

gerrit迁移方案及实施步骤

迁移原因：因原gerrit服务器（192.168.16.81）磁盘空间不足，会导致数据无法写入的情况发生，且数据库为h2，导致后续数据量增多时，会无法存储，故此对gerrit进行迁移。

迁移方案：方案有两种，具体如下：

1. h2数据库方案

该方案的特点为，保持原有数据结构和数据存储方式，保证原有gerrit服务可正常使用。缺点是，当数据量超出一定数量后（官方给出的h2数据库最大存储容量为2G），将无法存储。

2. mysql/oracle/postgresql数据库方案

该方案的特点为：数据库可以存放的数据量成几何倍数增长，且无需考虑最大存储容量。缺点是，数据迁移过程比较繁琐，且容易出现数据迁移后，无法呈现数据的问题，需要深入研究，才可线上实施。

目前，新gerrit服务器采用的部署方案为方案一，请知悉！

迁移步骤：

1. 初始化服务器

(1) 打开文件数限制设置

编辑文件：vim /etc/security/limits.conf

在文件末尾，添加如下内容：

```
* soft nproc 20480
* hard nproc 20480
root soft nproc unlimited
root hard nproc unlimited
* soft nofile 1048576
* hard nofile 1048576
```

保存后，执行sysctl -p，使得配置生效。

(2) jdk设置

jdk的安装方式有两种，一种是yum安装，一种是编译安装。本次安装采用编译方式

下载oraclejdk8u版本，地址为：<https://www.oracle.com/technetwork/java/javase/downloads/java-archive-javase8-2177648.html>

下载后，将下载包，上传到服务器上，并进行解压。

解压后，将路径记录，配置环境变量，文件为：vim /etc/profile

```
export JAVA_HOME=/opt/platform/tools/jdk1.8.0_121
export JAVA_BIN=$JAVA_HOME/bin
export PATH=$JAVA_HOME/bin:$PATH
export CLASSPATH=.:$JAVA_HOME/lib/dt.jar:$JAVA_HOME/lib/tools.jar
```

添加完成后，source /etc/profile，使得配置生效。

(3) git设置

git的安装方式非常简单，直接操作命令：yum -y install git

(4) 创建gerrit用户

命令为：groupadd gerrit && useradd -g gerrit gerrit

(5) 创建gerrit存储目录，并授权

命令为：mkdir /data && chown -R gerrit.gerrit /data

2. 安装gerrit

(1) 下载gerrit.war文件，地址为：<https://gerrit-ci.gerritforge.com/>，根据需要的版本，下载相关文件，本地下载的版本为2.15.3版本。将下载好的文件，放置在/data目录下

```
[root@node2 data]# ll
total 70612
-rw-r--r-- 1 gerrit gerrit 72306588 Dec 10 01:31 gerrit.war
[root@node2 data]# pwd
/data
[root@node2 data]#
```

