## 2、资源监控

### 2.1、监控Tomcat服务

使用jmeter的tomcat监视器功能，可以通过向tomcat的status页面发送get请求，得到资源使用信息，然后转换为只直观的图像方式，这样的话，就可以监视到服务器的资源使用情况，不过需要注意的是，要使用tomcat监视器功能，就必须在要监视的服务器上装有tomcat。

1、配置/conf/tomcat-users.xml，内容如下

<?xml version='1.0' encoding='utf-8'?>  
 <tomcat-users> **<role rolename="manager-gui"/>  
  <role rolename="manager-script"/>**

**<role rolename="manager-jmx"/>**

**<role rolename="manager-status"/>  
  <user username="admin" password="admin" roles="manager-gui"/>**

**<user username="admin" password="admin" roles="manager-script"/>**

**<user username="admin" password="admin" roles="manager-jmx"/>**

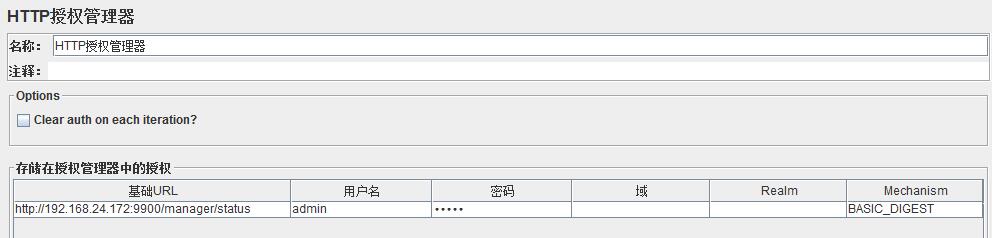
**<user username="admin" password="admin" roles="manager-status"/>** </tomcat-users>

2、配置/conf/context.xml，内容如下

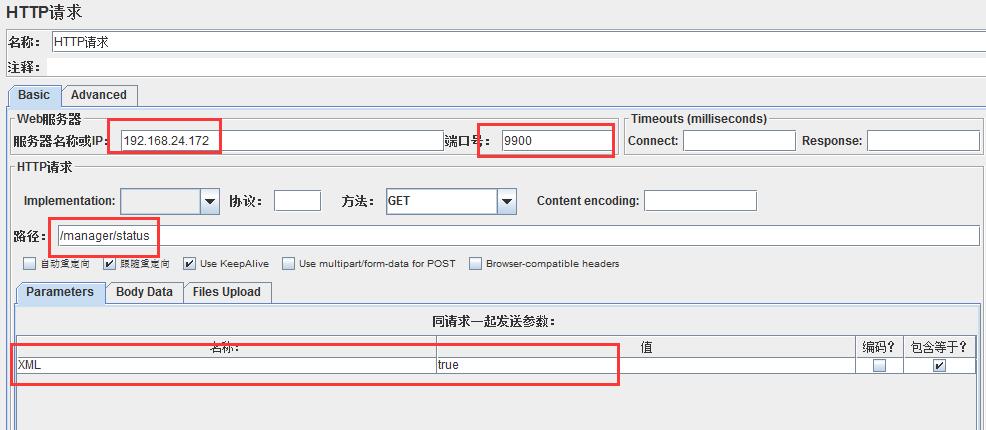
<Context>   
<WatchedResource>WEB-INF/web.xml</WatchedResource>     **<Manager pathname="/manager" debug="0" privileged="true" docBase="/home/tomcat-9900/webapps/manager" />      
     <Valve className="org.apache.catalina.valves.CometConnectionManagerValve" />** </Context>

