

# 自动化远程备份 Mysql+Web 数据加备份服务检测，为 blog 数据保驾护航

作者: [KangKang-Wang](#)

原文链接: <https://ld246.com/article/1566919218334>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

之前都没有备份blog数据的习惯，而且blog更多时候是闲着的，甚至自己很久都没有打开过。有一次务器过期了很久才发现，最后已经为时已晚了。blog虽小，但数据珍贵。这次闲下来写了一份远程备份Mysql+Web脚本和一份备份服务检测脚本，可以大大提升数据的安全性！

系统环境：

Server 端:Centos 7.6

远程备份端：群辉NAS(已经配置好DDNS)

适用站点：**个人blog / 数据量不太大的站点**

如果没有群辉的话也没有关系，有另外一台单独的linux机器也是可以的，只不过后面的提醒会比较麻烦一点，要配置好邮件服务，能向外发送邮件，遇到异常可以第一时间处置，不至于像我之前一样丢失数据。

Server端脚本：

```
#!/bin/bash
NOW_DATE=$(date +%Y%m%d_%H%M%S)
MYSQL_FILENAME="solo_${NOW_DATE}.sql" #mysql导出的文件名
WEB_FILENAME="web_backup${NOW_DATE}.tar.gz" #web打包备份后的文件名
mysqldump -uroot -p password solo > /data/backup/Mysqldump/${MYSQL_FILENAME}

HOST=nas_domain #我的NAS域名，或者你的备份端地址
USER=user #ftp账号
PASSWORD=password #ftp密码
LOCAL_MYSQL_PATH=/data/backup/Mysqldump #Server端mysql备份文件存放目录
LOCAL_WEB_BACKUP_PATH=/data/backup/Web_backup #Server端web打包文件存放目录
WEB_PATH=/data/www/wwwroot #Webroot路径
REMOTE_MYSQL_PATH=/Server_backup/Mysql/ #备份端存SQL文件放路径
REMOTE_WEB_PATH=/Server_backup/Web/ #备份端Web文件存放路径

/usr/bin/ftp -vn <<EOF
open ${HOST}
user ${USER} ${PASSWORD}
binary
prompt
cd ${REMOTE_MYSQL_PATH}
lcd ${LOCAL_MYSQL_PATH}
put ${MYSQL_FILENAME}
bye
EOF

find ${LOCAL_MYSQL_PATH} -mtime +10 -name "*.sql" | xargs rm -f #删除Server端老旧的备份文件

tar -czf /${LOCAL_WEB_BACKUP_PATH}/${WEB_FILENAME} -C /data/www/wwwroot/ webroot
#这样打包出来的文件不含绝对路径，这样解压也比较安全
/usr/bin/ftp -vn <<EOF
open ${HOST}
user ${USER} ${PASSWORD}
binary
prompt
cd ${REMOTE_WEB_PATH}
lcd ${LOCAL_WEB_BACKUP_PATH}
```

```
put ${WEB_FILENAME}
bye
EOF
```

```
find ${LOCAL_WEB_PATH} -mtime +10 -name "*.gz" | xargs rm -f    #删除Server端老旧的We
备份文件
```

这个脚本本身不适用于数据量大的站点，第一是数据量大可能会有文件损坏的情况发生，第二是这个本每次都是全量备份，占据的空间量大。大型的站点应该采用更可靠的传输方式以及一次全量备份多增量备份的形式，更加稳妥。

以上脚本利用crontab定时任务来执行，`crontab -l`查看所有的定时任务；`crontab -e`编辑定时任务我的是`0 0,12 * * *`，也就是每天0点和中午12点准点执行一次。

检测备份服务脚本(在备份端执行)：

```
#!/bin/ash
declare -i MYSQL_FILES=$(find /volume1/Server_backup/Mysql -type f -mtime -1|wc -l)
declare -i WEB_FILES=$(find /volume1/Server_backup/Web -type f -mtime -1|wc -l)
declare -i TODAY_FILE_NUMS=${MYSQL_FILES}+${WEB_FILES}
```

```
case ${TODAY_FILE_NUMS} in
0)
    echo "Web服务器备份服务不可用，请检查！"
    exit 1
    ;;
1)
    echo "Web服务器备份服务不可用，请检查！"
    exit 1
    ;;
2)
    echo "Web服务器备份服务疑似不可用，请检查！"
    exit 1
    ;;
3)
    echo "接收到的Web服务器备份文件(Mysql与Web)数量不对齐，请检查！"
    exit 1
    ;;
4)
    echo "今日收到备份文件数量正确，服务端正常可用。"
esac
exit 0
```

原理很简单，只要检查24小时内收到的文件数量对齐，说明备份服务正常可用。反之则不可用。

因为群辉自带的定时任务功能可以将执行输出发送邮件到我的邮箱，所以这里免去了很多麻烦。如果的是其他Linux机器，可能需要手动配置好邮件服务，上面的脚本也需要对应修改。

清理老旧的备份文件,避免存储空间的浪费：

```
#!/bin/ash
MYSQL_PATH=/volume1/Server_backup/Mysql
WEB_PATH=/volume1/Server_backup/Web
```

```
find ${MYSQL_PATH} -mtime +7 -name "*.sql" | xargs rm -f  
find ${WEB_PATH} -mtime +7 -name "*.gz" | xargs rm -f
```