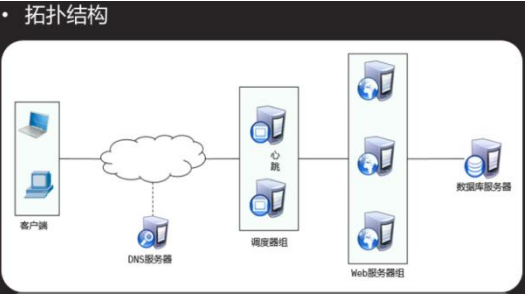


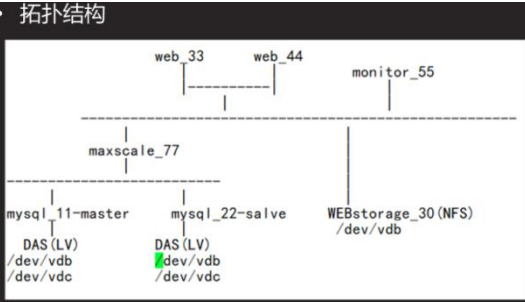
13_project201 项目概述+部署数据库服务 mysql+部署共享存储服务 NFS+配置网
站服务+测试配置+部署监控服务 zabbix

一 项目概述-项目介绍

1.1 项目一回顾



1.2 项目二拓扑



二项目概述-角色与技术

2.1 IP 地址及角色规划

主机名	IP地址	角色
mysql11	192.168.4.11	MySQL数据库服务器
mysql22	192.168.4.22	MySQL数据库服务器
maxscale77	192.168.4.77	读写分离服务器
nfs30	192.168.4.30	共享存储服务器
web33	192.168.4.33	网站服务器
web44	192.168.4.44	网站服务器
zabbix55	192.168.4.55	监控服务器

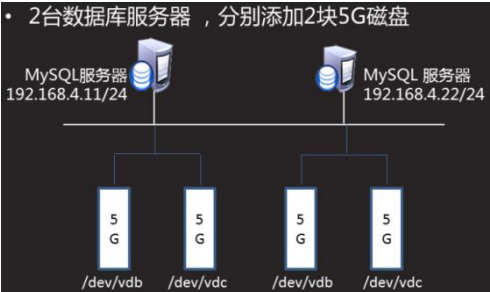
2.2 涉及技术

逻辑卷 数据库服务 MySQL 主从同步 数据读写分离: 写 11, 读 12

共享存储 nfs(存储网站页面数据) 监控服务

三 部署数据库服务 mysql-配置逻辑卷

3.1 拓扑图



3.2 添加磁盘(在第三 四 节中,11 和 22 的操作相同,此处只写 11 上的操作.)

按 3.1 给 192.168.4.11 和 192.168.4.22 各添加 2 块 5G 的磁盘

3.3 创建 lv(lv 卷便于数据存储卷的扩展)

3.3.1 分区,修改系统类型为 8e

`fdisk -l` 列出所有磁盘的分区情况

3.3.2 安装 lvm2

```
mysql11 ~]# yum -y install lvm2
```

3.3.3 创建 pv

```
mysql11 ~]# pvcreate /dev/vdb1 /dev/vdc1
```

```
mysql11 ~]# pvdisplay
```

3.3.4 创建 vg

```
mysql11 ~]# vgcreate vg0 /dev/vdb1 /dev/vdc1
```

```
mysql11 ~]# vgdisplay vg0
```

3.3.5 创建 lv

```
mysql11 ~]# lvcreate -n lv0 -L 9.99G vg0
```

```
mysql11 ~]# lvscan
```

3.3.6 格式化 lv 为 xfs

```
mysql11 ~]# mkfs.xfs /dev/vg0/lv0
```

```
mysql11 ~]# blkid /dev/vg0/lv0
```

```
/dev/vg0/lv0: UUID="6542f96a-2d0e-4874-abe0-41fbf9e04a73" TYPE="xfs"
```

四 部署数据库服务 mysql-配置数据库服务

4.1 安装 MySQL 软件

```
room9pc01 ~]$ scp -r /linux-soft/03/mysql/ root@192.168.4.11:/root
```

```
mysql11 ~]# cd mysql/
```

```
mysql11 mysql]# tar -xf mysql-5.7.17.tar
```

```
mysql11 mysql]# yum -y localinstall mysql-community-*
```

