综合项目报告书

黄达明

根据项目要求，搭建要遵循以下要求：

1、MYSQL 的搭建

2、nginx 的七层的负载均衡集群

3、tomcat 的集群（后端节点会话（session）的一致性）

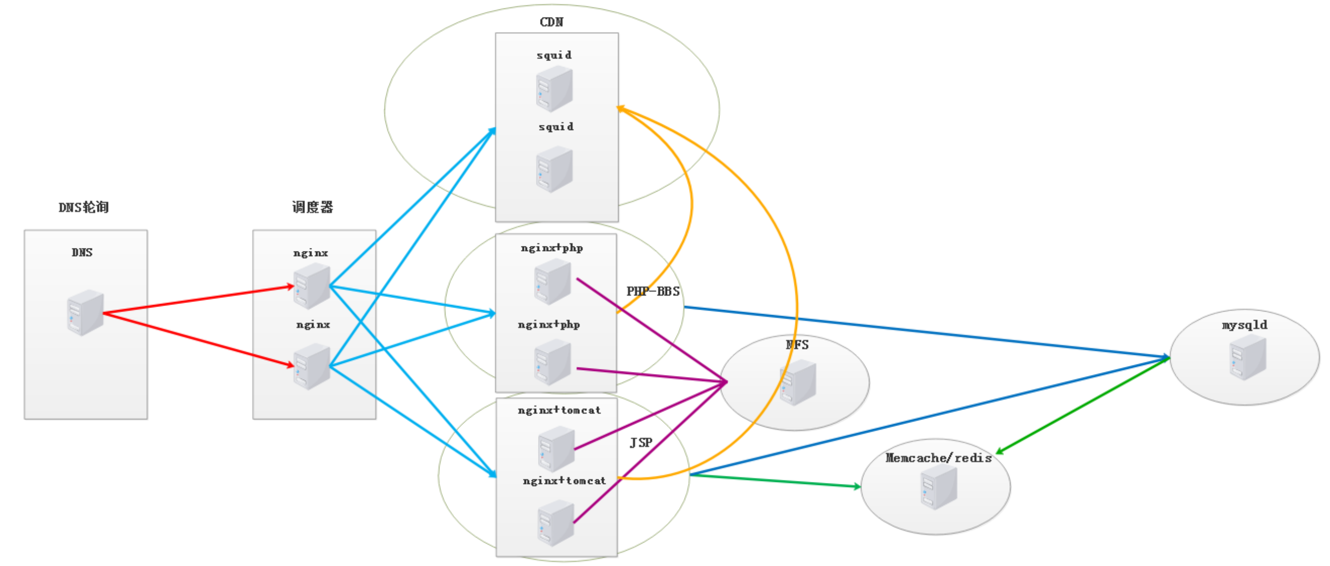
4、利用分布性存储（NFS）实现页面一致性

5、引入 CDN 内容分发网络，实现网站静态元素加速

6、利用 nginx 七层分发器实现基于内容的分发

7、利用智能 DNS 实现大并发流量切割

总的拓扑图如下



+++

准备工作：10 台服务器

DNS 轮询　　172.25.254.4(真机)

Nginx 反向代理(调度)

nginx-001 172.25.4.10

nginx-002 172.25.4.11

Squid 静态缓存加速器

squid-001 172.25.4.12

squid-002 172.25.4.13

Web 服务器

PHP 论坛: www.php-f4.com

nginx1+php1 172.25.4.14

\

 NFS(Discuz) 页面数据一致性   172.25.254.4(真机)

/

nginx2+php2 172.25.4.15

JSP 门户网站: www.jsp-f4.com

nginx1+tomcat1 172.25.4.16

\

 NFS(ejforum) 页面数据一致性   172.25.254.4(真机)

/

nginx1+tomcat1 172.25.4.17

memcached/redis 数据库缓存加速器(保证 session 一致性) 172.25.4.18

数据库: mariadb-server 172.25.4.19

++

批量上传公钥并做系统初始化操作:

```
[root@foundation4 ~]# vim /shells/hosts   #创建远程主机 ip 信息
172.25.4.10
172.25.4.11
172.25.4.12
172.25.4.13
172.25.4.14
172.25.4.15
172.25.4.16
172.25.4.17
172.25.4.18
172.25.4.19
```

```
[root@foundation4 shells]# vim ssh_init.sh
!/bin/bash
keydir=$HOME/.ssh
skey=$keydir/id_rsa
pkey=$keydir/id_rsa.pub
passwd=uplooking

# genkey
```

```
genkey(){
    ssh-keygen -q -f $skey -N ""    #生成本地公钥和私钥
    }

# key is exist.

[ -f $skey -a -f $pkey ] || genkey  #判断本地是否存在公钥和私钥，如无则生成

rinit(){
        ssh root@$rhost "iptables -F;setenforce 0;sed -i
's/SELINUX=.*/SELINUX=disabled/' /etc/selinux/config"   #远程关闭防火墙、
selinux
}

for rhost  in $(cat host.txt)           #遍历远程主机 ip
   do
      expect <<EOF
      spawn  ssh-copy-id  root@$rhost   #远程上传公钥
      expect {
            "*(yes/no)?"  { send "yes\r";exp_continue }
            "*password:"  { send "$passwd\r";exp_continue }
            eof { exit }
            }
EOF
        rinit
   done

[root@foundation4 shells]# chmod +x ssh_init.sh
[root@foundation4 shells]# ./ssh_init.sh
```