(2) 安装gerrit

命令为：java -jar gerrit.war init -d /data/gerrit，该命令解释为：启动gerrit.war包程序，装载目录为/data/gerrit

```
[root@node2 data]# java -jar gerrit.war init -d /data/gerrit
Using secure store: com.google.gerrit.server.securestore.DefaultSecureStore
[2018-12-10 01:34:34,706] [main] INFO com.google.gerrit.server.config.GerritServerConfigProvider : No /data/gerrit/etc/gerrit.config; assuming defaults
```

```
*** Gerrit Code Review 2.15.3
***
```

```
Create '/data/gerrit'      [Y/n]? y
```

```
*** Git Repositories
***
```

```
Location of Git repositories [git]:
```

```
*** SQL Database
***
```

```
Database server type      [h2]:
```

```
*** NoteDb Database
***
```

```
Use NoteDb for change metadata?
```

```
  See documentation:
  https://gerrit-review.googlesource.com/Documentation/note-db.html
```

```
Enable                  [Y/n]? n
```

```
Use NoteDb for change metadata?
```

```
  See documentation:
```

```
  https://gerrit-review.googlesource.com/Documentation/note-db.html
```

```
Enable                  [Y/n]? n
```

```
*** Index
***
```

```
Type                  [lucene/?]:
```

```
*** User Authentication
***
```

```
Authentication method    [openid/?]: LDAP
```

```
Git/HTTP authentication  [http/?]:
```

```
LDAP server             [ldap://localhost]: 192.16.16.202:389
```

```
Use SSL                 [y/N]? n
```

```
LDAP username            : auth@deepblueai.com
```

```
auth@deepblueai.com's password :
```

```
        confirm password :
```

```
Account BaseDN          [DC=168,DC=16,DC=202:389]: ou=deepblueai,dc=deepblueai,dc=com
```

```
Group BaseDN            [ou=deepblueai,dc=deepblueai,dc=com]: ou=deepblueai,dc=deepblueai,dc=com
```

```
Enable signed push support  [y/N]? n
```

```
*** Review Labels
***
```

```
Install Verified label   [y/N]? y
```

```

Install Verified label      [y/N]? y
*** Email Delivery
*** 

SMTP server hostname      [localhost]: smtp.exmail.qq.com
SMTP server port          [(default)]: 465
SMTP encryption           [none/?]: ssl
SMTP username              [root]: smtp.exmail.qq.com
smtp.exmail.qq.com's password :
    confirm password :

*** Container Process
*** 

Run as                  [root]: gerrit
Java runtime             [/opt/platform/tools/jdk1.8.0_121/jre]:
Copy gerrit.war to /data/gerrit/bin/gerrit.war [Y/n]? y
Copying gerrit.war to /data/gerrit/bin/gerrit.war

*** SSH Daemon
*** 

Listen on address        [*]:
Listen on port            [29418]:
Generating SSH host key ... rsa... dsa... ed25519... ecdsa 256... ecdsa 384... ecdsa 521... done

*** HTTP Daemon
*** 

Behind reverse proxy     [y/N]? n
Use SSL (https://)        [y/N]? n
Listen on address        [*]:
Listen on port            [8080]:
Canonical URL            [http://node2:8080/]:

*** Cache
*** 

*** Plugins
*** 

Installing plugins.
Install plugin commit-message-length-validator version v2.15.3 [y/N]? y
Installed commit-message-length-validator v2.15.3
Install plugin download-commands version v2.15.3 [y/N]? y
Installed download-commands v2.15.3
Install plugin hooks version v2.15.3 [y/N]? y
Installed hooks v2.15.3
Install plugin replication version v2.15.3 [y/N]? y
Installed replication v2.15.3
Install plugin reviewnotes version v2.15.3 [y/N]? y
Installed reviewnotes v2.15.3
Install plugin singleusergroup version v2.15.3 [y/N]? y
Installed singleusergroup v2.15.3
Initializing plugins.

*** Experimental features
*** 

Enable any experimental features [y/N]? n
Initialized /data/gerrit

gerrit安装已完成。

```

(3) 修改配置文件/data/gerrit/etc/gerrit.config，修改后，配置文件如下：

```

[gerrit]
  basePath = git
  serverId = ad625552-4dc5-4269-b385-e231b52e036f
  canonicalWebUrl = http://node2:8080/
[database]
  type = h2
  database = /data/gerrit/db/ReviewDB
[index]
  type = LUCENE
[auth]
  type = LDAP
[ldap]
  server = ldap://192.168.16.202:389
  username = auth@deepblueai.com
  accountBase = ou=deepblueai,dc=deepblueai,dc=com
  groupBase = ou=deepblueai,dc=deepblueai,dc=com
  accountPattern = (&(objectClass=user)(sAMAccountName=${username}))
  groupPattern = (&(objectClass=group)(cn=${groupname}))
  accountFullName = displayName
  accountEmailAddress = mail
  sslVerify = false
[receive]
  enableSignedPush = false
[sendemail]
  smtpServer = smtp.exmail.qq.com
  smtpServerPort = 465
  smtpEncryption = ssl
  smtpUser = support@deepblueai.com

[receive]
  enableSignedPush = false
[sendemail]
  smtpServer = smtp.exmail.qq.com
  smtpServerPort = 465
  smtpEncryption = ssl
  smtpUser = support@deepblueai.com
  sslVerify = false
  from=CodeReview<support@deepblueai.com>
[container]
  user = gerrit
  javaHome = /opt/platform/tools/jdk1.8.0_121/jre
[sshd]
  listenAddress = *:29418
[httpd]
  listenUrl = http://*:8080/
[cache]
  directory = cache
[plugins]
  allowRemoteAdmin = true
[plugin "events-log"]
  maxAge = 20
  returnLimit = 10000
  storeUrl = jdbc:h2:gerrit.deepblueai.com/data/db
  urlOptions = logLevel=INFO
  urlOptions = logUnclosedConnections=true
  copyLocal = true