4.2 挂载 lv 分区

```
mysql11 ~]# blkid /dev/vg0/lv0
```

```
/dev/vg0/lv0: UUID="6542f96a-2d0e-4874-abe0-41fbf9e04a73" TYPE="xfs"
```

```
mysql11 ~]# vim /etc/fstab #设置 lv0 挂载到/var/lib/mysql 上
```

```
UUID="6542...4a73" /var/lib/mysql xfs defaults 0 0
```

```
mysql11 ~]# mount -a
```

```
mysql11 ~]# df -h /var/lib/mysql #显示结果为 10G
```

```
mysql11 ~]# mount | grep lv0
```

```
/dev/mapper/vg0-lv0 on /var/lib/mysql type xfs (rw,relatime,attr2,inode64,noquota)
```

4.3 启动服务

```
mysql11 ~]# systemctl restart mysqld
```

```
mysql11 ~]# systemctl enable mysqld
```

```
mysql11 ~]# ls /var/lib/mysql #此时目录下有内容了
```

```
mysql11 ~]# ss -antulp | grep :3306
```

4.4 管理员登录用户管理

```
mysql11 ~]# grep password /var/log/mysqld.log

mysql11 ~]# mysql -uroot -p 日志内密码

mysql> alter user root@"localhost" identified by "123qqq...A";

mysql> exit

mysql11 ~]# mysql -uroot -p123qqq...A
```

五 部署数据库服务 mysql-配置主从同步

5.1 配置主服务器 192.168.4.11

```
mysql11 ~]# vim /etc/my.cnf

[mysqld]

server_id=11

log_bin=master11

mysql11 ~]# systemctl restart mysqld

mysql11 ~]# mysql -uroot -p123qqq...A

mysql> grant replication slave on *.* to repluser@"%" identified by "123qqq...A";

mysql> show grants for repluser@"%";

mysql> show master status;

master11.000001      441
```

5.2 配置从服务器 192.168.4.22

```
mysql22 ~]# vim /etc/my.cnf
```

```
[mysqld]
```

```
server_id=22
```

```
mysql22 ~]# systemctl restart mysqld
```

```
mysql22 ~]# mysql -uroot -p123qqq...A
```

```
mysql> change master to
```

```
-> master_host="192.168.4.11",
```

```
-> master_user="repluser",
```

```
-> master_password="123qqq...A",
```

```
-> master_log_file="master11.000001",
```

```
-> master_log_pos=441;
```

```
mysql> start slave;
```

```
mysql> show slave status\G;
```

5.3 配置主机 192.168.4.50 测试主从功能

5.3.1 主服务器创建库、表及授权连接用户

```
11 mysql> create database projectdb;
```

```
11 mysql> create table projectdb.user(name char(10));
```

```
11 mysql> grant select,insert on projectdb.* to
```

```
admin@"%" identified by "123qqq...A";
```

5.3.2 主机 192.168.4.50 上使用授权连接用户登录 11 上的数据库,并写入数据

```
host50 ~]# mysql -h192.168.4.11 -uadmin -p123qqq...A
```

```
50 mysql> show databases;
```

```
50 mysql> use projectdb;
```

```
50 mysql> show tables;
```

```
50 mysql> insert into projectdb.user values("tom");
```

```
50 mysql> insert into projectdb.user values("jerry");
```

5.3.3 22 上使用 root 用户登录本机数据库,查看数据

```
mysql22 ~]# mysql -uroot -p123qqq...A
```

```
22 mysql> select * from projectdb.user;    #有数据 tom 和 jerry
```

六 部署数据库服务 mysql-配置读写分离服务

6.1 安装软件

```
room9pc01 ~]$ scp -r /linux-soft/03/mysql/ root@192.168.4.77:/root
```

```
maxscale77 ~]# cd mysql/
```

```
maxscale77 mysql]# rpm -ivh maxscale-2.1.2-1.rhel.7.x86_64.rpm
```