1）配置 nginx+php

部署前准备：

```
[root@php-server1 ~]# hostnamectl set-hostname php-server1
```

安装软件：

```
[root@php-server1 ~]# rpm -ivh nginx-1.8.1-1.el7.ngx.x86_64.rpm spawn-fcgi-
1.6.3-5.el7.x86_64.rpm
[root@php-server1 ~]# yum install -y php php-mysql
```

配置虚拟主机：

```
[root@php-server1 ~]# mv /etc/nginx/conf.d/default.conf
/etc/nginx/conf.d/www.php-f4.com.conf
[root@php-server1 ~]# vim /etc/nginx/conf.d/www.php-f4.com.conf
```

```
listen 80;
server_name www.php-f4.com;
location / {
root /usr/share/nginx/php-f4.com;
index index.php index.html index.htm;
}
}
```

创建目录：

```
[root@php-server1 ~]# mkdir /usr/share/nginx/php-f4.com
```

下载论坛压缩包、解压并拷贝 bbs 论坛文件到网站根目录：

```
[root@php-server1 ~]# cp -r upload/* /usr/share/nginx/php-f4.com/
```

修改权限：

```
[root@php-server1 ~]# chown nginx. -R /usr/share/nginx/php-f4.com/
```

编辑 fcgi 配置文件：

```
[root@php-server1 ~]# vim /etc/sysconfig/spawn-fcgi
OPTIONS="-u nginx -g nginx -p 9000 -C 32 -F 1 -P /var/run/spawn-fcgi.pid --
/usr/bin/php-cgi"
```

启动服务：

```
[root@php-server1 ~]# systemctl start nginx
[root@php-server1 ~]# systemctl enable nginx
[root@php-server1 ~]# systemctl start spawn-fcgi.service
[root@php-server1 ~]# chkconfig spawn-fcgi on
```

2)配置数据库：

部署前准备：

```
[root@mysql-server ~]# hostnamectl set-hostname mariadb-server
```

安装软件：

```
[root@mysql-server ~]# yum install -y mariadb-server
```

启动服务：

```
[root@mysql-server ~]# systemctl start mariadb
[root@mysql-server ~]# systemctl enable mariadb
```

进入数据库配置：

```
[root@mysql-server ~]# mysql
MariaDB [(none)]> create database bbs default charset utf8;
MariaDB [(none)]> delete from mysql.user where user='';
MariaDB [(none)]> update mysql.user set password=password('redhat');
MariaDB [(none)]> grant all on bbs.* to bbs@'172.25.4.%' identified by
'uplooking';
MariaDB [(none)]> flush privileges;
MariaDB [(none)]> exit;
```

--------------------lnmp 架构搭建完成----------

修改hosts文件，指向php-server1上

测试：ok，打开页面：http://www.php-f4.com，

数据库地址：192.168.4.19

数据库用户名：bbs

数据库密码：redhat

3）部署php-server2

部署前准备：

```
[root@php-server2 pkg]# hostnamectl set-hostname php-server2
```

安装软件：

```
[root@php-server2 pkg]# rpm -ivh nginx-1.8.1-1.el7.ngx.x86_64.rpm spawn-fcgi-1.6.3-5.el7.x86_64.rpm
[root@php-server2 pkg]# yum install -y php php-mysql
```

同步文件给php-server2

```
[root@php-server1 ~]# rsync -avzR /usr/share/nginx/php-f4.com/ root@172.25.4.15:/
[root@php-server1 ~]# rsync -avzR /etc/sysconfig/spawn-fcgi root@172.25.4.15:/
[root@php-server1 ~]# rsync -avzR /etc/nginx/conf.d/www.php-f4.com.conf root@172.25.4.15:/
```

启动服务：

```
[root@php-server2 ~]# systemctl start nginx
[root@php-server2 ~]# systemctl enable nginx
[root@php-server2 ~]# systemctl start spawn-fcgi.service
[root@php-server2 ~]# chkconfig spawn-fcgi on
```

测试：

修改hosts文件，指向php-server2上

停止 php-server1上的nginx服务，页面可正常访问

----

4)部署tomcat-server1

部署前准备：

```
[root@tomcat-server1 ~]# hostnamectl set-hostname tomcat-server1
```

安装软件：

```
[root@tomcat-server1 pkg]# rpm -ivh nginx-1.8.1-1.el7.ngx.x86_64.rpm
[root@tomcat-server1 pkg]# tar -xf apache-tomcat-8.0.24.tar.gz
[root@tomcat-server1 pkg]# mkdir /usr/local/tomcat
[root@tomcat-server1 apache-tomcat-8.0.24]# cp -r ./* /usr/local/tomcat/
[root@tomcat-server1 pkg]# tar -xf jdk-7u15-linux-x64.tar.gz
[root@tomcat-server1 pkg]# mv jdk1.7.0_15/ /usr/local/java/
```

```
[root@tomcat-server1 pkg]# export JAVA_HOME=/usr/local/java #申明 jdk 所在位置
```

添加用户和组：

```
[root@tomcat-server1 bin]# groupadd -g 499 tomcat
[root@tomcat-server1 bin]# useradd -g 499 -u 499 tomcat
```