**注意：配置完成后先访问http://192.168.24.172:9900/manager/status，检查是否配置成功。**

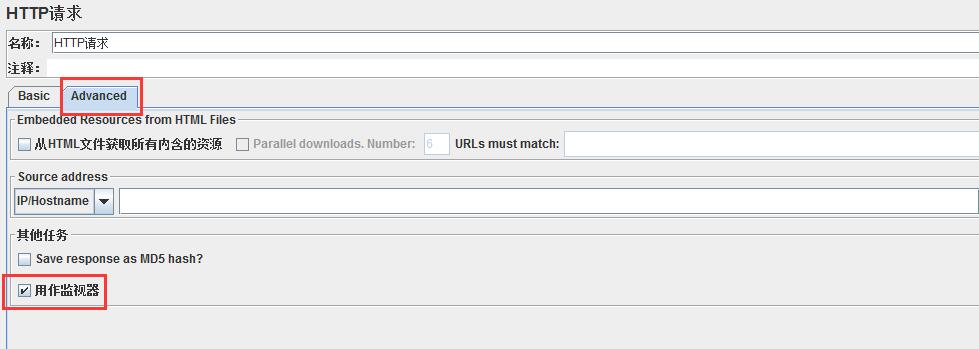
3、打开Jmeter在线程组下添加“**配置元件-HTTP授权管理器”**，在HTTP授权管理的“基础URL”中输入http://192.168.24.172:9900/manager/status，用户名为上面新建的admin，密码是admin，其他默认，如下图所示



4、接着在当前线程组下添加“**HTTP请求**”，输入被检测服务器中的TOMCAT访问IP和端口号；路径输入 /manager/status；请求参数输入XML（必须大写），值为true，如下图A1所示并且将此HTTP请求设置为“用作监视器”，如下图A2所示



图A1

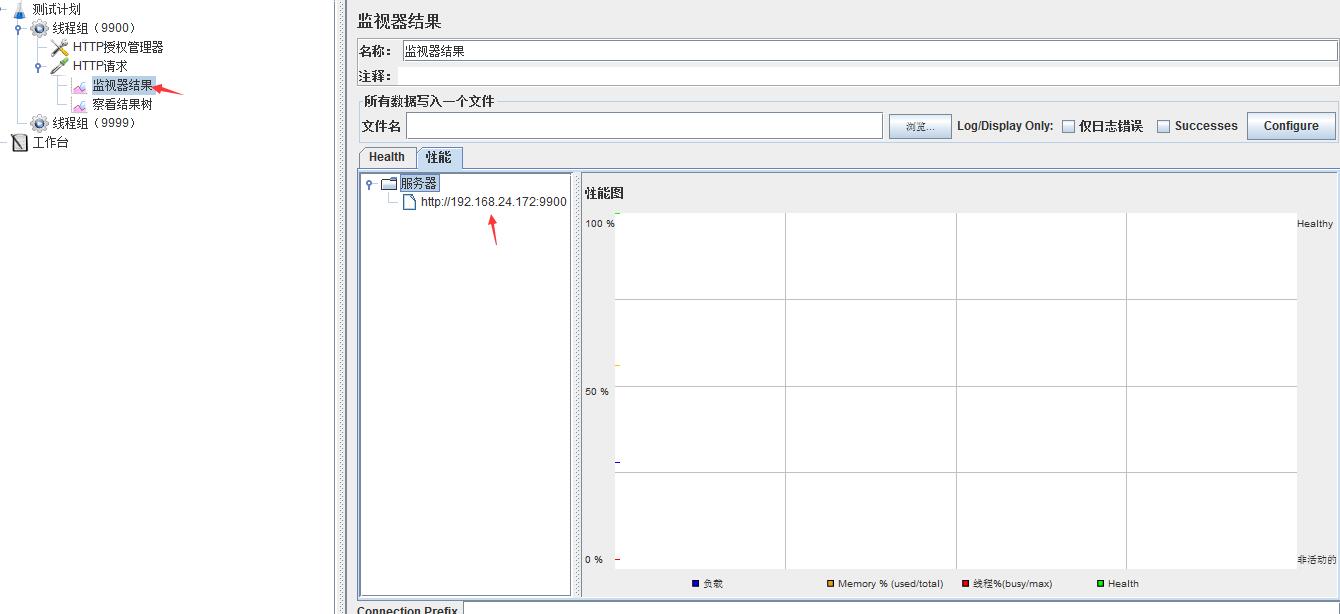


图A2

5、接着在当前HTTP请求下添加“监视器结果”和“察看结果树”，如下图所示



6、最后运行JMETER脚本，此时进入到“监视器结果”界面可以看到被测服务器IP和资源（图片），如下图所示



### 2.2、监控linux资源

1、首先下载JMeterPlugins-Extras-1.4.0和JMeterJMeterPlugins-Standard-1.3.1插件（下载地址：https://jmeter-plugins.org/downloads/old/）