6.2 修改配置文件

```
maxscale77 ~]# ls /etc/maxscale.cnf    #maxscale 主配置文件
maxscale77 ~]# ls /var/log/maxscale/    #maxscale 日志目录

maxscale77 ~]# vim /etc/maxscale.cnf

9 [maxscale]

10 threads=auto    #服务线程数量

18 [server1]    #第 1 台数据库服务器
19 type=server
20 address=192.168.4.11
21 port=3306
22 protocol=MySQLBackend    #复制此 5 行, 粘贴到 24-28 行

24 [server2]    #第 2 台数据库服务器
25 type=server
26 address=192.168.4.22
27 port=3306
28 protocol=MySQLBackend
```


35 [MySQL Monitor] #定义监控服务器的监控内容及监控用户

36 type=monitor

37 module=mysqlmon

38 servers=server1,server2

39 user=maxscalemon #监控用户

40 passwd=123qqq...A #监控用户密码

41 monitor_interval=10000

52-58 85-89 注释掉

63 [Read-Write Service] #定义读写分离服务

64 type=service

65 router=readwritesplit

66 servers=server1,server2

67 user=maxscalrouter #路由用户

68 passwd=123qqq...A #路由用户密码

69 max_slave_connections=100%

75 [MaxAdmin Service] #定义管理服务

76 type=service

```
77 router=cli

91 [Read-Write Listener]    #定义读写分离服务端口
92 type=listener
93 service=Read-Write Service
94 protocol=MySQLClient
95 port=4006

97 [MaxAdmin Listener]    #定义管理服务端口
98 type=listener
99 service=MaxAdmin Service
100 protocol=maxscaled
101 socket=default
102 port=4016    #添加此行
```

6.3 配置数据库服务器

6.3.1 在 11 上用 root 用户登录数据库, 授权监控用户和路由用户

```
11 mysql> grant replication slave,replication client on *.* to
maxscalemon@"%" identified by "123qqq...A";    #授权监控用户

11 mysql> grant select on mysql.* to maxscalerouter@"%" identified
```

by "123qqq...A"; #授权路由用户

6.3.2 在 22 上用 root 用户登录数据库, 查看数据是否同步(正常情况应已同步)

```
mysql22 ~]# mysql -uroot -p123qqq...A
```

```
22 mysql> select user,host from mysql.user;
```

```
22 mysql> show grants for maxscalemon@"%";
```

```
22 mysql> show grants for maxscalerouter@"%";
```

6.4 在 maxscale 服务器本机启动 maxscale 读写分离服务, 查看进程及端口

```
maxscale77 ~]# maxscale -f /etc/maxscale.cnf
```

```
maxscale77 ~]# ps -C maxscale
```

PID	TTY	TIME	CMD
23134	?	00:00:00	maxscale

```
maxscale77 ~]# netstat -antulp | grep maxscale   #有 4006 和 4016
```

maxscale 日志文件: /var/log/maxscale/maxscale.log

6.5 在 maxscale 服务器本机访问管理服务, 查看监控信息

```
axscale77 ~]# maxadmin -uadmin -pmariadb -P4016
```

```
MaxScale> list servers
```

Server	Address	Port	Connections	Status
server1	192.168.4.11	3306	0	Master, Running
server2	192.168.4.22	3306	0	Slave, Running

6.6 读写分离测试(写 11, 读 22, 11 数据同步到 22)

6.6.1 使用授权连接用户 amdin 在 50 上登录 77 主机上数据库, 在 projectdb.user 中写入数据

```
host50 ~]# mysql -h192.168.4.77 -uadmin -p123qqq...A -P4006
```

```
host50 mysql> insert into projectdb.user values("natasha");
```

```
host50 mysql> select * from projectdb.user;
```

6.6.2 11 22 使用 root 用户登录数据库, 在 projectdb.user 中查看数据是否同步(正常情况为已同步)

```
mysql11 ~]# mysql -uroot -p123qqq...A
```

```
11 mysql> select * from projectdb.user;
```

```
mysql22 ~]# mysql -uroot -p123qqq...A
```

```
22 mysql> select * from projectdb.user;
```

6.6.3 使用授权连接用户 amdin 在 50 上登录 22 主机上数据库, 在 projectdb.user 中写入数据

```
host50 ~]# mysql -h192.168.4.22 -uadmin -p123qqq...A
```

```
host50 mysql> insert into projectdb.user values("harry");
```

```
host50 mysql> select * from projectdb.user;
```

