https://testerhome.com/topics/10998

[性能测试工具](https://testerhome.com/topics/node3) 用 Jmeter 工具压测上传文件接口的过程和自己的一些疑问，希望有高人能指点一下

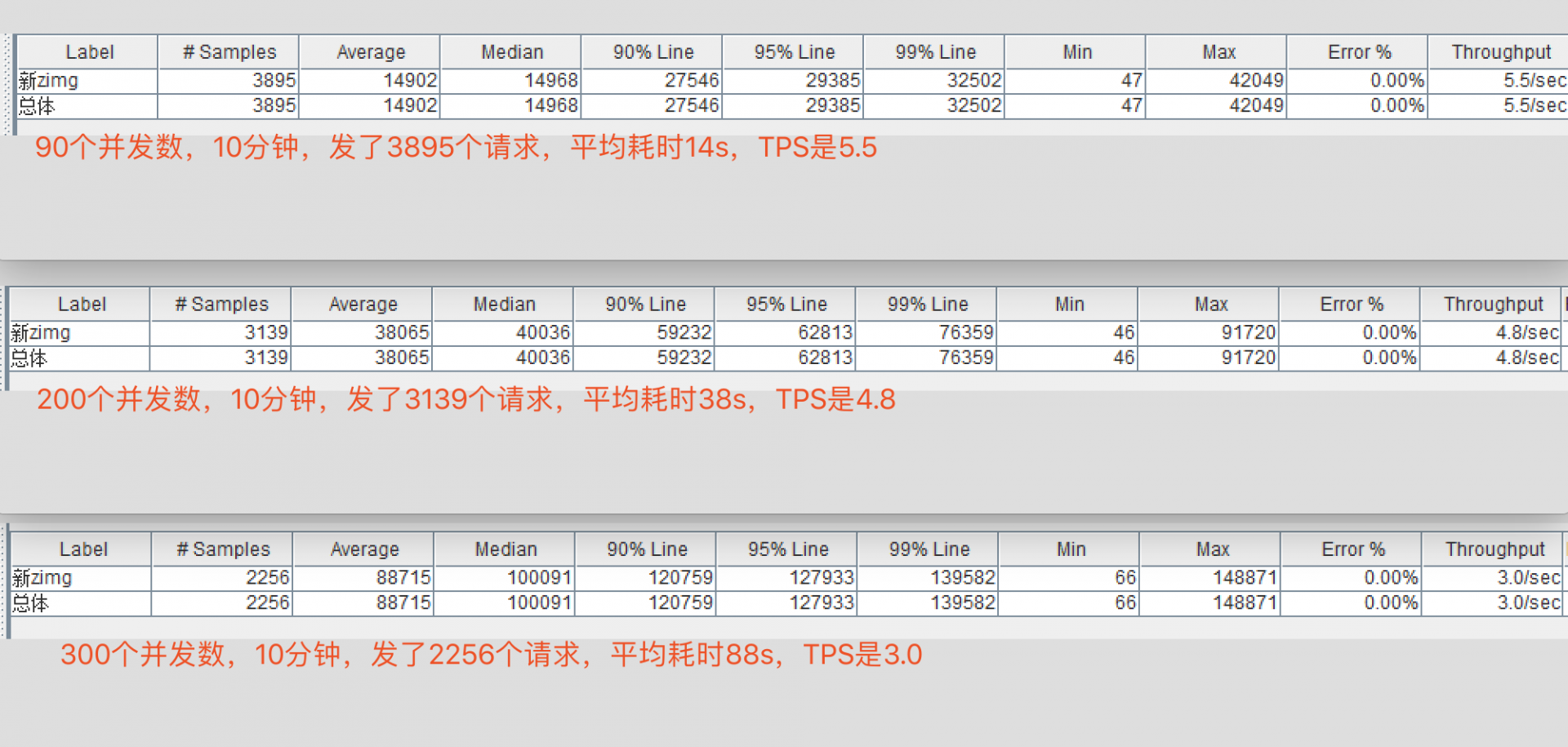
[肥猫](https://testerhome.com/imyangdy) · 2017年11月27日 · 最后由 [Baozhida](https://testerhome.com/utopia) 回复于 2017年12月11日 · 4396 次阅读

