

weblogic打印线程信息和堆Dump文件

通过命令行的方式获取线程日志和堆栈信息。

使用背景

一般应用于服务器繁忙或是无响应时，我们需要看一下线程的状态和详细信息。

注意事项

先不要重启服务器，在服务器无响应的时候执行下面获取步骤。

Windows

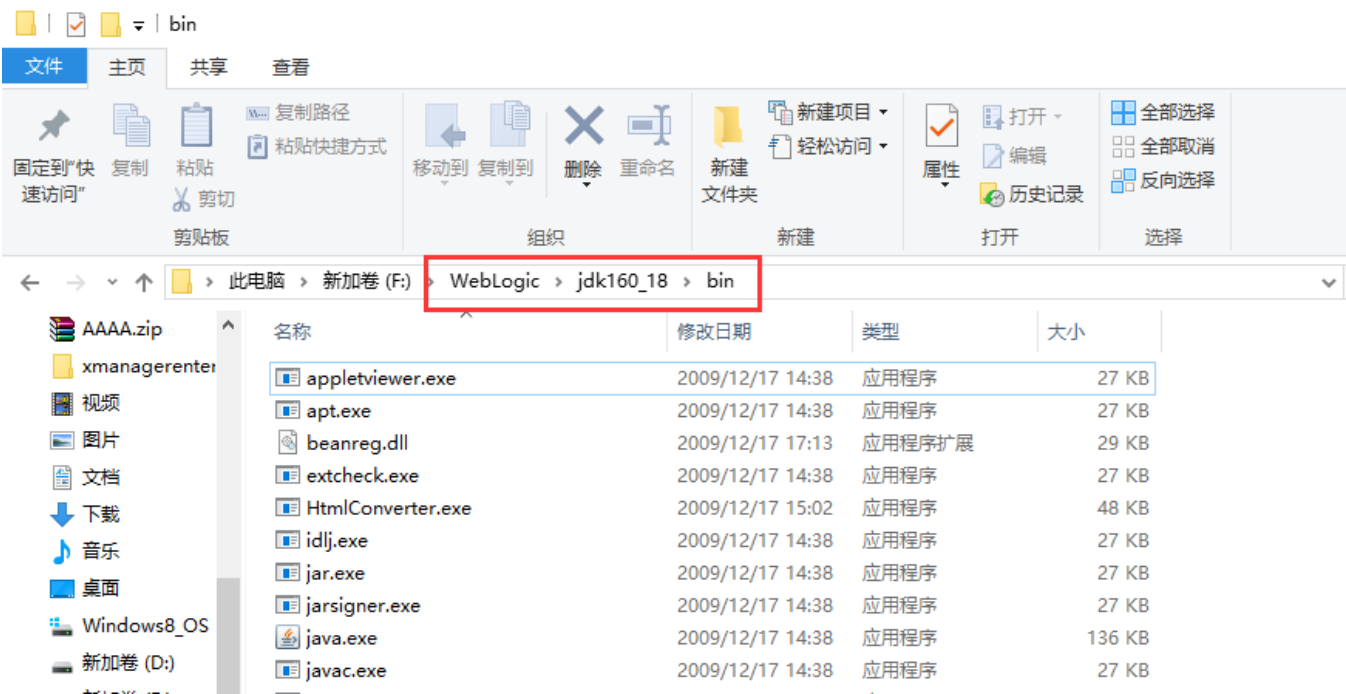
一、线程信息

请在服务器无响应时，执行打印进程信息，打印线程信息有两个方法：

方法一

要求JDK版本为1.6及其以上版本。

- 1、在运行中打开cmd命令行窗口。
- 2、在cmd窗口进入JDK的bin目录下，执行 **jps**获取进程信息，此处要保证执行的JDK是服务器使用的JDK。



命令行: `cd F:\WebLogic\jdk160_18\bin`

`jps`

```
命令提示符
Microsoft Windows [版本 10.0.15063]
(c) 2017 Microsoft Corporation。保留所有权利。

C:\Users\JOAN>f:

F:\>cd F:\WebLogic\jdk160_18\bin
F:\WebLogic\jdk160_18\bin>jps
3216 Server
1800 Jps
F:\WebLogic\jdk160_18\bin>
```