编译 jsvc

```
[root@tomcat-server1 bin]# yum install -y gcc ---安装编译器
[root@tomcat-server1 bin]# cd /usr/local/tomcat/bin
[root@tomcat-server1 bin]# tar -xf commons-daemon-native.tar.gz
[root@tomcat-server1 unix]# ./configure
[root@tomcat-server1 unix]# make
[root@tomcat-server1 unix]# cp jsvc /usr/local/tomcat/bin/
[root@tomcat-server1 unix]# cd /usr/local/tomcat/bin
[root@tomcat-server1 bin]# cp daemon.sh /etc/init.d/tomcat # jsvc 启动脚本复制到/etc/init.d 目录下
[root@tomcat-server1 bin]# vim /etc/init.d/tomcat
# chkconfig: 2345 20 10 # 设置开机自启参数
CATALINA_HOME=/usr/local/tomcat # 申明 tomcat 命令和库文件所在位置
CATALINA_BASE=/usr/local/tomcat # 申明 tomcat 程序和配置文件及网站根目录所在位置
JAVA_HOME=/usr/local/java # 申明 jdk 所在位置

[root@tomcat-server1 bin]# chkconfig --add tomcat
[root@tomcat-server1 bin]# chkconfig tomcat on
[root@tomcat-server1 bin]# chown tomcat. -R /usr/local/tomcat # 由于实际处理请求的用户身份为 tomcat，该用户则必须有对应访问配置文件等的权限。
[root@servera tomcat]# ps -ef | grep tomcat # 查看进程
```

配置 tomcat 主配置文件：

```
[root@tomcat-server1 ~]# vim /usr/local/tomcat/conf/server.xml

<Host name="www.jsp-f4.com" appBase="jsp-f4.com" #设置主页和网页根目录
unpackWARs="true" autoDeploy="true">
<Valve className="org.apache.catalina.authenticator.SingleSignOn" />
<Valve className="org.apache.catalina.valves.AccessLogValve" directory="logs"
prefix="jsp-f4.com_access_log" suffix=".txt" #设置日志
pattern="%h %l %u %t &quot;%r&quot; %s %b" />

</Host>
```

```
[root@tomcat-server1 tomcat]# service tomcat stop
[root@tomcat-server1 tomcat]# service tomcat start
[root@tomcat-server1 ~]# cd /usr/local/tomcat/jsp-f4.com/
[root@tomcat-server1 jsp-f4.com]# mkdir ROOT
[root@tomcat-server1 jsp-f4.com]# cd ROOT/
[root@tomcat-server1 ROOT]# echo "tomcat-server1" > index.jsp
```

配置 nginx 主配置文件：

```
[root@tomcat-server1 pkg]# vim /etc/nginx/nginx.conf
upstream java_upstream {
server 127.0.0.1:8080 max_fails=2 fail_timeout=30s; #设置轮询器
}
[root@tomcat-server1 pkg]# vim /etc/nginx/conf.d/default.conf
server {
listen 80;
server_name www.jsp-f4.com;

location / {
root /usr/local/tomcat/jsp-f4.com/ROOT;
index index.jsp index.html index.htm;
}

location ~ \.jsp$ { #匹配到以.jsp结尾的动态页面解析交给tomcat处理
proxy_set_header Host $host;
proxy_set_header X-Forward-For $remote_addr;
proxy_pass http://java_upstream; #添加代理
}
}
[root@tomcat-server1 pkg]# systemctl start nginx
[root@tomcat-server1 pkg]# systemctl enable nginx
-----------------------------------------------------------------
```

部署门户网站：

1.下载安装包<ejforum-2.3.zip>并解压

```
[root@tomcat-server1 pkg]# unzip ejforum-2.3.zip
```

2.将网页文件放置网站根目录下

```
[root@tomcat-server1 ejforum-2.3]# \cp -r ejforum/* /usr/local/tomcat/jsp-f4.com/ROOT/
```

3.配置和数据库的连接

下载安装包<mysql-connector-java-5.1.36.tar.gz>并解压

```
[root@tomcat-server1 pkg]# tar -xf mysql-connector-java-5.1.36.tar.gz
[root@tomcat-server1 mysql-connector-java-5.1.36]# pwd
/root/pkg/mysql-connector-java-5.1.36
```

```
[root@tomcat-server1 mysql-connector-java-5.1.36]# cp mysql-connector-java-5.1.36-bin.jar /usr/local/tomcat/lib/ # 将数据库的连接文件放置到 lib 目录下
[root@tomcat-server1 ~]# cd /usr/local/tomcat/jsp-f4.com/ROOT/WEB-INF/
[root@tomcat-server1 WEB-INF]# ls
[root@tomcat-server1 WEB-INF]# cd conf/
[root@tomcat-server1 conf]#
[root@servera conf]# vim config.xml # 将第一段注释掉，打开第二段 mysql 的连接
<!-- DB Connection Pool - Mysql-->
<database maxActive="10" maxIdle="10" minIdle="2" maxWait="10000"
username="portal" password="uplooking" #连接数据库用户名和密码
driverClassName="com.mysql.jdbc.Driver"
url="jdbc:mysql://172.25.4.19:3306/portal? #数据位置
characterEncoding=gbk&amp;autoReconnect=true&amp;autoReconnectForPools=true&amp;zeroDateTimeBehavior=convertToNull"
sqlAdapter="sql.MysqlAdapter"/>
```