 目录

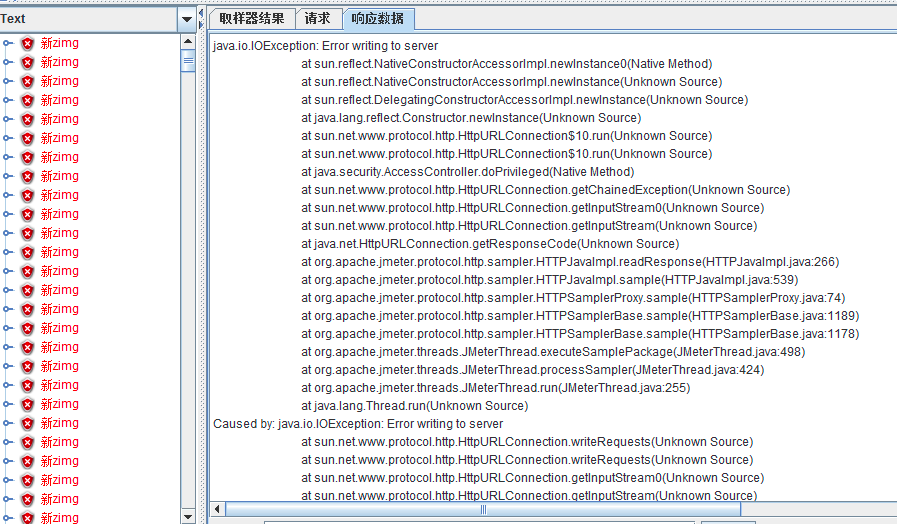
**遇到的疑问以及自己的思考（不一定对，希望知道的人可以说两句:）**

**1、** 当并发线程数设置越大时（100--200--300），TPS反而越低，但前后台都没有报错，服务器的资源占用也可以忽略不计

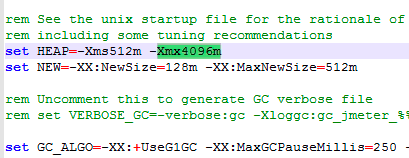
**个人认为原因**：因为网络带宽原因限制了，当线程数在增加时，同样时间内请求发送的数量越来越少，压力并没有到后台服务器，时间都消耗在了上传文件这个阶段

[[](https://testerhome.com/uploads/photo/2017/c0e34b97-ddf7-48e3-85d5-d63d0df9ddd5.png!large)](https://testerhome.com/uploads/photo/2017/c0e34b97-ddf7-48e3-85d5-d63d0df9ddd5.png!large)

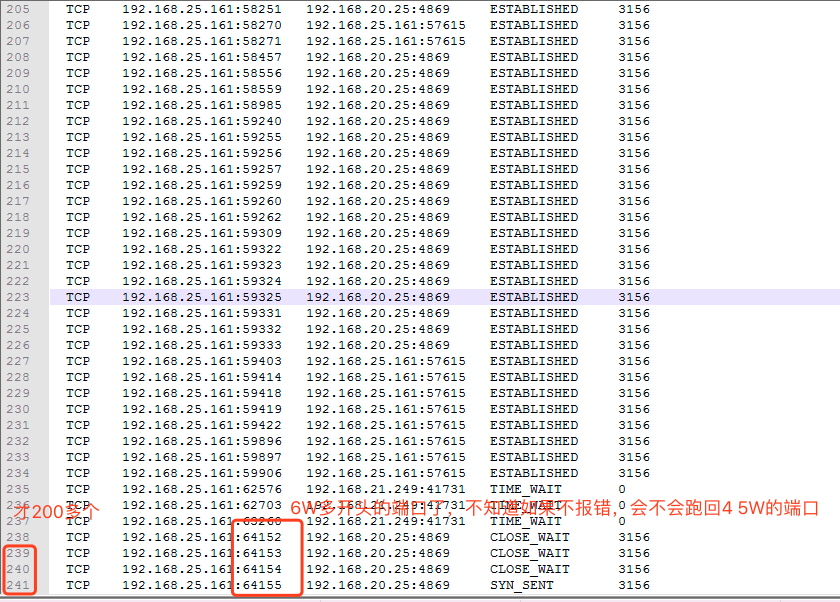
**2、** 每次当并发数到400左右的时候（不管是单台机压400，或者是五台客户机搞分布式压400，又或者是每个客户机独立同时向服务器发80），总是会报错，错误信息都是一样  
java.io.IOException: Error writing to server

[[](https://testerhome.com/uploads/photo/2017/2306175b-637c-4ea4-9ec4-60245b3d998a.png!large)](https://testerhome.com/uploads/photo/2017/2306175b-637c-4ea4-9ec4-60245b3d998a.png!large)

在网上也看了一些解决办法，但还是没有解决，比如修改jmeter.bat文件里的 HEAP的大小

[[](https://testerhome.com/uploads/photo/2017/2f9d3b1c-967c-4d1e-8242-1444087c771f.png!large)](https://testerhome.com/uploads/photo/2017/2f9d3b1c-967c-4d1e-8242-1444087c771f.png!large)

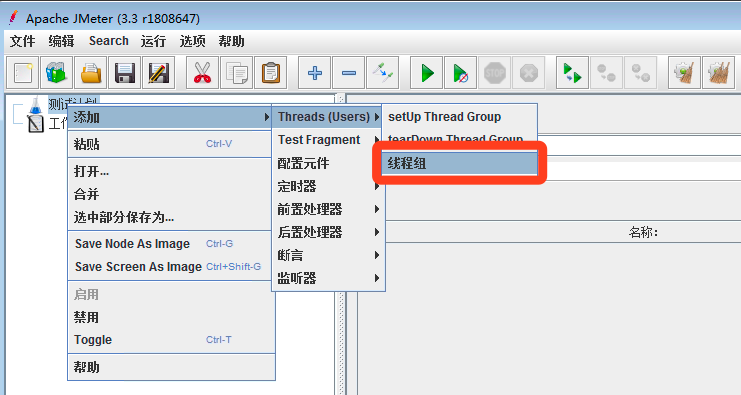
**个人认为原因**：每个请求时间过长，2-3分钟发不一个请求，导致启动的连接都在客户机，发不出去，然后端口占用完了，才会报错，因为每次报错的时候，端口都用到了6W开头了，如图，但我用netstat -ano去查看，一共才几百个端口，不知道用什么命令可以把所有占用的端口都显示出来，不知道为什么4W多的端口用了几十个，一下子就跳到了5W多的端口，然后又跳到6W多的端口，希望知道的高手能指点一下像我这种小白