6.6.4 使用授权连接用户 amdin 在 50 上登录 77 主机上数据库, 在 projectdb.user 中读取刚写入的数据 harry

```
host50 ~]# mysql -h192.168.4.77 -uadmin -p123qqq...A -P4006
```

```
host50 mysql> select * from projectdb.user;    #能读到数据 harry
```

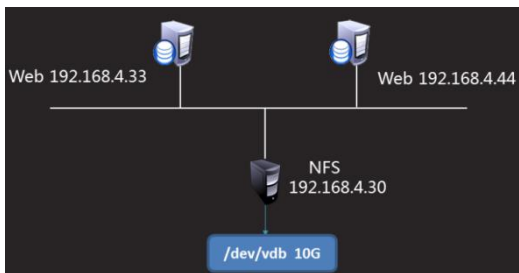
6.6.5 11 上使用 root 用户登录数据库, 查看 projectdb.user 是否有数据 harry

```
mysql11 ~]# mysql -uroot -p123qqq...A
```

```
11 mysql> select * from projectdb.user;    #无数据 harry
```

七 配置共享存储服务 NFS-准备磁盘

7.1 拓扑图



7.2 添加磁盘

kvm 虚拟机内为 nfs30 主机添加一块 10G 磁盘

7.3 挂载磁盘

7.3.1 将 10G 磁盘分区, 分为 1 个分区

```
nfs30 ~]# fdisk /dev/vdb    #将/dev/vdb 分区
```

```
nfs30 ~]# fdisk -l /dev/vdb    #查看分区结果
```

7.3.2 格式化分区/dev/vdb1 为 xfs, 查看格式化结果

```
nfs30 ~]# mkfs.xfs /dev/vdb1      #将/dev/vdb1 格式化为 xfs
```

```
nfs30 ~]# blkid /dev/vdb1          #查看格式化结果
```

```
/dev/vdb1: UUID="feb33a88-9088-4570-b88b-20870c72c6b5" TYPE="xfs"
```

7.3.3 配置开机挂载

```
nfs30 ~]# mkdir /sitedir          #创建挂载目录
```

```
nfs30 ~]# vim /etc/fstab           #编辑开启挂载文件
```

```
UUID="feb3...c6b5" /sitedir xfs defaults 0 0
```

```
nfs30 ~]# mount -a                #挂载
```

```
nfs30 ~]# df -h | grep sitedir    #挂载检查
```

```
/dev/vdb1          10G   33M   10G    1% /sitedir
```

八 配置共享存储服务 NFS-配置 NFS 服务

8.1 安装软件

```
nfs30 ~]# yum -y install nfs-utils rpcbind
```

8.2 修改 nfs 服务配置文件

```
nfs30 ~]# vim /etc/exports
```

```
/sitedir *(rw)  # *表示共享给所有主机, (rw)表示被共享的权限
```

```
nfs30 ~]# chmod o+w /sitedir/    #赋予其他人的写权限
```

```
nfs30 ~]# exportfs -r            #加载配置,无提示则表示成功
```

8.3 启动服务

```
nfs30 ~]# systemctl start rpcbind
nfs30 ~]# systemctl enable rpcbind
nfs30 ~]# systemctl start nfs
nfs30 ~]# systemctl enable nfs
```

8.4 查看共享信息

```
nfs30 ~]# showmount -e localhost
```

Export list for localhost:

```
/sitedir *
```

九 配置完整服务-运行 httpd 服务(在第九节中,33 和 44 的操作相同,此处只写 33 上的操作.)

