

zabbix 4.4.7 监控 redis

作者: [Leif160519](#)

原文链接: <https://ld246.com/article/1585880545206>

来源网站: [链滴](#)

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



1. 创建redis监控脚本

1.1 创建脚本

在`/etc/zabbix/zabbix_agentd.d/`下新建文件`redis_status.sh`,内容如下:

```
#!/bin/bash
REDISCLI="/usr/bin/redis-cli"
HOST="127.0.0.1"
PORT=6379
PASS=""

if [[ $# == 1 ]];then
    case $1 in
        version)
            result=`$REDISCLI -h $HOST -p $PORT info server | grep -w "redis_version" | awk -F':' '{print $2}'`"echo $result
        ;;
        uptime)
            result=`$REDISCLI -h $HOST -p $PORT info server | grep -w "uptime_in_seconds" | awk -F':' '{print $2}'`"echo $result
        ;;
        connected_clients)
            result=`$REDISCLI -h $HOST -p $PORT info clients | grep -w "connected_clients" | awk -F':' '{print $2}'`"echo $result
        ;;
        blocked_clients)
```

```

result=`$REDISCLI -h $HOST -p $PORT info clients | grep -w "blocked_clients" | awk -F'[^[:space:]]+' '{print $2}'` 
echo $result
;;
used_memory)
result=`$REDISCLI -h $HOST -p $PORT info memory | grep -w "used_memory" | awk -F'[^[:space:]]+' '{print $2}'` 
echo $result
;;
used_memory_rss)
result=`$REDISCLI -h $HOST -p $PORT info memory | grep -w "used_memory_rss" | awk -F'[^[:space:]]+' '{print $2}'` 
echo $result
;;
used_memory_peak)
result=`$REDISCLI -h $HOST -p $PORT info memory | grep -w "used_memory_peak" | awk -F'[^[:space:]]+' '{print $2}'` 
echo $result
;;
used_memory_lua)
result=`$REDISCLI -h $HOST -p $PORT info memory | grep -w "used_memory_lua" | awk -F'[^[:space:]]+' '{print $2}'` 
echo $result
;;
used_cpu_sys)
result=`$REDISCLI -h $HOST -p $PORT info cpu | grep -w "used_cpu_sys" | awk -F'[^[:space:]]+' '{print $2}'` 
echo $result
;;
used_cpu_user)
result=`$REDISCLI -h $HOST -p $PORT info cpu | grep -w "used_cpu_user" | awk -F'[^[:space:]]+' '{print $2}'` 
echo $result
;;
used_cpu_sys_children)
result=`$REDISCLI -h $HOST -p $PORT info cpu | grep -w "used_cpu_sys_children" | awk -F'[^[:space:]]+' '{print $2}'` 
echo $result
;;
used_cpu_user_children)
result=`$REDISCLI -h $HOST -p $PORT info cpu | grep -w "used_cpu_user_children" | awk -F'[^[:space:]]+' '{print $2}'` 
echo $result
;;
rdb_last_bgsave_status)
result=`$REDISCLI -h $HOST -p $PORT info Persistence | grep -w "rdb_last_bgsave_status" | awk -F'[^[:space:]]+' '{print $2}' | grep -c ok` 
echo $result
;;
aof_last_bgrewrite_status)
result=`$REDISCLI -h $HOST -p $PORT info Persistence | grep -w "aof_last_bgrewrite_status" | awk -F'[^[:space:]]+' '{print $2}' | grep -c ok` 
echo $result
;;