[[](https://testerhome.com/uploads/photo/2017/2ac772aa-bba2-423e-bbdd-06ae5663b76f.png!large)](https://testerhome.com/uploads/photo/2017/2ac772aa-bba2-423e-bbdd-06ae5663b76f.png!large)

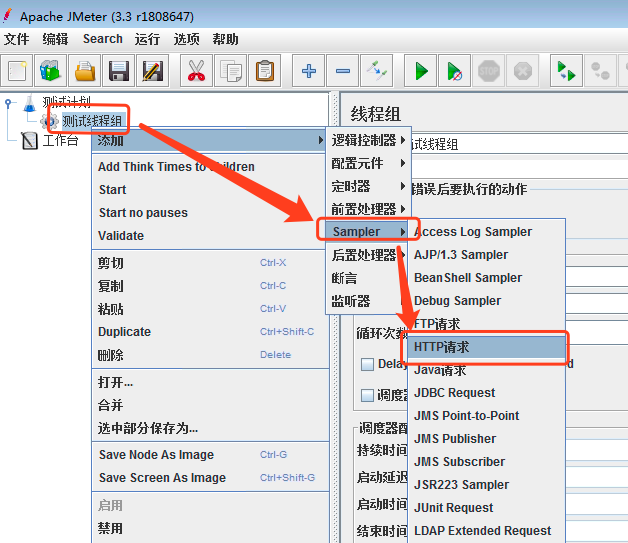
：

**脚本编写：**

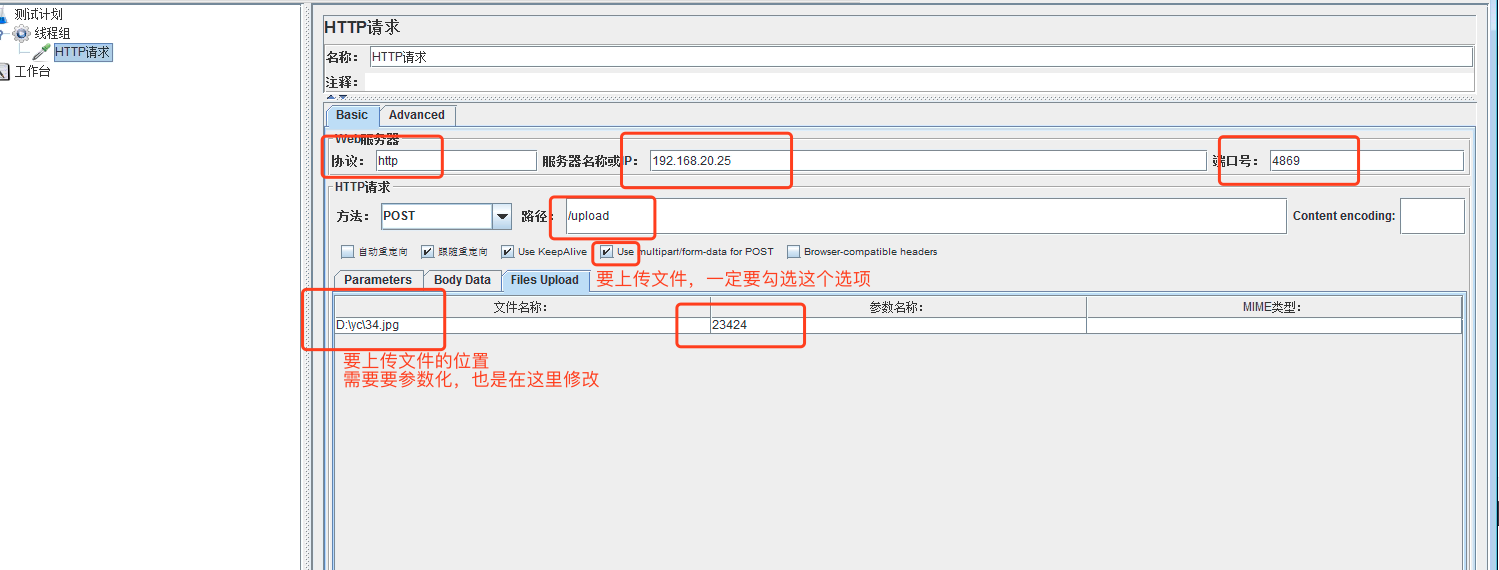
因为没有涉及cookie,token,加密之类的操作，就简单的一个上传文件的接口，所以脚本也很简单，  
看图就OK  
**1、**直接在测试计划中新建一个线程

[[](https://testerhome.com/uploads/photo/2017/adbacf63-9c28-4c5e-958e-1e9cc154addd.png!large)](https://testerhome.com/uploads/photo/2017/adbacf63-9c28-4c5e-958e-1e9cc154addd.png!large)