3、获取对应线程号，然后执行jstack +进程号 > 进程号.log 获取线程信息：WebLogic显示名称应该是Server

```
命令提示符
Microsoft Windows [版本 10.0.15063]
(c) 2017 Microsoft Corporation。保留所有权利。

C:\Users\JOAN>f:

F:\>cd F:\WebLogic\jdk160_18\bin
F:\WebLogic\jdk160_18\bin>jps
3216 Server
1800 Jps
F:\WebLogic\jdk160_18\bin>jstack 3216 >3216.log
F:\WebLogic\jdk160_18\bin>
```

4、可以在 F:\WebLogic\jdk160_18\bin 目录下看到生成的线程文件。

此电脑 > 新加卷 (F:) > WebLogic > jdk160_18 > bin				
名称	修改日期	类型	大小	
3216.log	2017/7/19 17:13	文本文档	42 KB	
appletviewer.exe	2009/12/17 14:38	应用程序	27 KB	

方法二

- 1、通过CMD命令行窗口进入JDK的bin目录下
- 2、使用netstat -ano|find “访问的端口号”|find “LISTENING” 获取进程号（注意双引号是必须的）
如访问smartbi的端口号是7001，则输入命令： netstat -ano|find “7001”|find “LISTENING”

命令提示符

Microsoft Windows [版本 10.0.15063]
(c) 2017 Microsoft Corporation。保留所有权利。

C:\Users\JOAN>F:

F:\>cd F:\WebLogic\jdk160_18\bin

F:\WebLogic\jdk160_18\bin>netstat -ano|find "7001"|find "LISTENING"

TCP	127.0.0.1:7001	0.0.0.0:0	LISTENING	17348
TCP	169.254.98.86:7001	0.0.0.0:0	LISTENING	17348
TCP	172.16.75.1:7001	0.0.0.0:0	LISTENING	17348
TCP	172.16.220.1:7001	0.0.0.0:0	LISTENING	17348
TCP	192.168.2.83:7001	0.0.0.0:0	LISTENING	17348
TCP	[::1]:7001	:::0	LISTENING	17348
TCP	[fe80::3033:663b:7b8c:95b1%7]:7001	:::0	LISTENING	17348
TCP	[fe80::4c84:63d8:854a:235a%5]:7001	:::0	LISTENING	17348
TCP	[fe80::697d:fde2:15b7:8d2e%4]:7001	:::0	LISTENING	17348
TCP	[fe80::943a:e902:f20b:90e0%11]:7001	:::0	LISTENING	17348
TCP	[fe80::953d:6390:9070:5a03%9]:7001	:::0	LISTENING	17348
TCP	[fe80::c9dc:915b:3b84:6256%3]:7001	:::0	LISTENING	17348
TCP	[fe80::f985:7aa2:b02c:aeda%2]:7001	:::0	LISTENING	17348

F:\WebLogic\jdk160_18\bin>

3、再使用jstack 进程号 >进程号.log 获取线程信息，如果提示无法连接到JVM，可以增加-F参数强制生成jstack -F 进程号 >进程号.log

上边截图获取到进程号是17348，可输入命令行：jstack 17348 >17348.log

命令提示符

Microsoft Windows [版本 10.0.15063]
(c) 2017 Microsoft Corporation。保留所有权利。

C:\Users\JOAN>F:

F:\>cd F:\WebLogic\jdk160_18\bin

F:\WebLogic\jdk160_18\bin>netstat -ano|find "7001"|find "LISTENING"

TCP	127.0.0.1:7001	0.0.0.0:0	LISTENING	17348
TCP	169.254.98.86:7001	0.0.0.0:0	LISTENING	17348
TCP	172.16.75.1:7001	0.0.0.0:0	LISTENING	17348
TCP	172.16.220.1:7001	0.0.0.0:0	LISTENING	17348
TCP	192.168.2.83:7001	0.0.0.0:0	LISTENING	17348
TCP	[::1]:7001	:::0	LISTENING	17348
TCP	[fe80::3033:663b:7b8c:95b1%7]:7001	:::0	LISTENING	17348
TCP	[fe80::4c84:63d8:854a:235a%5]:7001	:::0	LISTENING	17348
TCP	[fe80::697d:fde2:15b7:8d2e%4]:7001	:::0	LISTENING	17348
TCP	[fe80::943a:e902:f20b:90e0%11]:7001	:::0	LISTENING	17348
TCP	[fe80::953d:6390:9070:5a03%9]:7001	:::0	LISTENING	17348
TCP	[fe80::c9dc:915b:3b84:6256%3]:7001	:::0	LISTENING	17348
TCP	[fe80::f985:7aa2:b02c:aeda%2]:7001	:::0	LISTENING	17348