配置数据库服务器

```
[root@tomcat-server1 script]# pwd
/root/pkg/ejforum-2.3/install/script
[root@tomcat-server1 script]# scp easyjforum_mysql.sql root@172.25.4.19:/root # 拷贝 sql 文件到数据库服务器上
```

数据库服务器上设置：

```
[root@mysql-server ~]# mysqladmin create database portal #创建数据库
[root@mysql-server ~]# mysql -uroot -predhat < /root/easyjforum_mysql.sql #导入数据
```

数据库授权：

```
[root@mysql-server ~]# mysql -uroot -predhat
MariaDB [(none)]> grant all on portal.* to portal@'172.25.4.%' identified by 'uplooking';
MariaDB [(none)]> flush privileges;
```

修改权限相关

```
[root@tomcat-server1 ~]# chown tomcat. -R /usr/local/tomcat/
[root@tomcat-server1 ~]# service tomcat stop
[root@tomcat-server1 ~]# service tomcat start
```

修改 hosts 文件，指向 tomcat-server1 上

测试：浏览器访问 http:www.jsp-f4.com:8080,测试 ok

5)部署 tomcat-server2

```
[root@serverh pkg]# hostnamectl set-hostname tomcat-server2
```

安装 nginx 软件

```
[root@tomcat-server2 pkg]# rpm -ivh nginx-1.8.1-1.el7.ngx.x86_64.rpm
```

从 tomcat-server1 同步相关文件

```
[root@tomcat-server1 ~]# rsync -azvR /usr/local/tomcat/ /etc/rc.d/init.d/tomcat /usr/local/java/ root@172.25.4.17:/
[root@tomcat-server1 ~]# rsync -azvR /etc/nginx/ root@172.25.4.17:/
```

添加用户和组

```
[root@serverh pkg]# groupadd -g 499 tomcat
[root@serverh pkg]# useradd -g tomcat -u 499 tomcat
```

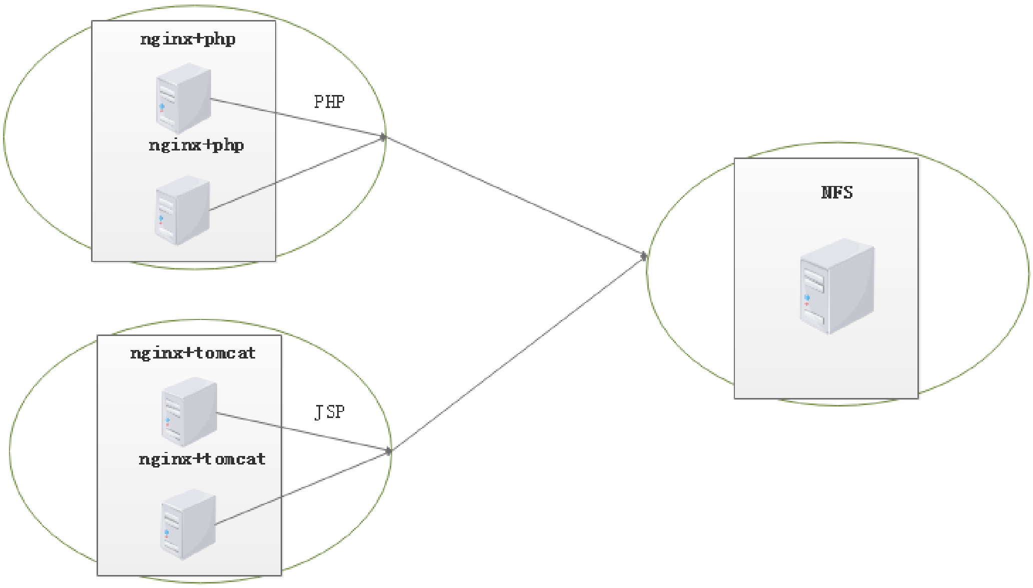
设置开机自启

```
[root@serverh pkg]# chkconfig --add tomcat
[root@serverh pkg]# chkconfig tomcat on
[root@tomcat-server2 pkg]# systemctl start nginx
[root@tomcat-server2 pkg]# systemctl enable nginx
[root@serverh pkg]# service tomcat start
```

测试：

修改 hosts 文件，指向 tomcat-server2 上

停止 tomcat-server1 上的服务，重新访问 http://www.jsp-f4.com:8080,测试 ok



6)部署 NFS

```
[root@foundation4 /]# yum install -y nfs-utils rpcbind
[root@foundation4 /]# setenforce 0;iptables -F
```

从 tomcat-server1 拷贝网站目录到 NFS 服务器：

```
[root@tomcat-server1 ~]# rsync -avRz /usr/local/tomcat/jsp-f4.com/ root@172.25.254.4:/
```

从 php-server1 拷贝网站目录到 NFS 服务器：

```
[root@php-server1 nginx]# rsync -azvR /usr/share/nginx/php-f4.com/ root@172.25.254.4:/
```