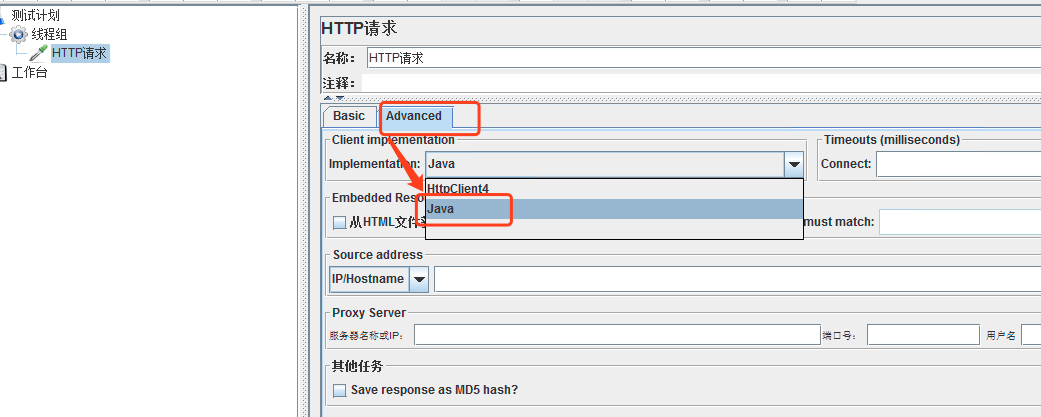
**2、**然后在线程组中新建一个http请求

[[](https://testerhome.com/uploads/photo/2017/c7a2484c-6187-4ea2-962d-58e4cd3f68fe.png!large)](https://testerhome.com/uploads/photo/2017/c7a2484c-6187-4ea2-962d-58e4cd3f68fe.png!large)

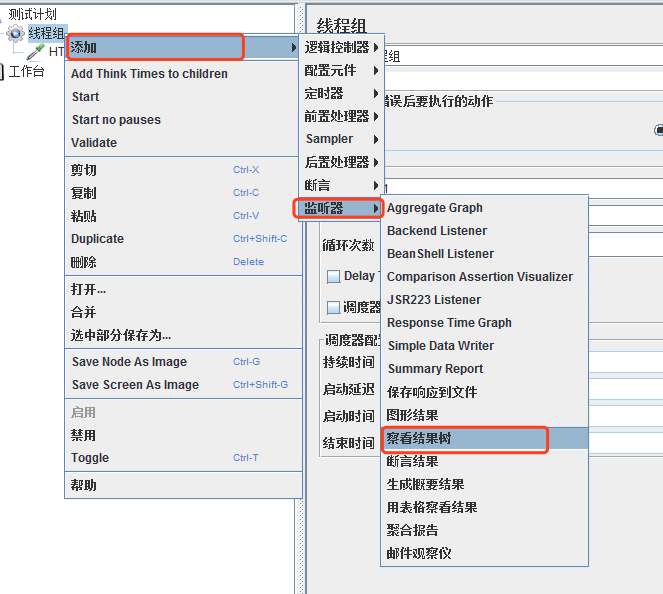
**3、**填写需要压测的接口和参数等内容

[[](https://testerhome.com/uploads/photo/2017/0529a479-0a17-4f00-a4cf-8b6c8d69519e.png!large)](https://testerhome.com/uploads/photo/2017/0529a479-0a17-4f00-a4cf-8b6c8d69519e.png!large)