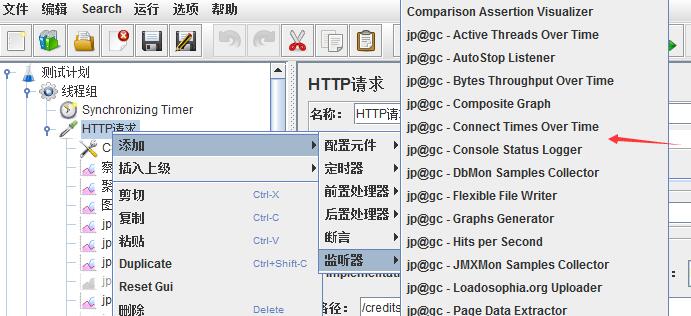
2、然后将解压出来的JAR包放到Jmeter/lib/ext目录下

3、再下载ServerAgent-2.2.1.zip（默认端口4444，下载地址：

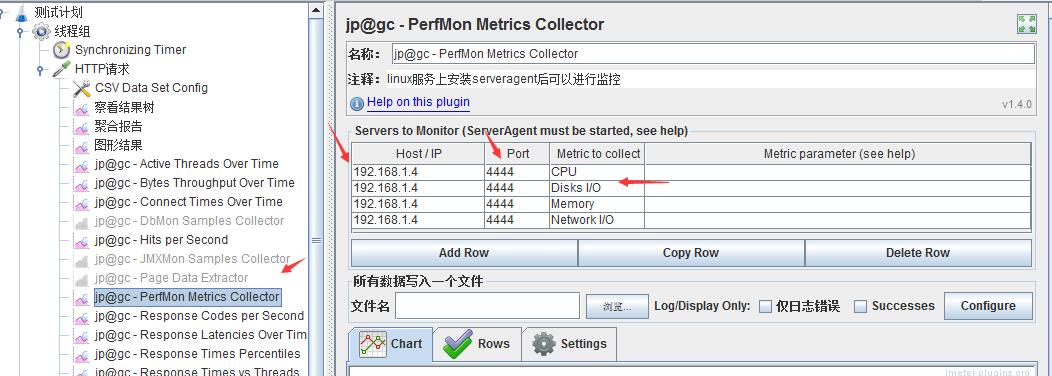
http://www.cnblogs.com/lsjdddddd/p/5709052.html）并部署到Linux下

4、在Linux下进入到ServerAgent目录下启动服务：sh startAgent.sh

5、打开Jmeter在HTTP请求下添加“监视器”即可看到JP@开头的监视元素



6、监控Linux服务器资源添加“jp@gc-PerMon Metrics Collector”，然后输入Linux的IP，端口号4444,和要监控的资源，如下图所示



**注意：监控完一次服务器资源后，ServerAgent会自动关闭，需要重新启动。否则第二次执行Jmeter脚本会报错。**

## 3、测试数据库性能

数据库驱动

| Database | Driver class | Database URL |
| --- | --- | --- |
| MySql | com.mysql.jdbc.Driver | jdbc:mysql://host:port/{dbname} |
| PostgreSQL | org.postgresql.Driver | jdbc:postgresql:{dbname} |
| Oracle | oracle.jdbc.driver.OracleDriver | jdbc:oracle:thin:user/pass@//host:port/service |
| Ingres (2006) | ingres.jdbc.IngresDriver | jdbc:ingres://host:port/db[;attr=value] |
| MSSQL | com.microsoft.sqlserver.jdbc.SQLServerDriver  或者  net.sourceforge.jtds.jdbc.Driver | jdbc:sqlserver://IP:1433;databaseName=DBname  或者  jdbc:jtds:sqlserver://localhost:1433/"+"library" |

1、安装对应数据库的JDBC驱动包，我这里用的是MySql，所以驱动包名是

mysql-connector-java-3.0.17-ga-bin.jar

2、将驱动包复制到jmeter/lib目录下，重启JMETER

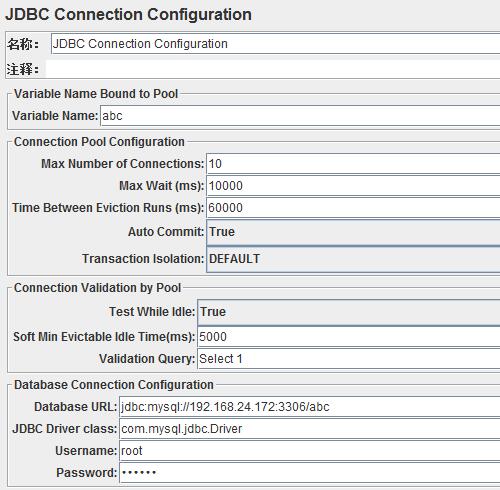
3、在JMETER中先新建线程组，然后在此线程组下新增“配置元件-JDBC Connection Configuration”，并设置此配置元件，配置如下（如下图所示）