F:\WebLogic\jdk160_18\bin>jstack 17348 >17348.log

F:\WebLogic\jdk160_18\bin>

此电脑 > 新加卷 (F:) > WebLogic > jdk160_18 > bin

名称	修改日期	类型	大小
17348.log	2017/7/20 9:35	文本文档	41 KB
beanreg.dll	2009/12/17 17:13	应用程序扩展	29 KB

或者：输入命令行：jstack -F 17348 >test.log

命令提示符

Microsoft Windows [版本 10.0.15063]
(c) 2017 Microsoft Corporation。保留所有权利。

C:\Users\JOAN>F:

F:\>cd F:\WebLogic\jdk160_18\bin

F:\WebLogic\jdk160_18\bin>netstat -ano|find "7001"|find "LISTENING"

TCP	127.0.0.1:7001	0.0.0.0:0	LISTENING	17348
TCP	169.254.98.86:7001	0.0.0.0:0	LISTENING	17348
TCP	172.16.75.1:7001	0.0.0.0:0	LISTENING	17348
TCP	172.16.220.1:7001	0.0.0.0:0	LISTENING	17348
TCP	192.168.2.83:7001	0.0.0.0:0	LISTENING	17348
TCP	[::1]:7001	:::0	LISTENING	17348
TCP	[fe80::3033:663b:7b8c:95b1%7]:7001	:::0	LISTENING	17348
TCP	[fe80::4c84:63d8:854a:235a%5]:7001	:::0	LISTENING	17348
TCP	[fe80::697d:fde2:15b7:8d2e%4]:7001	:::0	LISTENING	17348
TCP	[fe80::943a:e902:f20b:90e0%11]:7001	:::0	LISTENING	17348
TCP	[fe80::953d:6390:9070:5a03%9]:7001	:::0	LISTENING	17348
TCP	[fe80::c9dc:915b:3b84:6256%3]:7001	:::0	LISTENING	17348
TCP	[fe80::f985:7aa2:b02c:aeda%2]:7001	:::0	LISTENING	17348

F:\WebLogic\jdk160_18\bin>jstack 17348 >17348.log

F:\WebLogic\jdk160_18\bin>jstack -F 17348 >test.log

F:\WebLogic\jdk160_18\bin>

此电脑 > 新加卷 (F:) > WebLogic > jdk160_18 > bin

名称	修改日期	类型	大小
test.log	2017/7/20 9:38	文本文档	1 KB
17348.log	2017/7/20 9:35	文本文档	41 KB
beanreg.dll	2009/12/17 17:13	应用程序扩展	29 KB
jli.dll	2009/12/17 17:13	应用程序扩展	76 KB

二、堆Dump文件

1、同“一、线程信息”打印线程信息步骤一样，需要先进入对应的JDK的bin目录下，通过jps或netstat获取Java进程号：

2、在JDK的bin目录运行【jmap -histo:live 进程号 >进程号.map】

假设获取到的进程号是17348，则输入命令行为：jmap -histo:live 17348 >17348.map

F:\WebLogic\jdk160_18\bin>jmap -histo:live 17348 >17348.map

3、反馈文件以跟踪问题

此电脑 > 新加卷 (F:) > WebLogic > jdk160_18 > bin

名称	修改日期	类型	大小
17348.map	2017/7/20 10:10	MAP 文件	524 KB
17348.log	2017/7/20 9:35	文本文档	41 KB
beanreg.dll	2009/12/17 17:13	应用程序扩展	29 KB