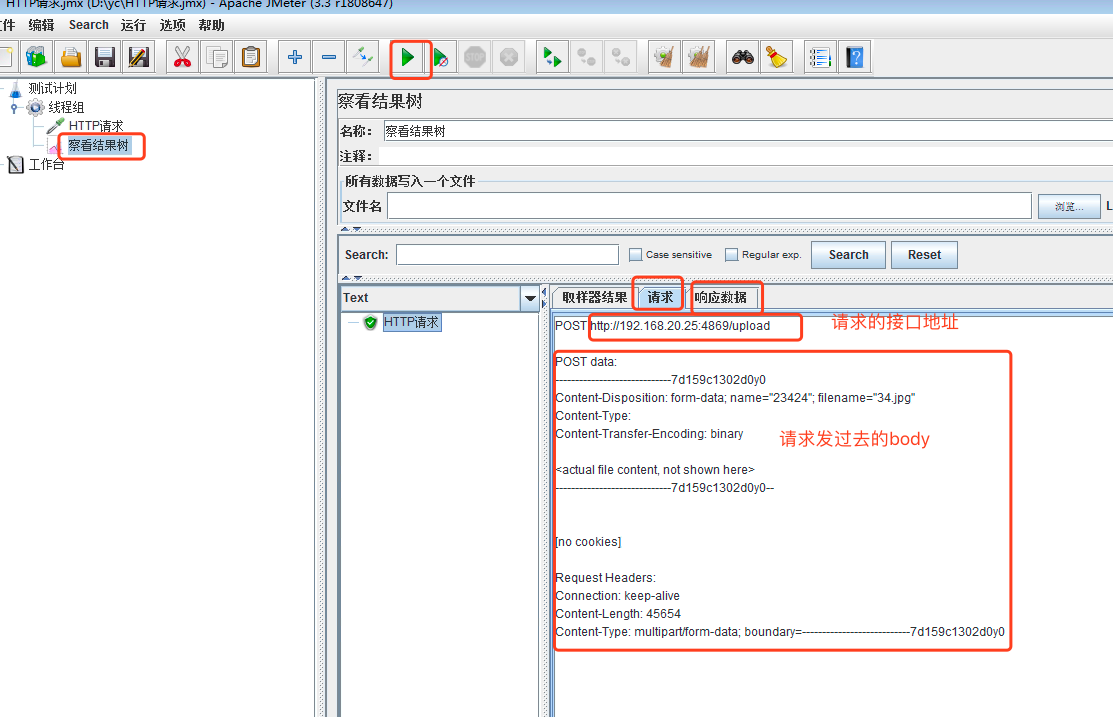
**4、** 到Advanced标签页选择请求的方式，通过Java方式发送请求

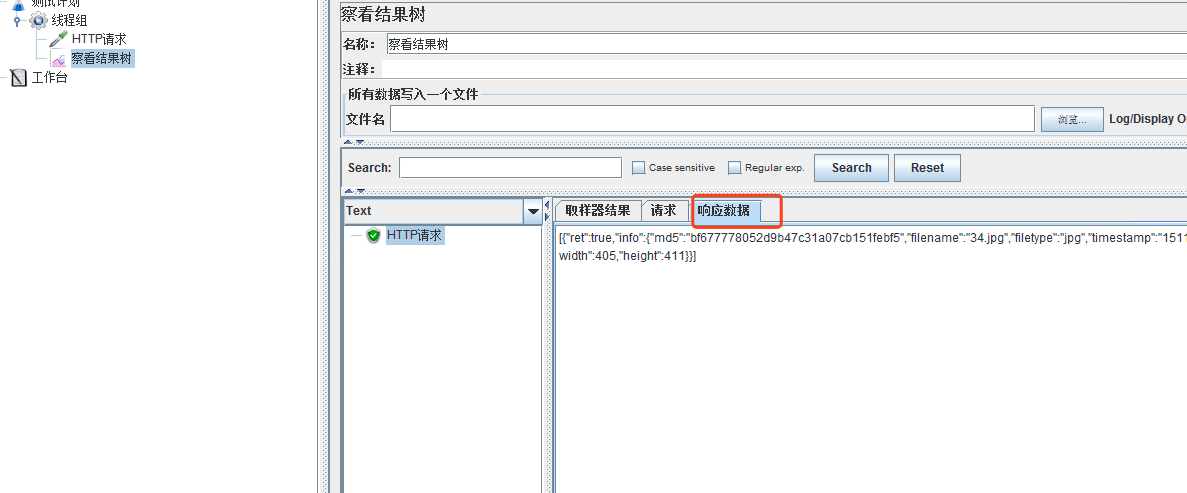
[[](https://testerhome.com/uploads/photo/2017/838b8458-aef5-4b4f-97b3-40177835eeda.png!large)](https://testerhome.com/uploads/photo/2017/838b8458-aef5-4b4f-97b3-40177835eeda.png!large)

**5、**这样就可以发送请求到服务器了，但我们还想看到我们发过去的请求内容和服务器返回给我们的内容，所以要添加一个监听器类型的控件，比如“查看结果树”

[[](https://testerhome.com/uploads/photo/2017/193141f2-9b8b-4cc1-877e-86022cab4f21.png!large)](https://testerhome.com/uploads/photo/2017/193141f2-9b8b-4cc1-877e-86022cab4f21.png!large)

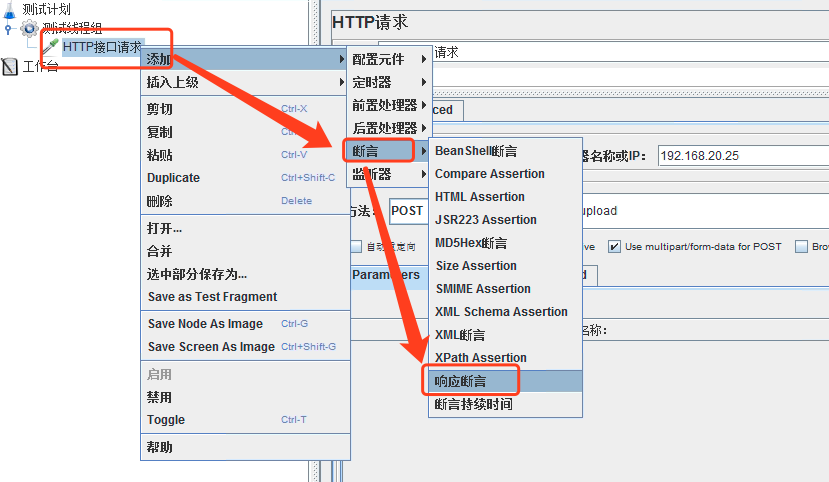
**6、**添加完查看结果树后，我们发送一次请求，然后到结果树查看发送过去的请求和服务器返回的数据

