发现应用进程掉线，且产生heap dump文件。当时执行定时批量作业，加载当地文本，申请内存空间时出现内存溢出。

深入分析heap dump文件，定位本次问题根源：

ThreadLocal 可以存储一个变量，该变量只有当前线程可以取出，对其它线程不可见，具有线程隔离性，在并发编程中常使用到。本地模块用于 异步文件定时扫描处理、报文处理、交易结果通知处理等。

此次故障中：本地模块在新建模块时引入公共方法封装jar包，公共方法里提供两个ThreadLocal封装类，分别是CacheStore和CacheRegister。两个类均提供remove方法进行内存回收，但调用CacheStore的set()方法时，CacheStore创建了新的ThreadLocal对象并将this（自己地址指针）添加到CacheRegister.mList静态变量中，而clearInfo()方法仅执行了ThreadLocal对象的remove()方法，没有将自己从CacheRegister.mList移除。

本地模块在进行ThreadLocal的清理时，调用的是CacheStore.clearInfo(),导致地址指针值一直存储在内存中，造成内存缓慢泄漏。由于对公共方法封装jar里的ThreadLocal封装类设计理念不熟悉，如果使用CacheRegister.clearInfo()，则可避免上述问题。

OBSS本地清算模块上线以来投产频率较高，每一两个月就有投产变更动作，未暴露该问题。自2405批次以来未有变更，其未被回收的ThreadLocal对象一直在积压增长，截止到故障发生时点，22:30定时任务执行需申请较大内存空间用于文件载入，触发了内存溢出。