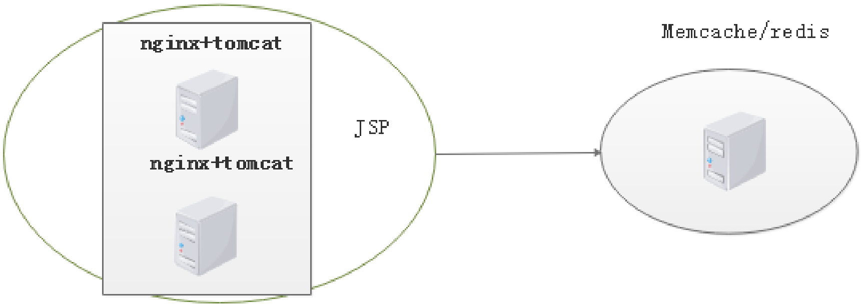
```
[root@foundation4 /]# vim /etc/exports
/usr/local/tomcat/jsp-f4.com 172.25.4.0/24(rw)
/usr/share/nginx/php-f4.com 172.25.4.0/24(rw)
```

```
[root@foundation4 ~]# systemctl start rpcbind
[root@foundation4 ~]# systemctl enable rpcbind
[root@foundation4 ~]# systemctl start nfs
[root@foundation4 ~]# systemctl enable nfs-server
```

设置开机自动挂载：

```
[root@tomcat-server1 ~]# vim /etc/fstab
172.25.254.4:/usr/local/tomcat/jsp-f4.com /usr/local/tomcat/jsp-f4.com nfs defaults 0 0
[root@tomcat-server2 ~]# vim /etc/fstab
172.25.254.4:/usr/local/tomcat/jsp-f4.com /usr/local/tomcat/jsp-f4.com nfs defaults 0 0
```

```
[root@php-server1 ~]# vim /etc/fstab
172.25.254.4:/usr/share/nginx/php-f4.com /usr/share/nginx/php-f4.com nfs defaults 0 0
[root@php-server2 ~]# vim /etc/fstab
172.25.254.4:/usr/share/nginx/php-f4.com /usr/share/nginx/php-f4.com nfs defaults 0 0
```



7)部署 memcached（session 一致）

```
[root@mysql-server ~]# hostnamectl set-hostname memcache-server
```

安装软件：

```
[root@memcache-server ~]# yum install -y memcached
```

第一步:

把下面的软件包都 scp 到 tomcat1 和 tomcat2 上的/usr/local/tomcat/lib/目录下

```
[root@tomcat-server1 ~]# lftp 172.25.254.250
[root@tomcat-server1 msm]# ls
asm-3.2.jar minlog-1.2.jar
kryo-1.04.jar msm-kryo-serializer-1.8.1.jar
kryo-serializers-0.11.jar reflectasm-1.01.jar
memcached-session-manager-1.8.1.jar spymemcached-2.11.1.jar
memcached-session-manager-tc8-1.8.1.jar

[root@tomcat-server1 msm]# cp ./* /usr/local/tomcat/lib/
[root@tomcat-server1 msm]# scp ./* root@172.25.4.17:/usr/local/tomcat/lib #同步给 tomcat-server2
```

第二步:

在 tomcat1 和 tomcat2 上操作

```
[root@tomcat-server1 ~]# vim /usr/local/tomcat/conf/context.xml
<?xml version='1.0’ encoding='utf-8’?>
<Context>
<WatchedResource>WEB-INF/web.xml</WatchedResource>
<WatchedResource>${catalina.base}/conf/web.xml</WatchedResource>
<Manager className="de.javakaffee.web.msm.MemcachedBackupSessionManager"
```

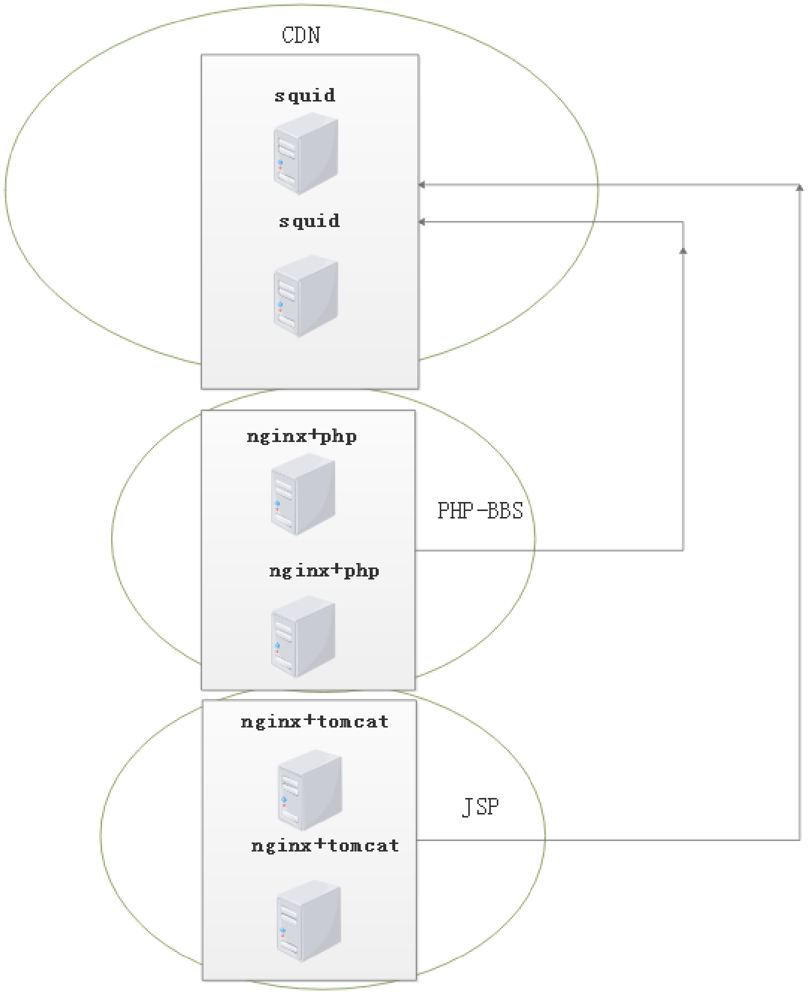
```
memcachedNodes="n1:172.25.1.18:11211" #memcache-server的ip地址，多台时，用逗号隔开，例如："n2:xxx.xxx.xxx.xxx:11211,n3:xxx.xxx.xxx.xxx:11211"
lockingMode="auto"
sticky="false"
requestUriIgnorePattern= ".*\.(png|gif|jpg|css|js)$"
sessionBackupAsync= "false"
sessionBackupTimeout= "100"
copyCollectionsForSerialization="true"
transcoderFactoryClass="de.javakaffee.web.msm.serializer.kryo.KryoTranscoderFactory" />
</Context>
```

```
[root@tomcat-server1 ~]# scp /usr/local/tomcat/conf/context.xml root@172.25.4.17:/usr/local/tomcat/conf/context.xml #同步给tomcat-server2
```