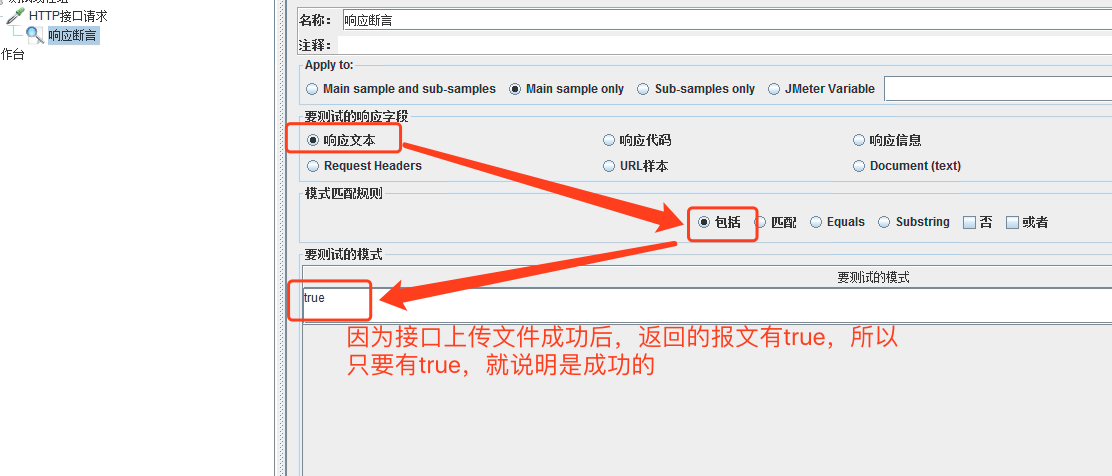
[[](https://testerhome.com/uploads/photo/2017/03c763c9-8ce5-4a2e-8b1c-b308416fa00d.png!large)](https://testerhome.com/uploads/photo/2017/03c763c9-8ce5-4a2e-8b1c-b308416fa00d.png!large)

[[](https://testerhome.com/uploads/photo/2017/c6064287-c5dd-4417-a47d-80bb73aca469.png!large)](https://testerhome.com/uploads/photo/2017/c6064287-c5dd-4417-a47d-80bb73aca469.png!large)

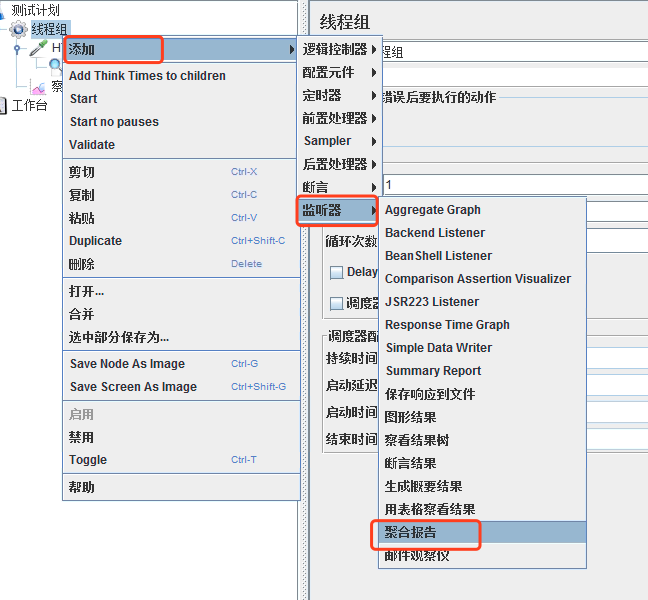
**7、**绿色代表成功，红色代表失败，如果失败，要排查就我们发过去的内容和服务器返回的内容来定位，具体问题具体分析

**8、**下一步，我们再添加一个断言，根据我们自己的定义来判断，什么时候是成功，什么时候是失败

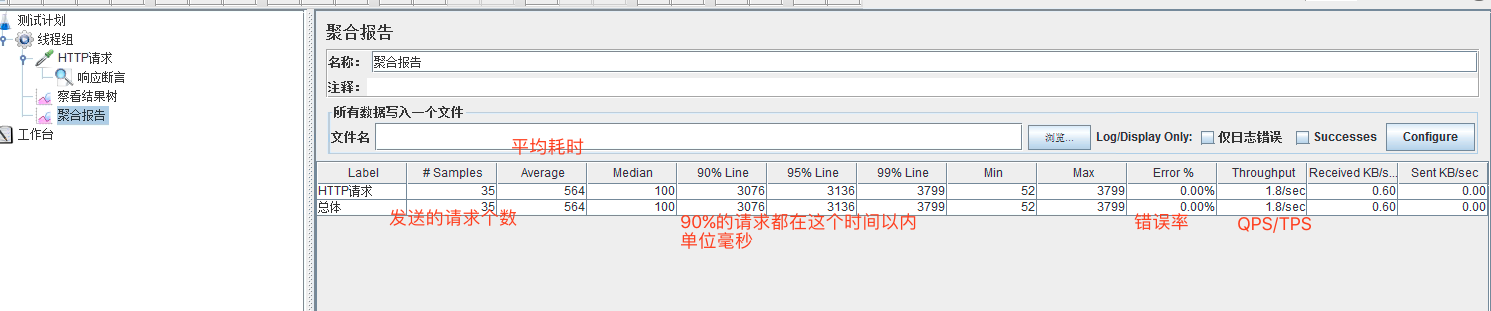
[[](https://testerhome.com/uploads/photo/2017/2859e22e-f7ec-4f38-bbcf-7cc63e41e0ab.png!large)](https://testerhome.com/uploads/photo/2017/2859e22e-f7ec-4f38-bbcf-7cc63e41e0ab.png!large)

[[](https://testerhome.com/uploads/photo/2017/95b3e079-1c78-4c6c-9e14-3b3a9d618f47.png!large)](https://testerhome.com/uploads/photo/2017/95b3e079-1c78-4c6c-9e14-3b3a9d618f47.png!large)