```

(4) 配置安装完成后，对/data目录进行授权，并启动gerrit

命令为：chown -R gerrit.gerrit /data && /data/gerrit/bin/gerrit.sh start

```

[root@node2 gerrit]# chown -R gerrit.gerrit /data/
[root@node2 gerrit]# /data/gerrit/bin/gerrit.sh start
Starting Gerrit Code Review: OK

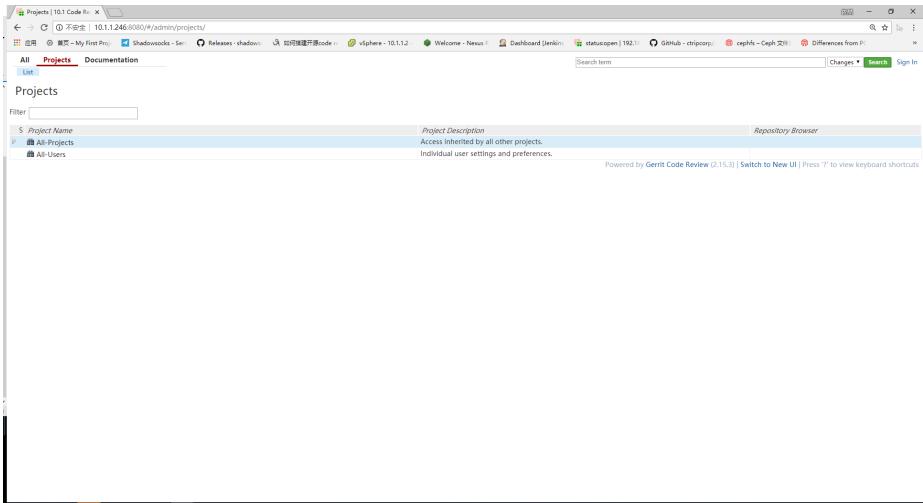
```

```

[root@node2 opt]# tailf /data/gerrit/logs/error_log
[2018-12-10 01:47:05,228] [main] INFO com.google.gerrit.server.plugins.PluginLoader : Loaded plugin singleusergroup, version v2.15.3
[2018-12-10 01:47:05,694] [main] INFO com.google.gerrit.server.git.GarbageCollectionRunner : Ignoring missing gc schedule configuration
[2018-12-10 01:47:05,695] [main] INFO com.google.gerrit.server.config.ScheduleConfig : accountDeactivation schedule parameter "accountDeactivation.interval" is not configured
[2018-12-10 01:47:05,695] [main] INFO com.google.gerrit.server.change.ChangeCleanupRunner : Ignoring missing changeCleanup schedule configuration
[2018-12-10 01:47:05,717] [main] INFO com.google.gerrit.sshd.SshDaemon : Started Gerrit SSHD-CORE-1.6.0 on *:29418
[2018-12-10 01:47:05,720] [main] INFO org.eclipse.jetty.server.Server : jetty-9.3.18.v20170406
[2018-12-10 01:47:06,081] [main] INFO org.eclipse.jetty.server.handler.ContextHandler : Started o.e.j.s.ServletContextHandler@739500b8{/,,null,AVAILABLE}
[2018-12-10 01:47:06,094] [main] INFO org.eclipse.jetty.server.AbstractConnector : Started ServerConnector@14f08a97{HTTP/1.1,[http/1.1]}{0.0.0.0:8080}
[2018-12-10 01:47:06,095] [main] INFO org.eclipse.jetty.server.Server : Started @11369ms
[2018-12-10 01:47:06,097] [main] INFO com.google.gerrit.pgm.Daemon : Gerrit Code Review 2.15.3 ready

