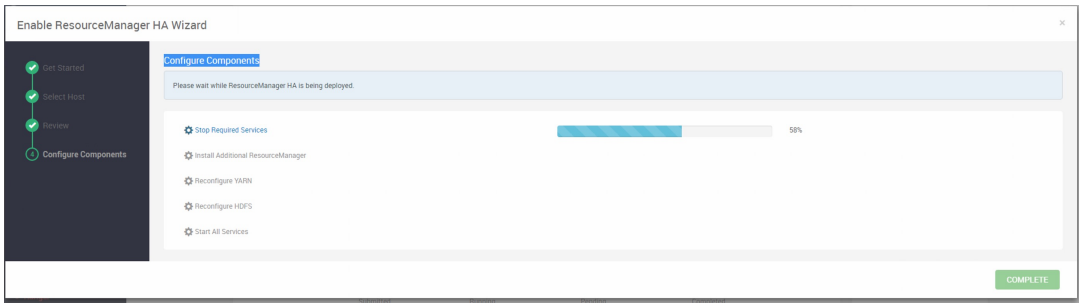






20.2 在"Select Host"窗口，选择合适的服务器后，点击"NEXT"进入"Review"窗口，确认后继续下一步，进入"Configure

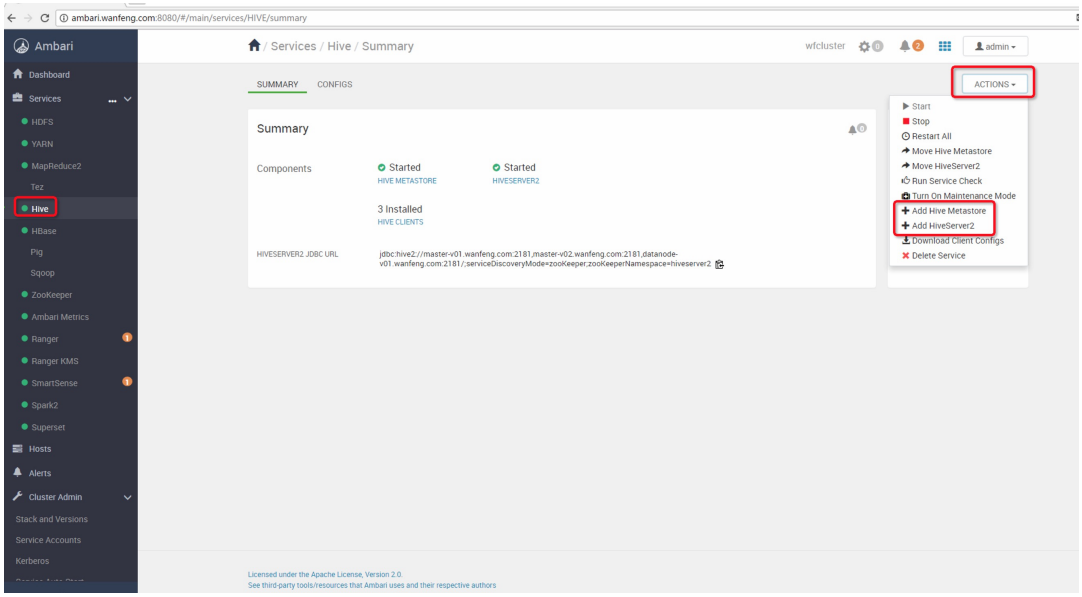
Components"窗口，如下图



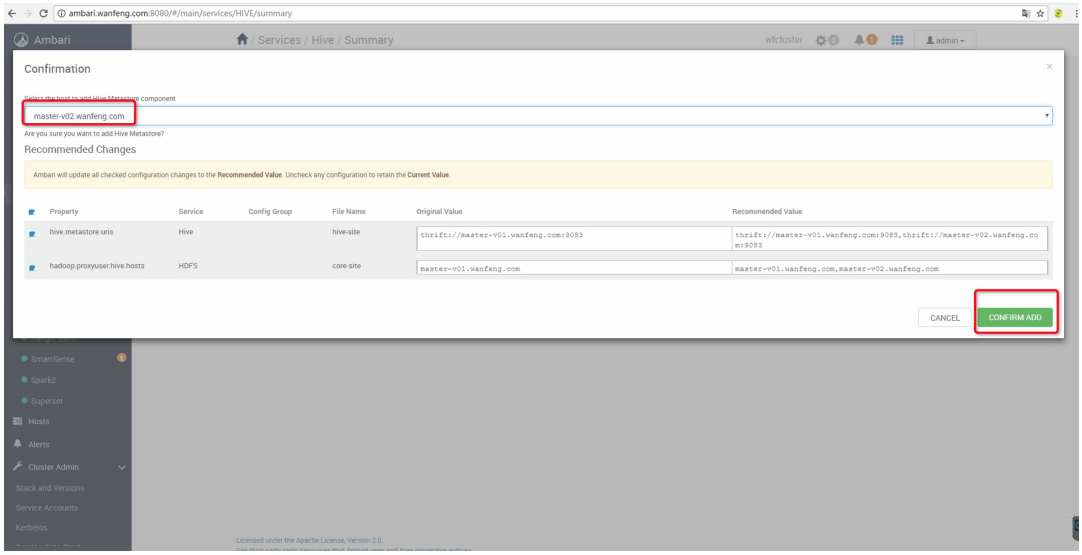
最后点击"COMPLETE"完成操作。

20.3 Hive 高可用配置

Hive 高可用配置很简单，我们已经在 master-v01.wanfeng.com 服务器上安装了"HIVE METASTORE" 和 "HIVESERVER2"两个 Hive相关的服务，现在我们只需要在另一台服务器上(如 master-v02.wanfeng.com)上新加这两个服务就OK了。具体操作如下图



点击"Hive"->"ACTIONS"->"ADD Hive Metastore" 在弹出的"Confirmation"窗口选择 master-v02.wanfeng.com 服务器，如下图