9.1 安装 httpd 软件

```
web33 ~]# yum -y install httpd    #仅安装,千万别启动 httpd 服务
```

9.2 挂载共享存储

```
web33 ~]# yum -y install nfs-utils    #安装 nfs 软件包
```

```
web33 ~]# showmount -e 192.168.4.30    #发现 192.168.4.30 的共享存储
```

Export list for 192.168.4.30:

```
/sitedir *
```

```
web33 ~]# vim /etc/fstab    #修改 fstab 文件,设置自动挂载,并检测
192.168.4.30:/sitedir /var/www/html nfs defaults 0 0

web33 ~]# mount -a

web33 ~]# df -h | grep sitedir

192.168.4.30:/sitedir    10G    32M    10G    1% /var/www/html
```

9.3 启动服务

```
web33 ~]# systemctl start httpd

web33 ~]# systemctl enable remote-fs.target

        #设置远程挂载服务开机运行

web33 ~]# systemctl enable httpd
```

9.4 查看挂载信息

```
web33 ~]# mount | grep "/var/www/html"

192.168.4.30:/sitedir      on      /var/www/html      type      nfs4
(rw,relatime,vers=4.1,rsz=262144,wsz=262144,namlen=255,hard,proto=tcp,port=0,timeo=600,retrans=2,sec=sys,clientaddr=192.168.4.33,local_lock=none,addr=192.168.4.30)
```


十 测试配置-测试 NFS 服务

10.1 编写网页文件

在 NFS 服务器创建测试页面 test.html

```
nfs30 ~]# echo "web test page" > /sitedir/test.html
```

10.2 真机通过 192.168.4.33 192.168.4.44 访问测试页面 test.html

```
room9pc01 ~]$ curl http://192.168.4.33/test.html
```

web test page

```
room9pc01 ~]$ curl http://192.168.4.44/test.html
```

web test page

十一 测试配置-测试 MySQL 服务

见 5.3 的主从同步测试和 6.6 的读写分离测试

十二 部署监控服务 zabbix-部署监控服务

12.1 部署运行环境 LNMP

12.1.1 部署 nginx(只部署,不启动)

```
room9pc01 ~]$ scp -r /linux-soft/02/lnmp_soft.tar.gz root@192.168.4.55:/root
```

```
zabbix55 ~]# tar -xf lnmp_soft.tar.gz
```

```
zabbix55 ~]# cd lnmp_soft/
```

```
zabbix55 lnmp_soft]# tar -xf nginx-1.12.2.tar.gz
```

```
zabbix55 lnmp_soft]# cd nginx-1.12.2/
zabbix55 nginx-1.12.2]# yum -y install gcc pcre-devel openssl-devel
zabbix55 nginx-1.12.2]# ./configure --with-http_ssl_module
zabbix55 nginx-1.12.2]# make && make install
```

12.1.2 部署 mariadb(只部署,不启动)

```
zabbix55 ~]# yum -y install mariadb mariadb-server mariadb-devel
```

12.1.3 部署 php(只部署,不启动)

```
zabbix55 ~]# yum -y install php php-fpm php-mysql
```

12.1.4 修改 nginx 配置文件,设置 nginx 支持 PHP 动态网站

```
zabbix55 ~]# vim /usr/local/nginx/conf/nginx.conf

20 fastcgi_buffers 8 16k;          #缓存 php 生成的页面内容, 8 个 16k
21 fastcgi_buffer_size 32k;        #缓存 php 生产的头部信息
22 fastcgi_connect_timeout 300;    #连接 PHP 的超时时间
23 fastcgi_send_timeout 300;       #发送请求的超时时间
24 fastcgi_read_timeout 300;       #读取请求的超时时间

69     location ~ \.php$ {
70         root                html;
71         fastcgi_pass         127.0.0.1:9000;
72         fastcgi_index        index.php;
73         # fastcgi...script_name;
```

```
74         include         fastcgi.conf;
75     }
```

12.1.5 启动 nginx php-fpm mariadb 服务,关闭 selinux 与防火墙

```
zabbix55 ~]# systemctl restart mariadb
zabbix55 ~]# systemctl restart php-fpm
zabbix55 ~]# /usr/local/nginx/sbin/nginx
zabbix55 ~]# ss -antulp | grep :3306
zabbix55 ~]# ss -antulp | grep :9000
zabbix55 ~]# ss -antulp | grep :80
```

12.1.6 创建 PHP 测试页面,浏览器访问页面测试网页联通性

```
zabbix55 ~]# vim /usr/local/nginx/html/test.php
```

```
<?php
```

```
$i=33;
```

```
echo $i;
```

```
?>
```

```
zabbix55 ~]# curl http://192.168.4.55/test.php
```

```
33
```