创建测试页面：

```
[root@tomcat-server2 ~]# vim /usr/local/tomcat/jsp-f4.com/ROOT/test.jsp
<html>
<body bgcolor="red">
<center>
<%out.print(request.getSession().getId()) ;%>
<h1>Tomcat</h1>
</body>
</html>
```

测试：开启一台tomcat服务器，写一个测试页面，刷新，查看结果，测试ok



8)部署缓存服务器 squid-server1

```
[root@serverc ~]# hostnamectl set-hostname squid-server1
```

安装软件：

```
[root@squid-server1 ~]# yum install -y squid
```

配置文件：

```
[root@squid-server1 ~]# vim /etc/squid/squid.conf
http_access allow all
http_port 3128 accel vhost vport
cache_dir ufs /var/spool/squid 256 16 256

cache_peer 172.25.4.14 parent 80 0 no-query originserver name=web1 #php 服务器
php-server1
cache_peer 172.25.4.15 parent 80 0 no-query originserver name=web2 #php 服务器
php-server2
cache_peer 172.25.4.16 parent 80 0 no-query originserver name=web3 #jsp 服务器
tomcat-server1
```

```
cache_peer 172.25.4.17 parent 80 0 no-query originserver name=web4 #jsp服务器
tomcat-server2
cache_peer_domain web1 www.php-f4.com
cache_peer_domain web1 172.25.4.14
cache_peer_domain web2 www.php-f4.com
cache_peer_domain web2 172.25.4.15
cache_peer_domain web3 www.jsp-f4.com
cache_peer_domain web3 172.25.4.16
cache_peer_domain web4 www.jsp-f4.com
cache_peer_domain web4 172.25.4.17
```

```
[root@foundation4 ~]# vim /etc/hosts
172.25.4.12 www.jsp-f4.com www.php-f4.com
```

测试：

```
#curl -I http://www.php-f4.com:3128/static/image/click/leiren.gif
```

结果ok

```
# curl -I http://www.jsp-f4.com:3128/images/google.png
```

结果ok

----

9)部署缓存服务器squid-server2

```
[root@serverd ~]# hostnamectl set-hostname squid-server2

[root@squid-server1 ~]# ssh root@172.25.4.13 "yum install -y squid" #远程安装软件
[root@squid-server1 ~]# rsync -avzR /etc/squid/squid.conf root@172.25.4.13:/ #同步文件给squid-server2
[root@squid-server1 ~]# ssh root@172.25.4.13 "systemctl start squid" #远程启动服务
```

```
[root@foundation4 ~]# vim /etc/hosts
172.25.4.13 www.jsp-f4.com www.php-f4.com
```