Database URL：jdbc:mysql://192.168.24.172:3306/abc

JDBC Driver class：com.mysql.jdbc.Driver

Username：root（数据库用户）

Password：111111

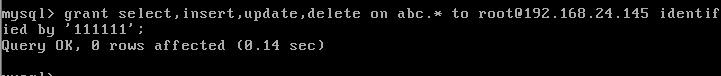


4、接着进入到MySql服务器，在abc数据库的shell界面，输入命令：

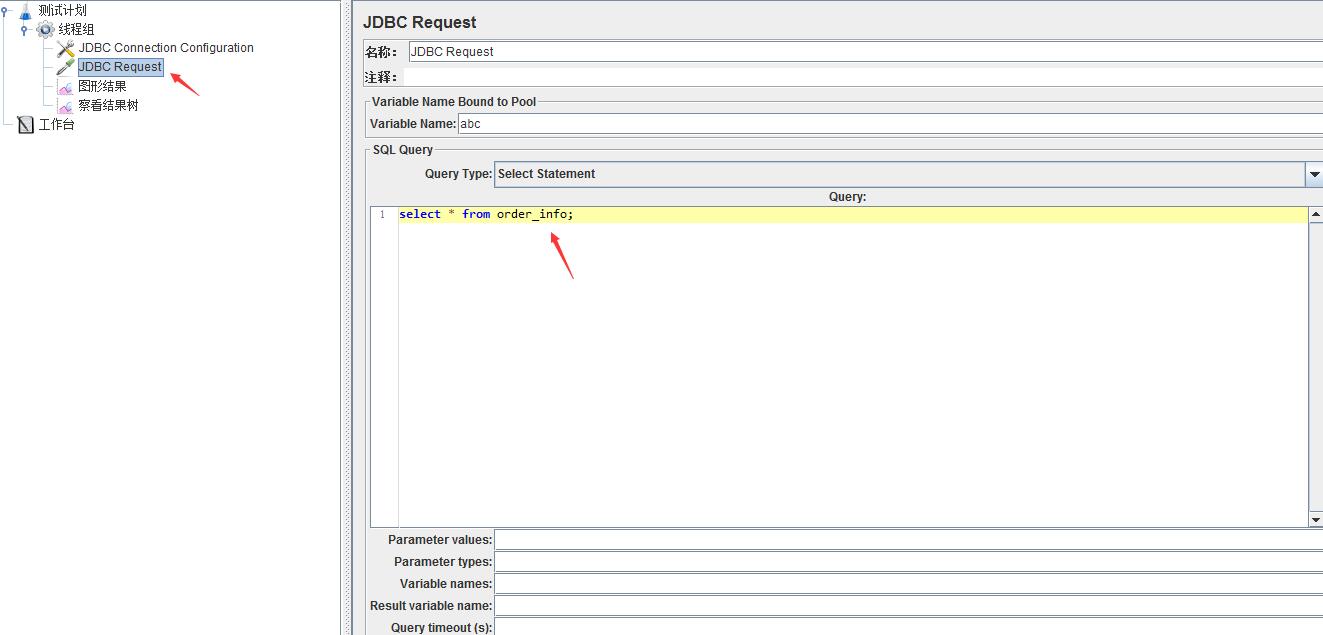
**Grant select,insert,update,delete on abc.\* to** [**root@192.168.24.145**](mailto:root@192.168.24.145) **identified by ‘’111111;**

abc是数据库名，将本地IP和用户root授予访问MySql服务器权限并且将密码设置为111111。

如果授权成功则显示如下图所示



5、此时重启jmeter，再在线程组下新增“Sampler-JDBC Request”，在SQL Query输入要执行的SQL语句，如下图所示



6、然后在该线程组下添加“察看结果树”和“图形结果”即可，最后执行脚本。然后再在“察看结果树”区域查看结果，如下图所示