12.2 安装 zabbix 软件

```
zabbix55 ~]# yum -y install net-snmp-devel curl-devel #依赖包
```

```
zabbix55 ~]# yum -y install libevent-devel-2.0.21-4.el7.x86_64.rpm

zabbix55 ~]# cd lnmp_soft/

zabbix55 lnmp_soft]# tar -xf zabbix-3.4.4.tar.gz

zabbix55 lnmp_soft]# cd zabbix-3.4.4/

zabbix55 zabbix-3.4.4]# ./configure \

> --enable-server \      #安装部署 Zabbix 监控服务器端软件

> --enable-proxy \      #安装部署 Zabbix 代理相关软件

> --enable-agent \      #安装部署 Zabbix 监控主机端软件

> --with-mysql=/usr/bin/mysql_config \ #配置 mysql_config 路径

> --with-net-snmp \      #允许 Zabbix 通过 snmp 协议监控其他设备

> --with-libcurl #安装相关 curl 库文件,这样 Zabbix 就可以通过 curl 连接

http 等服务,测试被监控主机服务的状态

zabbix55 zabbix-3.4.4]# make && make install
```

12.3 初始化配置

12.3.1 创建数据库

```
zabbix55 ~]# mysql

MariaDB [(none)]> create database zabbix character set utf8;

MariaDB [(none)]> grant all on zabbix.* to zabbix@"localhost"

identified by "zabbix";
```

```
zabbix55 ~]# cd /root/lnmp_soft/zabbix-3.4.4/database/mysql
zabbix55 mysql]# mysql -uzabbix -pzabbix zabbix < schema.sql
zabbix55 mysql]# mysql -uzabbix -pzabbix zabbix < images.sql
zabbix55 mysql]# mysql -uzabbix -pzabbix zabbix < data.sql
```

12.3.2 安装依赖

```
zabbix55 ~]#yum -y install
php-gd php-xml php-bcmath php-mbstring php-ldap
```

12.3.3 修改 php 程序运行参数

```
zabbix55 ~]# vim /etc/php.ini

878 date.timezone = Asia/Shanghai      #解除注释并修改,设置时区
384 max_execution_time = 300           #最大执行时间,秒
672 post_max_size = 32M                #POST 数据最大容量
394 max_input_time = 300                #服务器接收数据的时间限制

zabbix55 ~]# systemctl restart php-fpm
```

12.4 修改配置文件

```
zabbix55 ~]# vim /usr/local/etc/zabbix_server.conf

85 DBHost=localhost                    #解除该行注释,使用默认配置
95 DBName=zabbix                       #修改为 MySQL 中创建的库名
111 DBUser=zabbix                      #修改为 MySQL 中授权的用户名
```

119 DBPassword=zabbix #修改为 MySQL 中授权的用户名的密码

38 LogFile=/tmp/zabbix_server.log #默认

12.5 启动服务,检测端口 10051

```
zabbix55 ~]# useradd -s /sbin/nologin zabbix
```

```
zabbix55 ~]# zabbix_server
```

```
zabbix55 ~]# ss -antulp | grep zabbix_server #有端口 10051
```

十三 部署监控服务 zabbix-配置客户端(192.168.4.33 和 192.168.4.44)

13.1 安装 zabbix 软件(33 和 44 操作一致)

```
room9pc01 ~]$ scp /linux-soft/02/lnmp_soft.tar.gz root@192.168.4.33:/root/
```

```
web33 ~]# yum -y install gcc pcre-devel
```

```
web33 ~]# tar -xf lnmp_soft.tar.gz
```

```
web33 ~]# cd lnmp_soft/
```

```
web33 lnmp_soft]# tar -xf zabbix-3.4.4.tar.gz
```

```
web33 lnmp_soft]# cd zabbix-3.4.4/
```

```
web33 zabbix-3.4.4]# ./configure --enable-agent
```

```
web33 zabbix-3.4.4]# make && make install
```