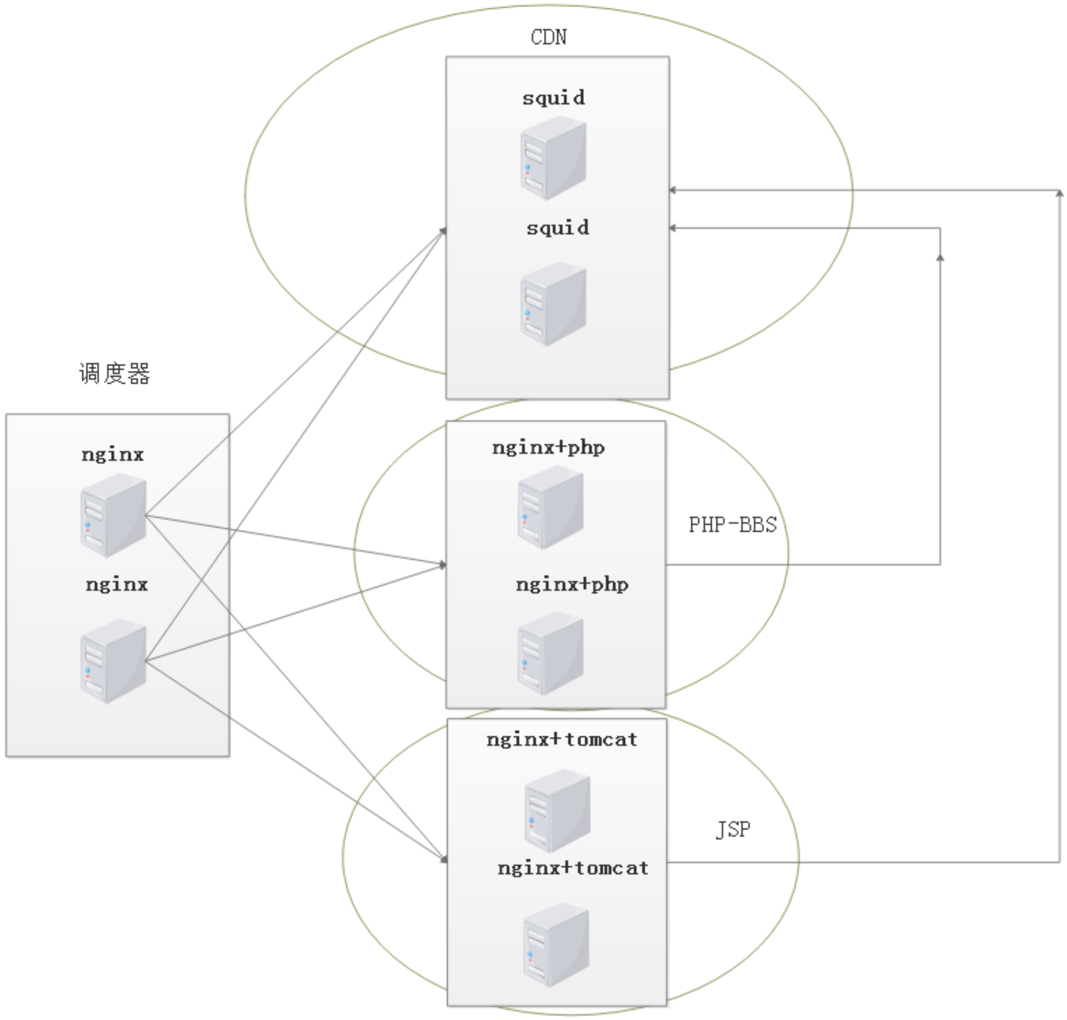
测试：

```
#curl -I http://www.php-f4.com:3128/static/image/click/leiren.gif
```

结果ok

```
# curl -I http://www.jsp-f4.com:3128/images/google.png
```

结果ok



10)部署 nginx-server1 调度器

```
[root@servera ~]# hostnamectl set-hostname nignx-server1
```

安装软件和修改配置文件:

```
[root@nignx-server1 ~]# rpm -ivh nginx-1.8.1-1.el7.ngx.x86_64.rpm
[root@nignx-server1 ~]# vim /etc/nginx/nginx.conf
upstream squid { #增加 3 个轮询器，分别为静态加速器，php 动态页面处理，jsp 动态页面处理
server 172.25.4.12:3128 weight=1 max_fails=2 fail_timeout=1s;
server 172.25.4.13:3128 weight=1 max_fails=2 fail_timeout=1s;
}
upstream php {
server 172.25.4.14:80 weight=1 max_fails=2 fail_timeout=1s;
server 172.25.4.15:80 weight=1 max_fails=2 fail_timeout=1s;

}
upstream jsp {
server 172.25.4.16:8080 weight=1 max_fails=2 fail_timeout=1s;
server 172.25.4.17:8080 weight=1 max_fails=2 fail_timeout=1s;

}

[root@nignx-server1 ~]# vim /etc/nginx/conf.d/default.conf

server {
listen 80;
server_name 127.0.0.1; #注意，添写为本地的ip地址，不然无法处理静态页面
location / { #匹配到php动态页面交给后面php的处理，匹配到jsp动态页面交给后面jsp的处理，其余交给缓存服务器处理
index index.php index.jsp index.html index.htm;
proxy_pass http://squid;
proxy_set_header Host $host;
proxy_set_header X-Forwarded-For $remote_addr;
}
location ~ .*\.php$ {
proxy_pass http://php;
proxy_set_header Host $host;
proxy_set_header X-Forwarded-For $remote_addr;
}

location ~ .*\.jsp$ {
proxy_pass http://jsp;
proxy_set_header Host $host;
proxy_set_header X-Forwarded-For $remote_addr;
}
}
```

测试：修改hosts文件

```
[root@foundation4 ~]# vim /etc/hosts
172.25.4.10 www.php-f4.com www.jsp-f4.com
```