```

```

aof_last_write_status)
    result=`$REDISCLI -h $HOST -p $PORT info Persistence | grep -w "aof_last_write_status" | awk -F': '{print $2}' | grep -c ok`
    echo $result
    ;;
*)
    echo -e "\033[33mUsage: $0 {connected_clients|blocked_clients|used_memory|used_memory_rss|used_memory_peak|used_memory_lua|used_cpu_sys|used_cpu_user|used_cpu_sys_children|used_cpu_user_children|rdb_last_bgsave_status|aof_last_bgrewrite_status|aof_last_write_status}\033[0m"
    ;;
esac
elif [[ $# == 2 ]];then
    case $2 in
        keys)
            result=`$REDISCLI -h $HOST -p $PORT info | grep -w "$1" | grep -w "keys" | awk -F'=|' '{print $2}'`
            echo $result
            ;;
        expires)
            result=`$REDISCLI -h $HOST -p $PORT info | grep -w "$1" | grep -w "keys" | awk -F'=|' '{print $4}'`
            echo $result
            ;;
        avg_ttl)
            result=`$REDISCLI -h $HOST -p $PORT info | grep -w "$1" | grep -w "avg_ttl" | awk -F'=|' '{print $6}'`
            echo $result
            ;;
        *)
            echo -e "\033[33mUsage: $0 {db0 keys|db0 expires|db0 avg_ttl}\033[0m"
            ;;
esac
fi

```

注意：若redis有密码，则需要在\$HOST后加上-a \$PASS参数。

1.2 赋予脚本可执行权限

```
chmod +x /etc/zabbix/zabbix_agentd.d/redis_status.sh
```

1.3 脚本测试

```
[root@test-server zabbix_agentd.d]# bash redis_status.sh version
3.2.12
```

2. 创建redis监控配置文件

2.1 创建文件

在 /etc/zabbix/zabbix_agentd.d下新建redis.conf文件，内容如下：

```
UserParameter=Redis.Status,status=`/usr/local/bin/redis-cli -h 127.0.0.1 -p 6379 ping|grep -c ONG` &&echo $status  
UserParameter=Redis.Info[*],/etc/zabbix/zabbix_agentd.d/redis_status.sh $1 $2
```

2.2 重启zabbix-agent:

```
systemctl restart zabbix-agent
```

3. 创建并导入监控模板

3.1 创建监控模板

redis-template.xml文件内容参考[github](#):