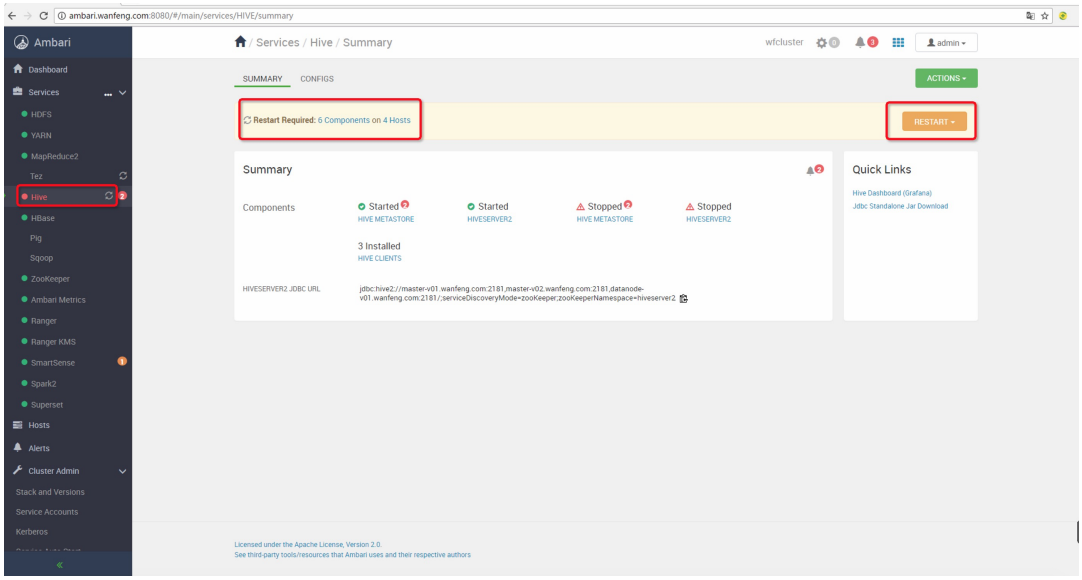


在上图中，配置后，点击"CONFIRM ADD"按钮，弹出"Background OPerations"窗口，点击"OK"按钮回到 Ambari 主控制页。如下图

注意： 不管"Background OPerations" 有没有相关操作在运行，均可随时退出去，后面也可以随时进入，不会影响相关操作的运行。
进入"Background OPerations"的按钮如下图红框所示。

同理添加 "HIVESERVER2"到 master-v02.wanfeng.com 服务器，不再赘述。

注意： 添加某组件或修改某配置文件后，如下图红框，带有刷新标记表示该组件有某些服务需要重启，具体可以点进去进一步看哪些服务需要重启，重启组件也很讲究方法（自己慢慢摸索，这里不赘述）。



至此安装告一段落，下一步我们将基于 FreeIPA 配置高安全的大数据环境。敬请期待。

欢迎加"**万峰大数据交流**" QQ群：**661945126 进一步学习交流。**