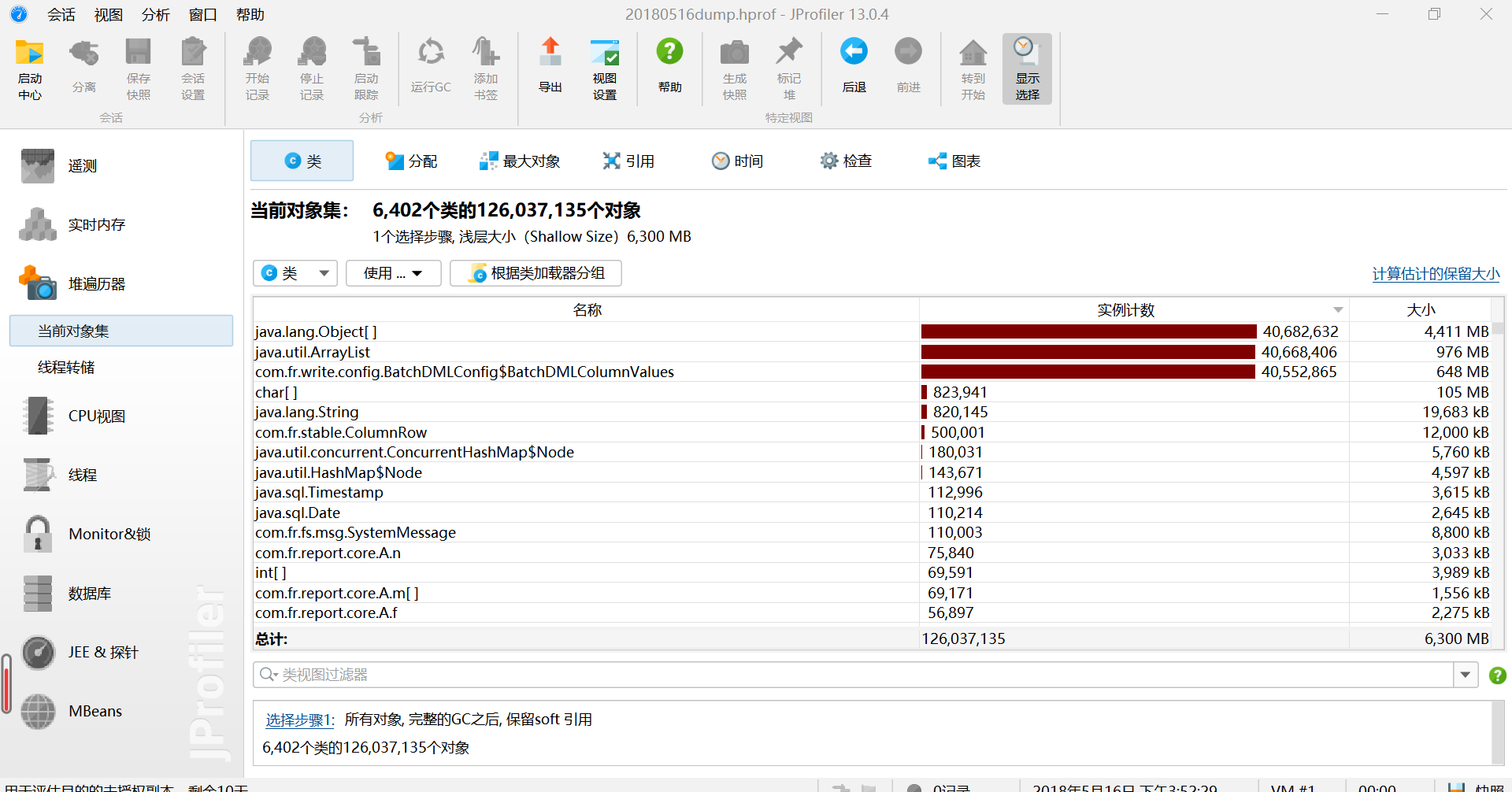
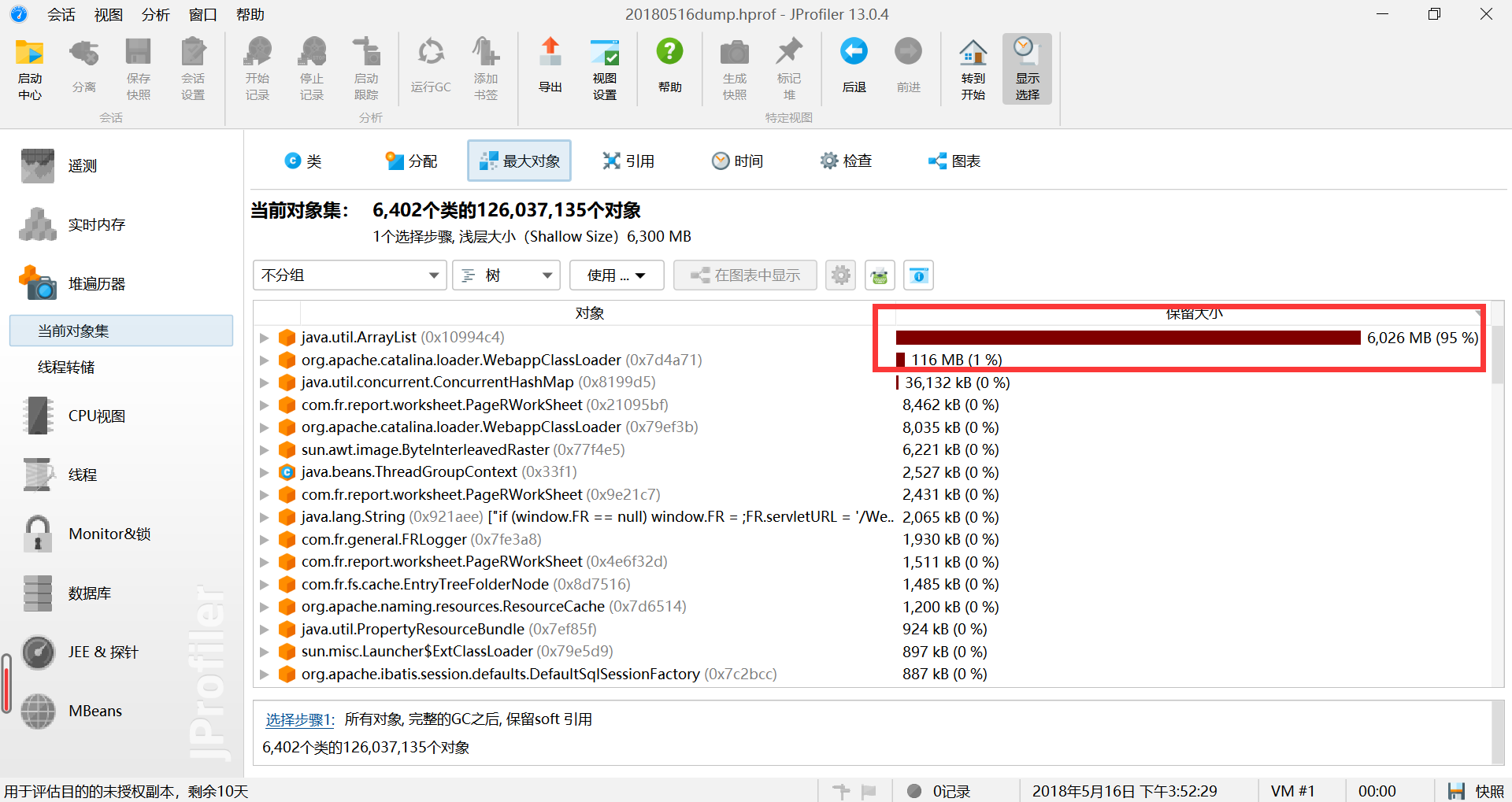