**9、**再下一步，我们再添加一个监听类型，统计我们的压测结果，比如聚合报告

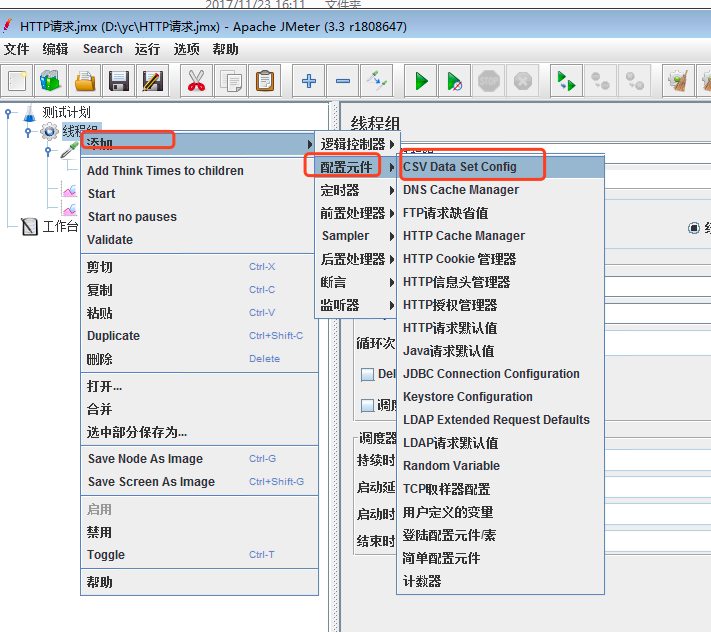
[[](https://testerhome.com/uploads/photo/2017/f8b58c6d-5793-4580-88f5-ef910f53b904.png!large)](https://testerhome.com/uploads/photo/2017/f8b58c6d-5793-4580-88f5-ef910f53b904.png!large)

**10、** 执行之后，我们就可以在聚合报告里查看结果

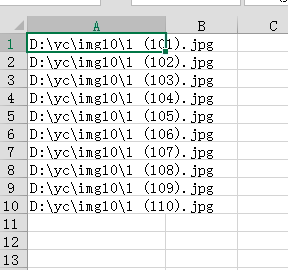
[[](https://testerhome.com/uploads/photo/2017/a38417d0-ba53-4741-bcd6-d48e8e2180d1.png!large)](https://testerhome.com/uploads/photo/2017/a38417d0-ba53-4741-bcd6-d48e8e2180d1.png!large)

**11、**参数化

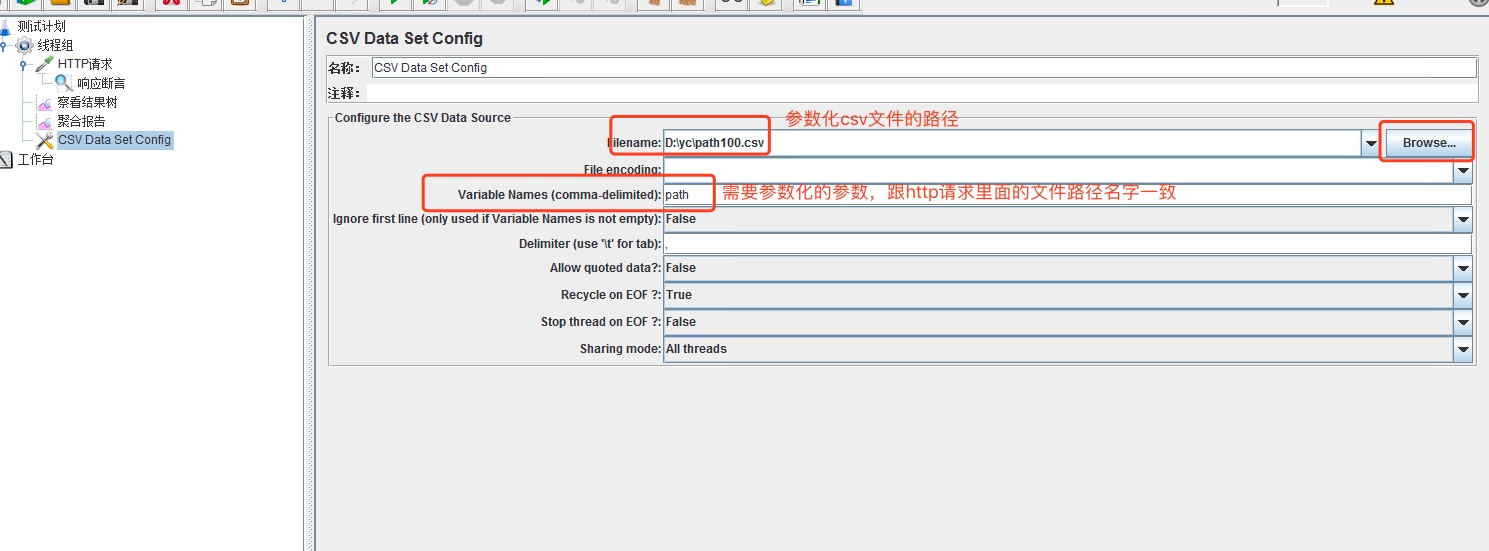
添加一个CSV文件配置

[[](https://testerhome.com/uploads/photo/2017/49946106-da23-487b-aae0-85f2ecdb7dce.png!large)](https://testerhome.com/uploads/photo/2017/49946106-da23-487b-aae0-85f2ecdb7dce.png!large)