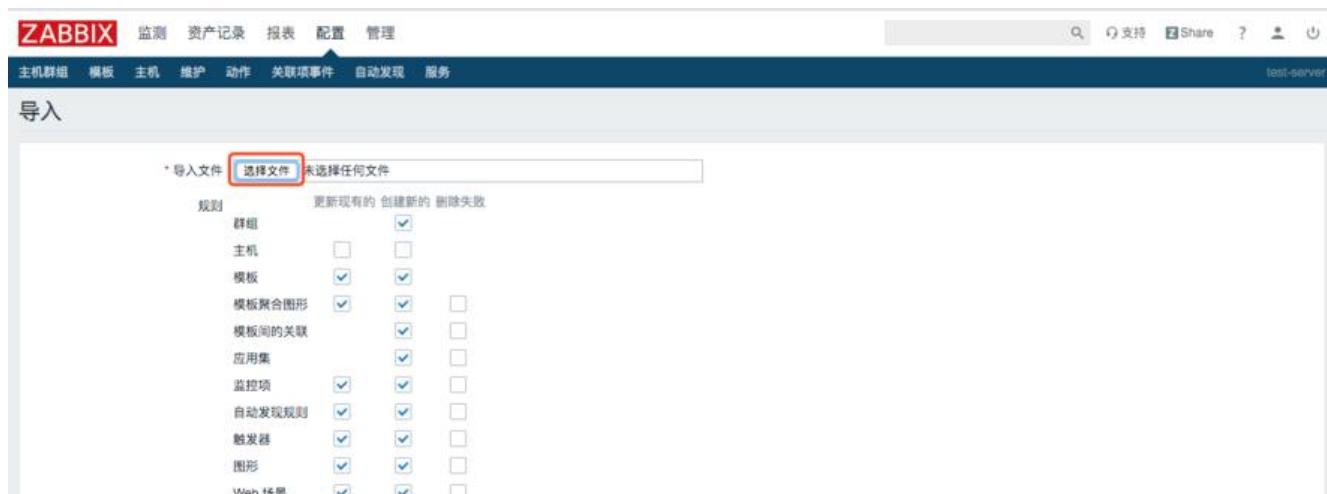
3.2 导入监控模板（不用zabbix自带的redis模板）

配置-模板-导入



The screenshot shows the Zabbix configuration interface. The top navigation bar has tabs for 'Monitoring', 'Assets', 'Reports', 'Configuration' (highlighted with a red box), and 'Management'. Below the navigation is a secondary menu with 'Host Groups' (highlighted with a red box), 'Templates' (highlighted with a red box), 'Hosts', 'Maintenance', 'Actions', 'Associated Events', 'Discovery', and 'Services'. A search bar and a 'Share' button are at the top right. The main area is titled 'Templates' (highlighted with a red box). It contains a search bar, a 'Create Template' button, and an 'Import' button (highlighted with a red box). Below these are sections for 'Name' (with a search input and 'Add' button), 'Linked Templates' (with a search input and 'Select' button), and 'Apply' and 'Reset' buttons. A large table lists existing templates: 'RedisMonitor' (5 applications, 19 items, 1 trigger, 5 graphs), 'Template App Apache by HTTP' (2 applications, 22 items, 5 triggers, 3 graphs, 1 dashboard), and 'Template App Apache by Zabbix agent' (2 applications, 26 items, 6 triggers, 4 graphs, 1 dashboard). Each row includes columns for 'Name', 'Applications', 'Items', 'Triggers', 'Graphs', 'Discovery', 'Web Monitoring', 'Linked Templates', and 'Tags'.

点击选择文件，找到redis-template.xml文件，将其导入



The screenshot shows the Zabbix import dialog. The top navigation bar is identical to the previous screenshot. The main area is titled 'Import' (highlighted with a red box). It has a 'Select File' button (highlighted with a red box) and a 'File' input field showing '未选择任何文件'. Below this is a 'Rules' section with checkboxes for various entities: '群组' (Group), '主机' (Host), '模板' (Template), '模板聚合图形' (Template Dashboard), '模板间的关联' (Template Association), '应用集' (Application Set), '监控项' (Item), '自动发现规则' (Discovery Rule), '触发器' (Trigger), '图形' (Graph), and 'Web 场景' (Web Scenario). Most checkboxes are checked, except for '主机' (Host) which is unchecked.



4.给主机添加监控模板

The screenshot shows the Zabbix configuration interface. The top navigation bar has tabs for 'ZABBIX', '监测', '资产记录', '报表', '配置' (highlighted with a red box), and '管理'. Below this, a secondary navigation bar has tabs for '主机群组', '模板', '主机' (highlighted with a red box), '维护', '动作', '关联项事件', '自动发现', and '服务'. The main content area is titled '主机' (Step 2). It contains fields for '名称' (Name), '要监控什么' (What to monitor), '模板' (Template), 'agent代理程序' (Agent program), 'DNS', 'IP地址' (IP address), and '端口' (Port). There are also '应用' (Apply) and '重设' (Reset) buttons. Below this is a table listing hosts. The first row, 'Zabbix server' (Step 3), is selected and highlighted with a red box. The table columns include '名称' (Name), '应用集' (Application set), '监控项' (Metrics), '触发器' (Triggers), '图形' (Graphs), '自动发现' (Auto-discovery), 'Web监测' (Web monitoring), '接口' (Interface), 'agent代理程序' (Agent program), '模板' (Template), '状态' (Status), '可用性' (Availability), 'agent加密' (Agent encryption), '信息' (Information), and '标记' (Tags). The 'Zabbix server' row shows details: '127.0.0.1:10050', 'RedisMonitor, Template App Zabbix Server, Template DB MySQL, Template OS Linux by Zabbix agent (Template Module Linux block devices by Zabbix agent, Template Module Linux CPU by Zabbix agent, Template Module Linux filesystems by Zabbix agent, Template Module Linux generic by Zabbix agent, Template Module Linux memory by Zabbix agent, Template Module Linux network interfaces by Zabbix agent, Template Module Zabbix agent)', '已启' (Enabled), 'ZBX|SNMP|JMX|IPMI' (Monitoring protocols), and '无' (None).

监控效果如图：



5.参考

[线上zabbix监控redis和redis集群](#)

有兴趣的可以尝试[官方提供的redis监控方式](#), [github地址](#)