测试结果ok

11)部署nginx-server2调度器

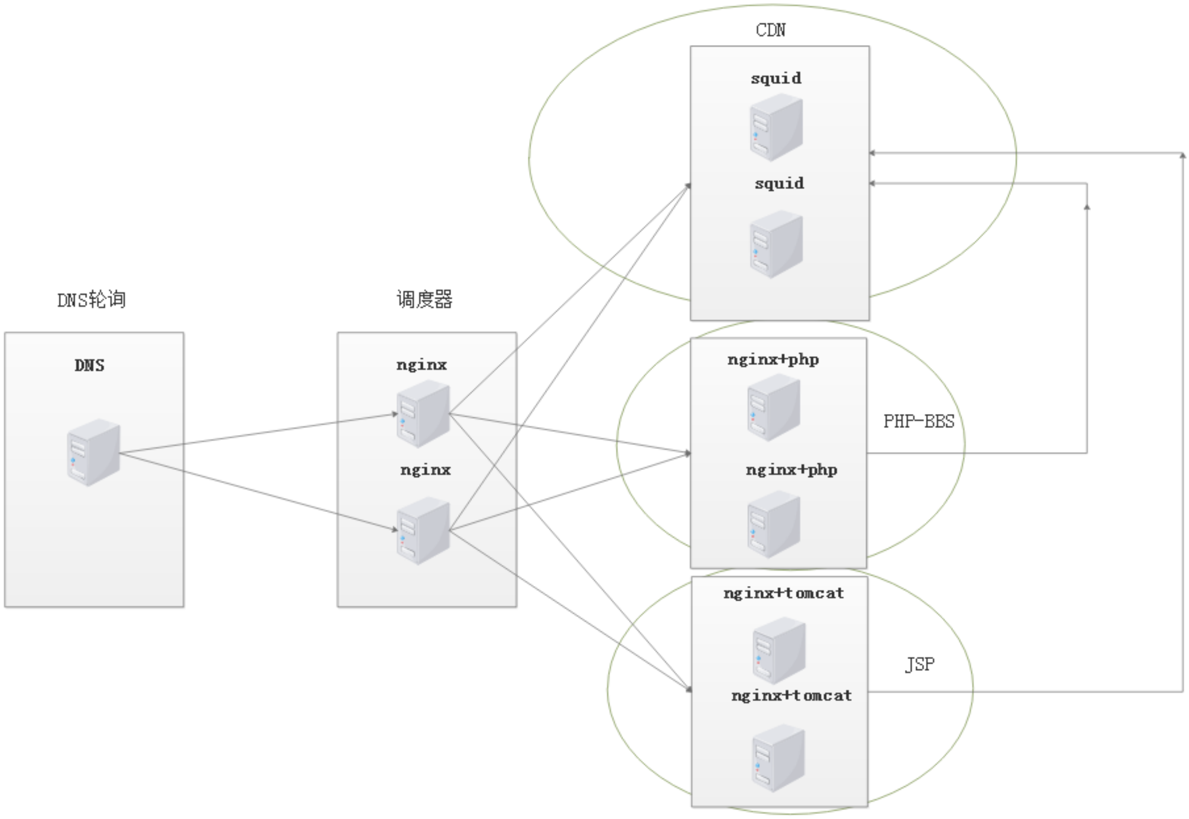
```
[root@serverb ~]# hostnamectl set-hostname nginx-server2
[root@nignx-server1 ~]# rsync -azvR /etc/nginx/ root@172.25.4.11:/ #同步文件给nginx-server2
[root@nignx-server1 ~]# ssh root@172.25.4.11 "systemctl start nginx"
```

测试：修改hosts文件，并停止nginx-server1的nginx服务

```
[root@foundation4 ~]# vim /etc/hosts
```

```
172.25.4.11 www.php-f4.com www.jsp-f4.com
```

测试 ok



12）部署 DNS

安装 DNS:

www.php-f4.com 172.25.4.10

www.jsp-f4.com 172.25.4.11

```
[root@foundation4 ~]# yum -y install bind

[root@foundation1 named]# vim /etc/named.conf
options {
listen-on port 53 { 127.0.0.1; any; };
listen-on-v6 port 53 { ::1; };
directory "/var/named";
dump-file "/var/named/data/cache_dump.db";
statistics-file "/var/named/data/named_stats.txt";
memstatistics-file "/var/named/data/named_mem_stats.txt";
allow-query { localhost; any; };
recursion yes;
dnssec-enable yes;
dnssec-validation yes;
bindkeys-file "/etc/named.iscdlv.key";
```

```
managed-keys-directory "/var/named/dynamic";
pid-file "/run/named/named.pid";
session-keyfile "/run/named/session.key";
};

logging {
channel default_debug {
file "data/named.run";
severity dynamic;
};
};

view "php" {
match-clients { 172.25.4.0/24; };
zone "." IN {
type hint;
file "named.ca";
};
zone "php-f4.com" IN {
type master;
file "php-f4.com.zone";
};
include "/etc/named.rfc1912.zones";
};

view "jsp" {
match-clients { 172.25.254.0/24; };
zone "." IN {
type hint;
file "named.ca";
};
zone "jsp-f4.com" IN {
type master;
file "jsp-f4.com.zone";
};
include "/etc/named.rfc1912.zones";
};
include "/etc/named.root.key";

[root@foundation4 ~]# vim /var/named/jsp-f4.com.zone
$TTL 1D
@ IN SOA ns1.jsp-f4.com. nsmail.jsp-f4.com. (
10 ; serial
1D ; refresh
```

```
1H ; retry
1W ; expire
3H ) ; minimum
@ NS ns1.jsp-f4.com.
ns1 A 172.25.254.4
www A 172.25.4.11
```

```
[root@foundation4 ~]# vim /var/named/php-f4.com.zone
$TTL 1D
@ IN SOA ns1.php-f4.com. nsmail.php-f4.com. (
10 ; serial
1D ; refresh
1H ; retry
1W ; expire
3H ) ; minimum
@ NS ns1.php-f4.com.
ns1 A 172.25.254.4
www A 172.25.4.10
```

```
[root@foundation4 ~]# systemctl start named
[root@foundation4 ~]# systemctl enable named
```