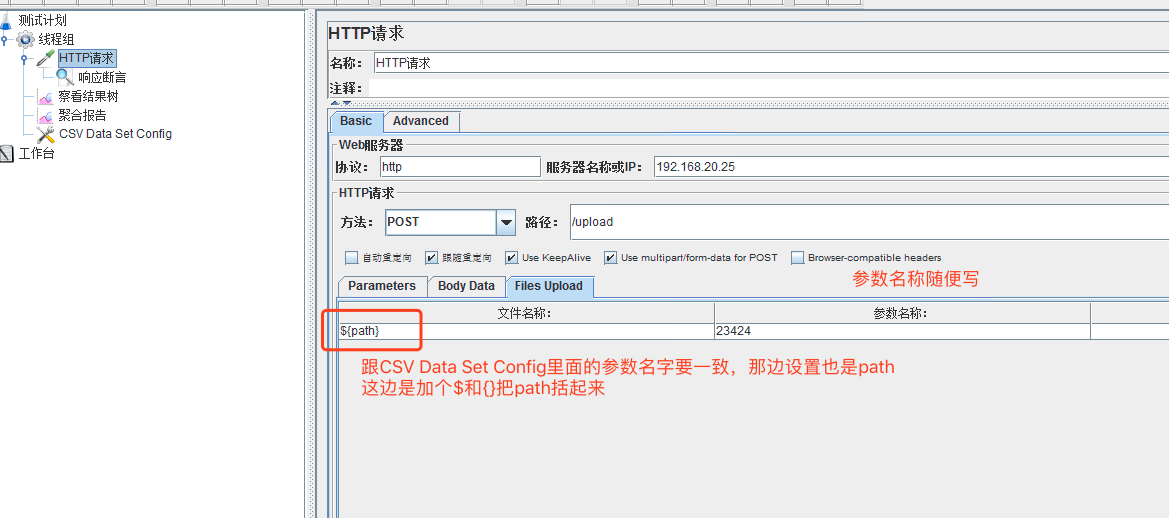
然后在本地新建一个csv文件，里面填上要上传文件的路径

[[](https://testerhome.com/uploads/photo/2017/fb90c525-15d6-49d1-ace5-802f02366614.png!large)](https://testerhome.com/uploads/photo/2017/fb90c525-15d6-49d1-ace5-802f02366614.png!large)

* 然后修改CSV文件配置，选上刚刚建的CSV文件和其它的配置信息

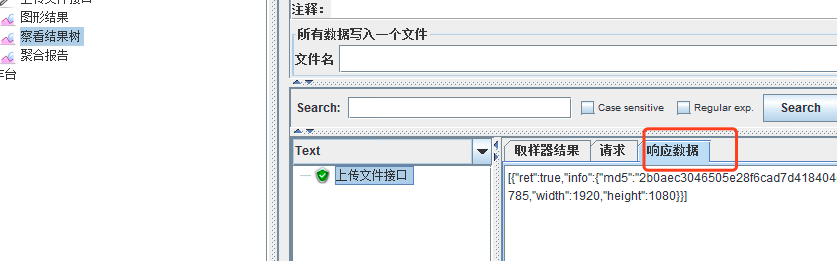
[[](https://testerhome.com/uploads/photo/2017/a70d1eed-8719-46b7-ba0c-602d6ec8e01b.png!large)](https://testerhome.com/uploads/photo/2017/a70d1eed-8719-46b7-ba0c-602d6ec8e01b.png!large)

最后修改一下发送请求的文件

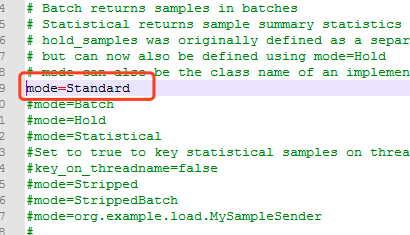
[[](https://testerhome.com/uploads/photo/2017/470f85aa-0ecf-4288-b699-c7b8c200e491.png!large)](https://testerhome.com/uploads/photo/2017/470f85aa-0ecf-4288-b699-c7b8c200e491.png!large)

**12、**关于多台机发布式时的问题：

* 执行机（slave）要到bin目录下的：jmeter-server.bat，启动该执行机
* 执行机IP的配置，在jmeter.properties里配置
* 需要需要在察看结果树里看到执行机的发送请求之后服务器返回的响应数据

[[](https://testerhome.com/uploads/photo/2017/236f2dcd-0714-47a9-83e4-a234845e1b9c.png!large)](https://testerhome.com/uploads/photo/2017/236f2dcd-0714-47a9-83e4-a234845e1b9c.png!large)

要在jmeter.properties里配置打开标准模式，如图

[[](https://testerhome.com/uploads/photo/2017/927981f8-033f-4c70-9740-1fb972d46bfb.png!large)](https://testerhome.com/uploads/photo/2017/927981f8-033f-4c70-9740-1fb972d46bfb.png!large)

「原创声明：保留所有权利，禁止转载」