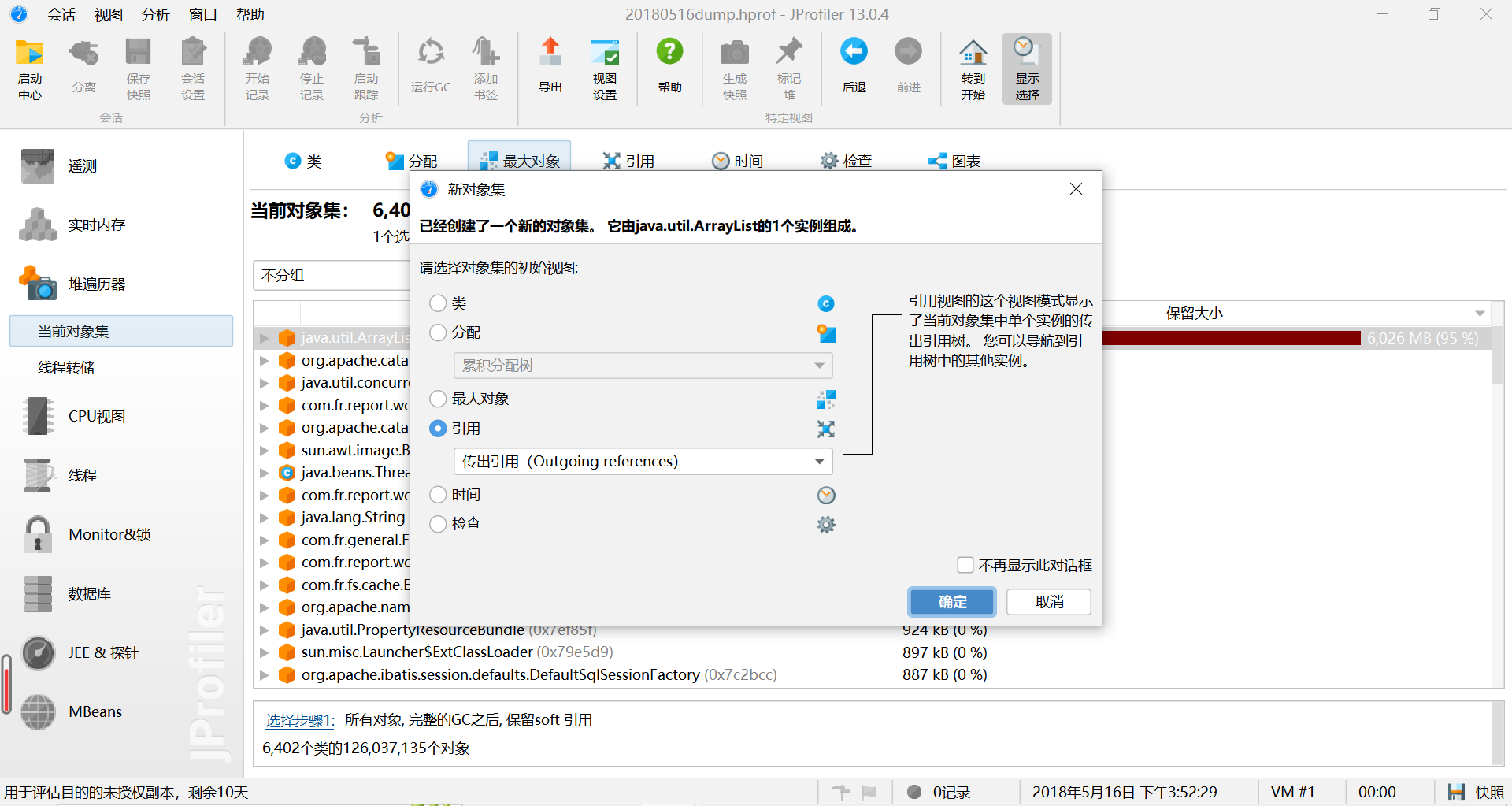
1.打开第二个文件如下所示



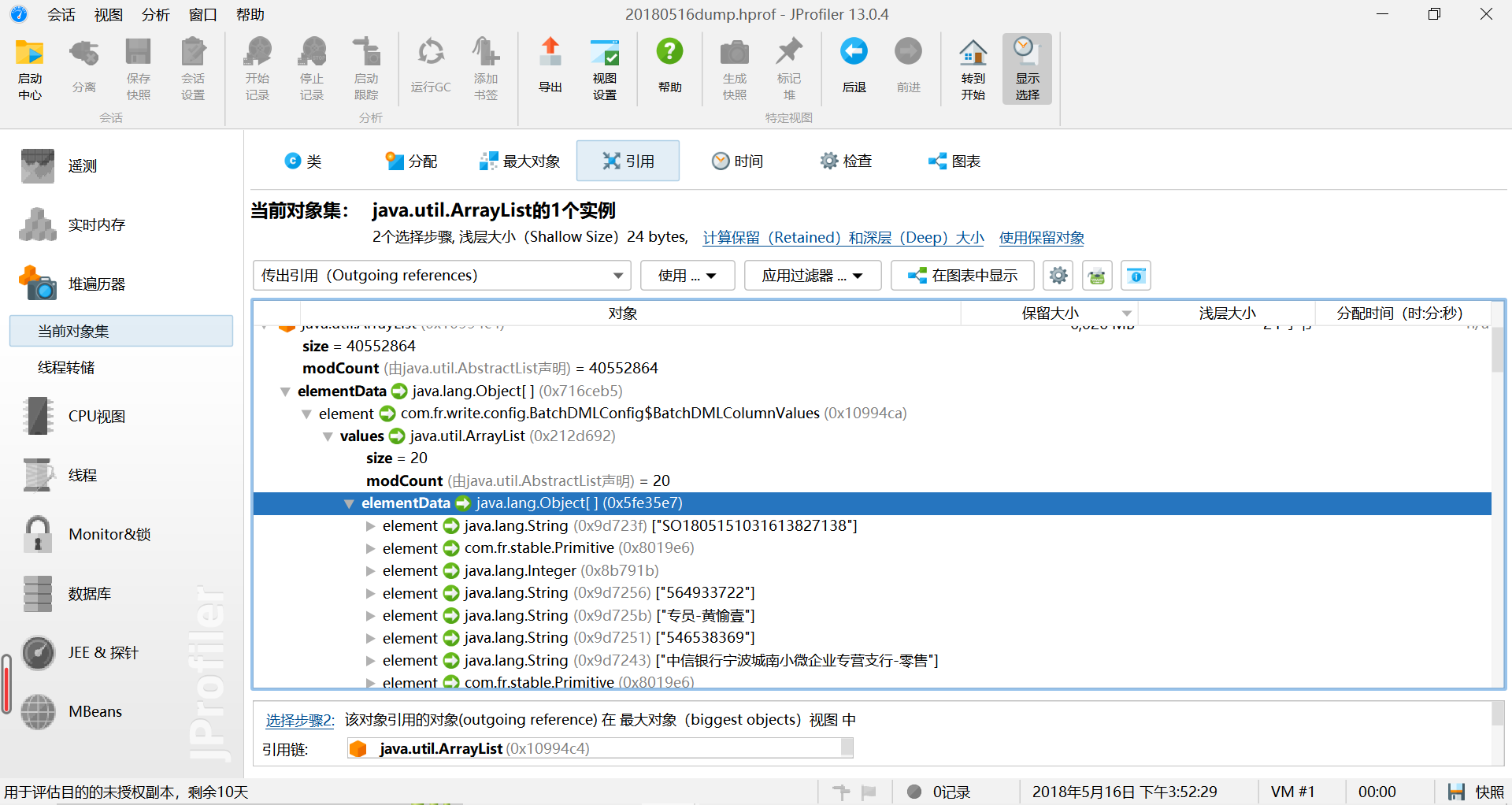
2.点击最大对象，发现该对象占有6GB