13.2 修改配置文件(33 和 44 配置不同)

```
web33 ~]# vim /usr/local/etc/zabbix_agentd.conf
```

```
93 Server=127.0.0.1,192.168.4.55
134 ServerActive=127.0.0.1,192.168.4.55
145 Hostname=web33
30 LogFile=/tmp/zabbix_agentd.log
web44 ~]# vim /usr/local/etc/zabbix_agentd.conf
93 Server=127.0.0.1,192.168.4.55
134 ServerActive=127.0.0.1,192.168.4.55
145 Hostname=web44
30 LogFile=/tmp/zabbix_agentd.log
```

13.3 启动服务(33 和 44 操作一致)

```
web33 ~]# useradd zabbix
web33 ~]# zabbix_agentd
web33 ~]# ss -antulp | grep :10050
```

13.4 上线页面

```
room9pc01 ~]$ scp -r /linux-soft/03/Zabbix/ root@192.168.4.55:/root
zabbix55 ~]# cd Zabbix/
zabbix55 Zabbix]# tar -xf zabbix-3.4.4.tar.gz
zabbix55 ~]# cd Zabbix/zabbix-3.4.4/frontends/php/
zabbix55 php]# cp -r * /usr/local/nginx/html/
zabbix55 ~]# chmod -R 777 /usr/local/nginx/html/*
```

十四 部署监控服务 zabbix- 监控配置

14.1 创建主动模式监控模板

14.2 配置自动发现

14.3 查看监控信息

步骤一：创建主动模式监控模板

1) 克隆监控模板，新模板名称为：Template OS Linux ServerActive。如图 -3 所示



图 -3

2) 修改模板中的监控项目的监控模式为主动模式，如图 -4 所示



3) 禁用不支持主动模式的监控项目如图-5 所示。

触发器	键值	间隔	历史记录	趋势	类型 ▲	应用集	状态
触发器 1	agent.version	1h	1w		Zabbix 客户端	Zabbix agent	停用的
触发器 1	agent.hostname	1h	1w		Zabbix 客户端	Zabbix agent	停用的
触发器 1	agent.ping	1m	1w	365d	Zabbix 客户端	Zabbix agent	停用的
触发器 1	kernel.maxproc	1h	1w	365d	Zabbix客户端(主动式)	OS	已启用

图-5

步骤二：配置自动发现

1) 创建自动发现规则

通过 Configuration (配置) -->Discovery (自动发现) -->Create discovery rule (创建发现规则)，如图-6 所示。



图-6

2) 填写规则

填写自动发现的 **IP 范围**（逗号隔开可以写多个），多久做一次自动发现（默认为 1 小时，仅实验修改为 1m），如图-7 所示。配置检查的方式：**Ping**、**HTTP**、**FTP**、**Agent** 的自定义 **key** 等检查，如图-8 所示。

自动发现规则

名称	Web_discovery_rule
由agent代理程序自动发现	没有agent代理程序
IP范围	192.168.2.0-254
更新间隔	1m

图-7

更新间隔	1m
检查	ICMP ping 编辑 移除
	新的
	检查类型 ICMP ping
	添加 取消

图-8

3) 创建 Action 动作

通过 Configuration (配置) --> Actions Event source(事件源): 自动发现 (Discovery) --> Create action (创建动作), 如图-9 所示。



图-9

4) 配置 Action 动作具体行为

配置动作, 添加动作名称, 添加触发动作的条件, 如图-10 所示。



图-10

点击操作（触发动作后要执行的操作指令），操作细节：添加主机到组，与模板链接（HTT
如图-11 所示。

动作

操作

默认接收人

Discovery: {DISCOVERY.DEVICE STATUS} {DISCOVER

默认信息

Discovery rule: {DISCOVERY.RULE NAME}
Device IP: {DISCOVERY.DEVICE IPADDRESS}
Device DNS: {DISCOVERY.DEVICE.DNS}
Device status: {DISCOVERY.DEVICE STATUS}
Device uptime: {DISCOVERY.DEVICE.UPTIME}
Device service name: {DISCOVERY.SERVICE NAME}

操作

细节

添加到主机群组: Linux servers

链接到模板: Template App HTTP Service

操作细节

操作类型

添加到主机群组

主机群组

Linux servers

在此输入搜索

添加

取消

更新

克隆

删除

取消

图-11

步骤三：查看监控信息

1) 验证监控效果，如图-12 所示



图-12

重启顺序

11->22->77->30->33->44->55