```

(5) 查看安装后，gerrit页面



3. 安装nginx

新gerrit服务，采用域名方式访问。即，需安装nginx或者tenginx来进行负载，安装方式如下：

(1) 安装初始化

安装nginx必要组件：gcc*/pcre*/openssl*/zlib*

命令为：yum -y install gcc* pcre* openssl* zlib*

(2) 下载并安装nginx

下载地址：<http://nginx.org/download/>，本地下载版本为nginx-1.14.2版本

安装：将下载的包，放置在/opt目录下，并解压。

```
#tar -zxf nginx-1.14.2.tar.gz
```

进入nginx安装目录：cd nginx-1.14.2

安装nginx：./configure --prefix=/usr/local/nginx --with-http_ssl_module --with-http_stub_status_module --with-pcre

```
# make -j6 && make install
```

(3) 配置nginx

编辑nginx.conf文件

```
#user nobody;
worker_processes auto;

#error_log  logs/error.log;
#error_log  logs/error.log  notice;
#error_log  logs/error.log  info;

pid      logs/nginx.pid;

events {
    use epoll;
    worker_connections 10240;
}

http {
    include      mime.types;
    default_type application/octet-stream;

    log_format  main  '$remote_addr - $remote_user [$time_local] "$request" '
                      '$status $body_bytes_sent "$http_referer" '
                      '"$http_user_agent" "$http_x_forwarded_for"';

    access_log  logs/access.log  main;
    sendfile    on;
```

```

events {
    use epoll;
    worker_connections 10240;
}

http {
    include mime.types;
    default_type application/octet-stream;

    log_format main '$remote_addr - $remote_user [$time_local] "$request" '
                    '$status $body_bytes_sent "$http_referer"'
                    '"$http_user_agent" "$http_x_forwarded_for"';

    access_log logs/access.log main;

    sendfile on;
    #tcp_nopush on;

    keepalive_timeout 0;
    #keepalive_timeout 65;

    gzip on;

    include /usr/local/nginx/vhosts/*.conf;
}

```

编辑gerrit.conf文件

```

upstream gerrit {
    server 10.1.1.246:8080;
}

server
{
    listen 80;
    server_name gerrituat.deepblueai.com;
    access_log /var/log/nginx/gerrit.access.log main;

    location / {
        proxy_pass http://gerrit;
        proxy_set_header Host $host;
        proxy_set_header X-Real-IP $remote_addr;
        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
    }
}
~
~
~
~
```

配置完成后，使用nginx -t命令进行检查，如没问题，则如下图，并启动nginx

```
[root@node2 vhosts]# nginx -t
nginx: the configuration file /usr/local/nginx/conf/nginx.conf syntax is ok
nginx: configuration file /usr/local/nginx/conf/nginx.conf test is successful
```

```
[root@node2 vhosts]# nginx
[root@node2 vhosts]# ps -ef | grep nginx
root 25183 1 3 02:34 ? 00:00:00 nginx: master process nginx
nobody 25184 25183 4 02:34 ? 00:00:00 nginx: worker process
nobody 25185 25183 5 02:34 ? 00:00:00 nginx: worker process
nobody 25186 25183 5 02:34 ? 00:00:00 nginx: worker process
nobody 25187 25183 5 02:34 ? 00:00:00 nginx: worker process
nobody 25188 25183 5 02:34 ? 00:00:00 nginx: worker process
nobody 25189 25183 5 02:34 ? 00:00:00 nginx: worker process
nobody 25190 25183 5 02:34 ? 00:00:00 nginx: worker process
nobody 25191 25183 0 02:34 ? 00:00:00 nginx: worker process
root 25193 21593 0 02:34 pts/2 00:00:00 grep --color=auto nginx
[root@node2 vhosts]#
```

修改gerrit配置文件

```
[gerrit]
  basePath = git
  serverId = ad625552-4dc5-4269-b385-e231b52e036f
  canonicalWebUrl = http://gerrituat.deepblueai.com
[database]
```

修改后，重启gerrit，/data/gerrit/bin/gerrit.sh restart

```
[root@node2 etc]# /data/gerrit/bin/gerrit.sh restart
Stopping Gerrit Code Review: OK
Starting Gerrit Code Review: OK
```

4. 数据迁移及配置修改，无需按照步骤，此项定义为数据迁移中，必须要做的注意事项

- (1) 重建索引：java -jar gerrit.war reindex -d /path/to/gerrit
- (2) 将原服务器上/data/gerrit目录中的，data/db/events-db/git/index/plugins目录打包，并发送到新的服务器上。
- (3) 将gerrit.config文件中，gerrit节点下的serverId字段，修改为原服务器的serverId，否则会出现错误或项目无法读取情况。