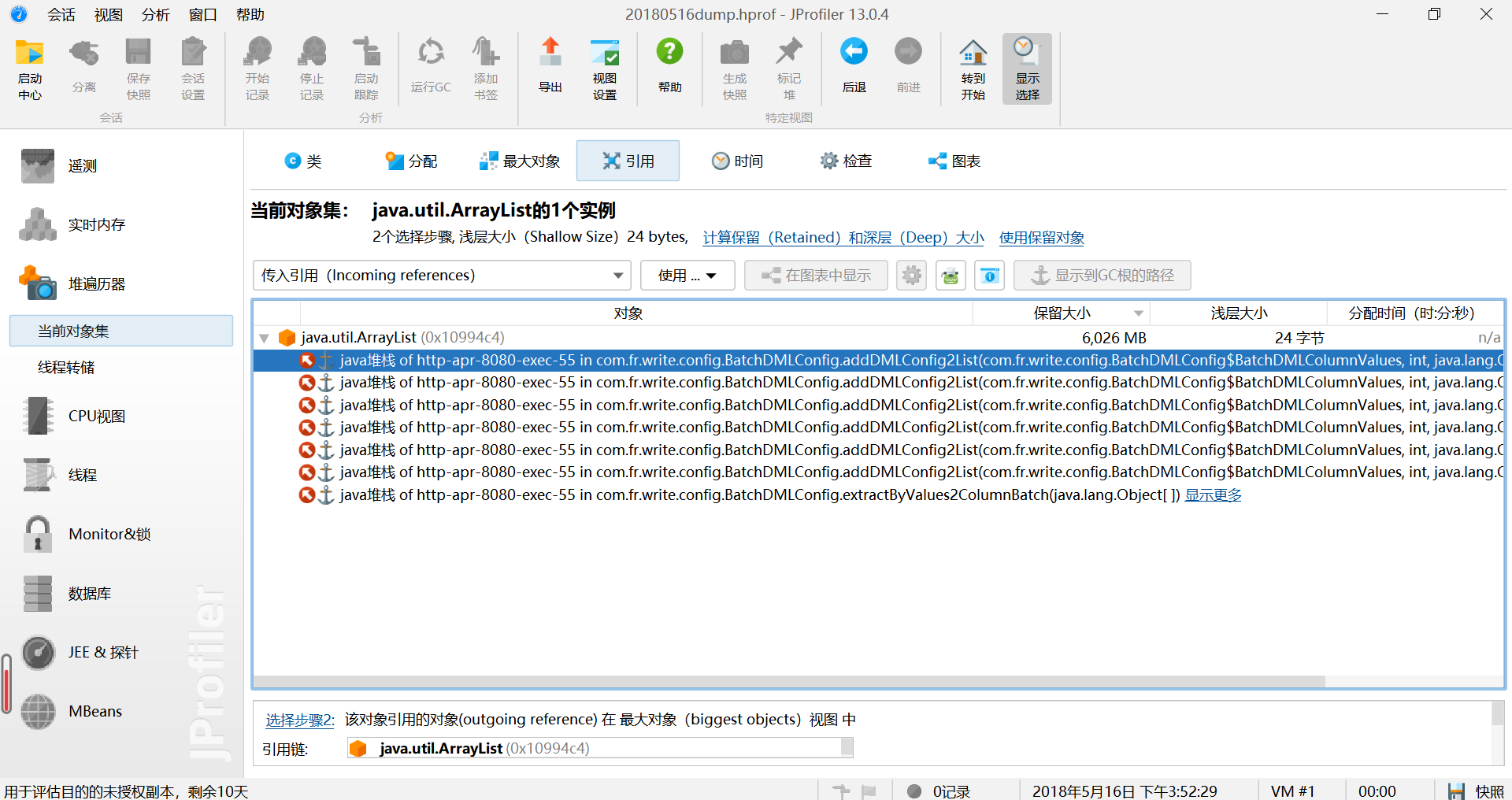
3.选中ArrayList，使用指定对象传出引用



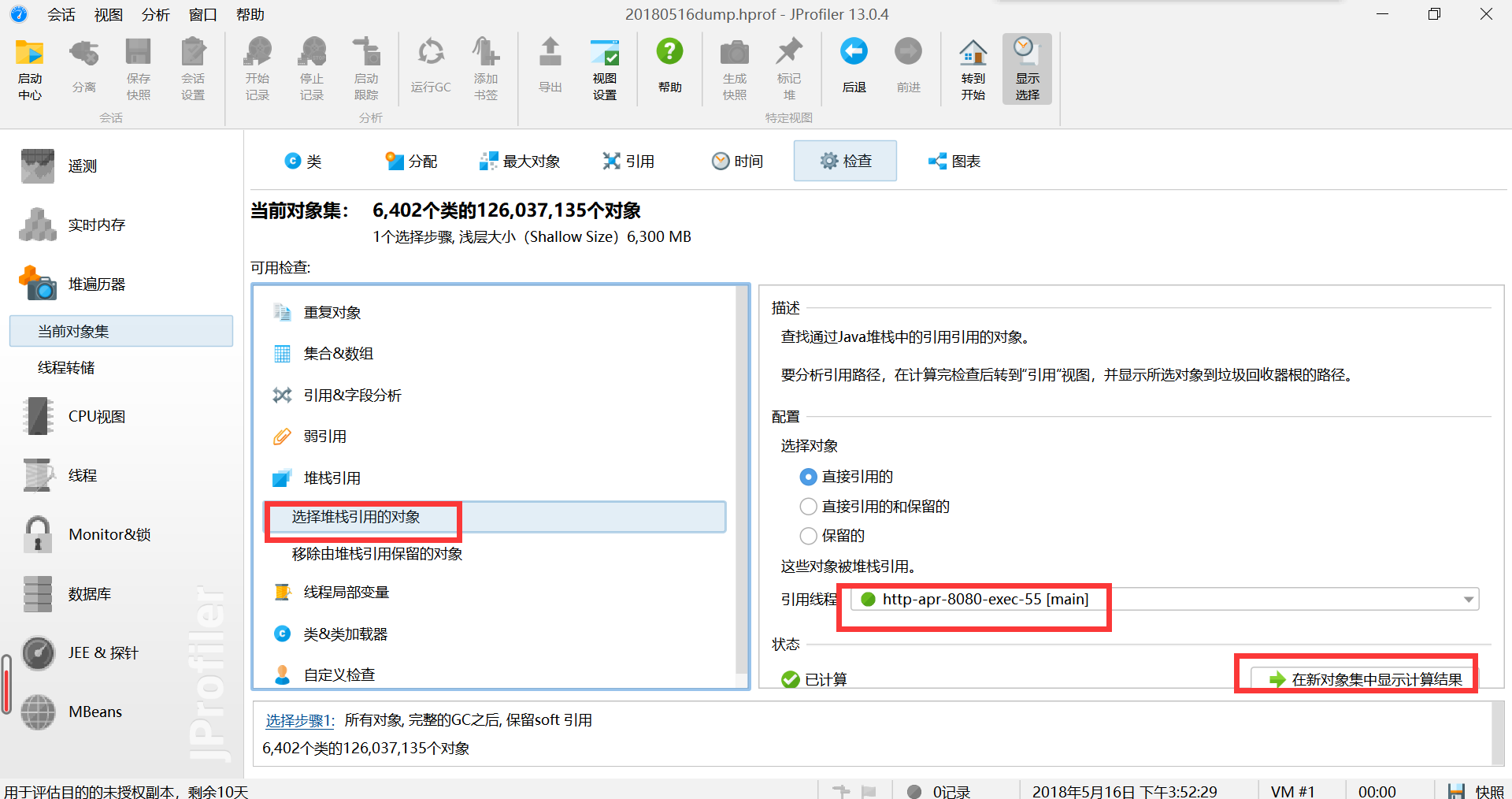
4.展开ArrayList，发现内部是一个填报对象，可以确定是笛卡尔积问题，去寻找模板



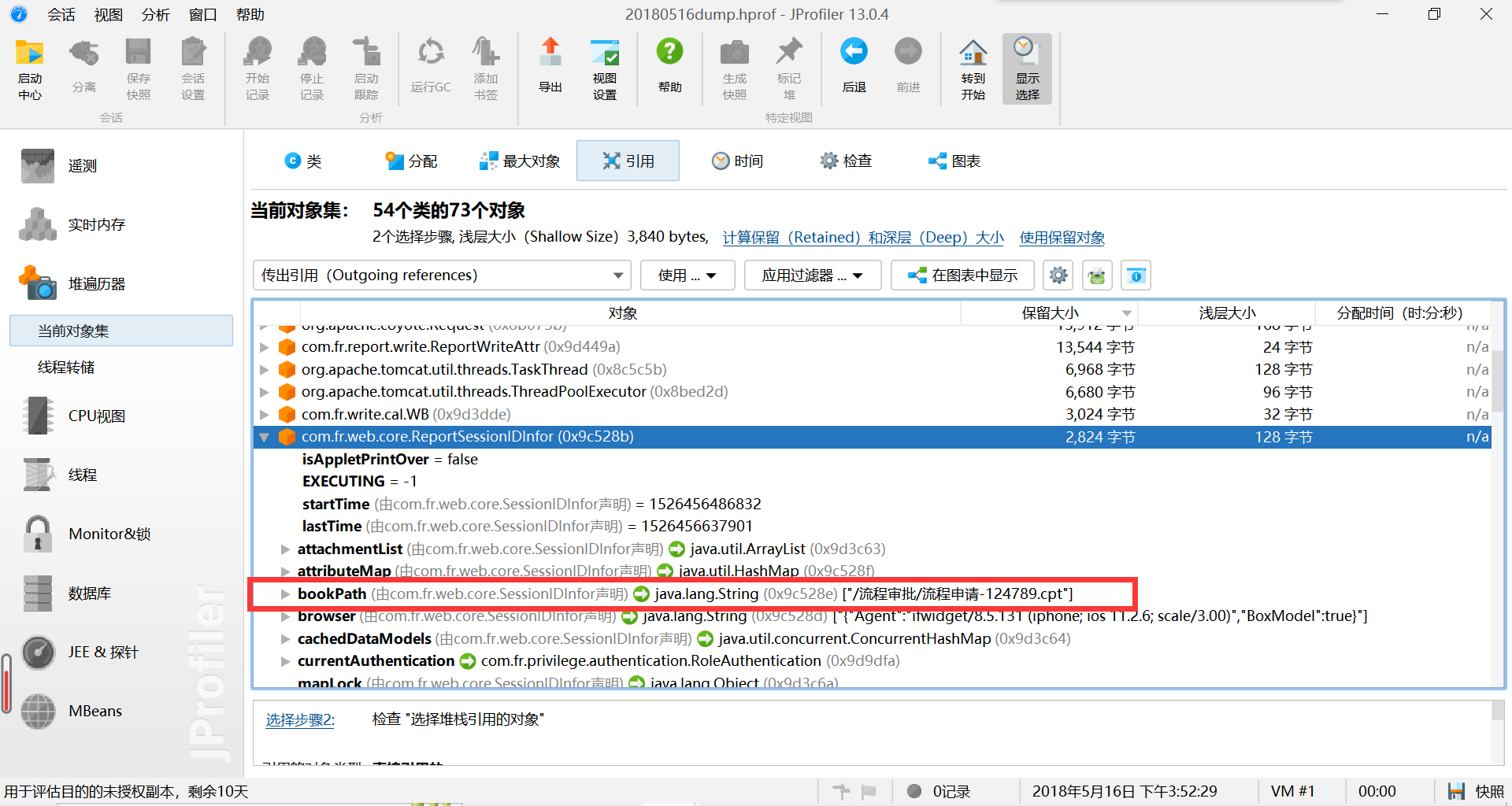
5.切换到传入引用，显示到gc root的路径，这是一个线程堆栈的引用栈，引用线程 http-apr-8080-exec-55。



6.回到开始，依次点击 检查、堆栈引用、选择堆栈引用的对象、引用线程http-apr-8080-exec-55[main]、计算检查并创建一个新的对象集



7.等待后如下，Ctrl F 搜索ReportSessionIdInfo并展开，得到问题模板：/流程审批/流程申请-124789.cpt



8.可以根据之前的大对象中的数据推测是填报模板中的哪条属性，总体来看是由于预览时产生笛卡尔积，导致java对象过大造成内存溢出而宕机，可以采用懒加载策略来缓解。