4、若条件允许，生成整个堆Dump文件【jmap -dump:live,format=b,file=进程号.bin 进程号】

如输入命令行：jmap -dump:live,format=b,file=17348.bin 17348

```
F:\WebLogic\jdk160_18\bin>jmap -dump:live,format=b,file=17348.bin 17348
Dumping heap to F:\WebLogic\jdk160_18\bin\17348.bin ...
Heap dump file created
```

5、该文件非常大，与Java进程的运行内存一样的大，必须压缩以后再反馈

此电脑 > 新加卷 (F:) > WebLogic > jdk160_18 > bin

名称	修改日期	类型	大小
 17348.bin	2017/7/20 10:17	BIN 文件	116,996 KB
 17348.map	2017/7/20 10:10	MAP 文件	524 KB
 17348.log	2017/7/20 9:35	文本文档	41 KB

注：反馈以上的信息有助于解决系统无响应的问题。

三、GC参数

1、在JDK的bin目录中运行【jstat -gcutil 进程号 5000 1000】

```
C:\jdk1.8.0_40\bin>jstat -gcutil 720 5000 1000
```

2、若最后一段GCT段（GC所花费的总时间）在持续快速增加则表示服务器的内存已经不足，若只是少量增加是正常现象：

0.00	0.00	43.32	97.21	93.86	28	2.243	6	4.419	6.662
0.00	0.00	43.32	97.21	93.86	28	2.243	6	4.419	6.662
0.00	0.00	43.32	97.21	93.86	28	2.243	6	4.419	6.662
0.00	0.00	43.32	97.21	93.86	28	2.243	6	4.419	6.662
0.08	100.00	54.70	96.79	93.14	29	2.718	6	35.964	38.682
0.08	100.00	95.42	96.79	93.14	29	2.718	6	35.964	38.682
0.00	0.00	42.63	96.79	93.14	30	2.725	8	40.009	42.734
0.09	100.00	95.42	96.79	93.14	31	3.163	8	43.693	46.856
0.14	0.00	42.63	96.80	93.14	33	4.069	10	47.397	51.466
0.14	0.00	42.63	96.80	93.14	33	4.069	10	47.397	51.466
0.14	0.00	42.63	96.80	93.14	33	4.069	10	47.397	51.466
0.14	0.00	42.63	96.80	93.14	33	4.069	10	47.397	51.466
0.14	0.00	42.63	96.80	93.14	33	4.069	10	47.397	51.466
0.14	0.00	42.63	96.80	93.14	33	4.069	10	47.397	51.466
0.14	0.00	42.63	96.80	93.14	33	4.069	10	47.397	51.466
0.14	0.00	42.63	96.80	93.14	33	4.069	10	47.397	51.466
0.14	0.00	42.63	96.80	93.14	33	4.069	10	47.397	51.466
0.14	0.00	42.63	96.80	93.14	33	4.069	10	47.397	51.466
0.00	0.00	42.63	96.80	93.14	34	4.075	11	99.245	103.320
0.69	100.00	95.42	96.85	93.22	35	4.483	12	103.894	108.377
0.22	0.00	42.64	96.85	93.22	37	5.372	13	107.698	113.070
0.22	0.00	42.64	96.85	93.22	37	5.372	13	107.698	113.070
0.22	0.00	42.64	96.85	93.22	37	5.372	13	107.698	113.070
0.22	0.00	42.64	96.85	93.22	37	5.372	13	107.698	113.070
0.22	0.00	42.64	96.85	93.22	37	5.372	13	107.698	113.070

Linux

一、线程信息

执行命令行生成日志：

1) 另外启动一个xshell连接窗口，进入到JDK的路径下（不知道如何选择适合的JDK，可看最下边备注信息），如：

/home/admin				
名字	扩展	大小	已改变	权限
.thumbnails			2016/4/18 18:17:...	rwX-----
firefox			2016/6/21 5:05:07	rwXrwxr-x
IBM			2014/12/30 23:3...	rwXr-xr-x
InforSuite			2016/10/7 5:04:02	rwXrwxr-x
jdk1.6.0_26			2015/8/24 23:00:...	rwXrwxrw.
jdk1.8.0_101			2016/6/22 18:13:...	rwXr-xr-x
Smartbi			2016/9/23 18:30:...	rwXr-xr-x
Smartbi_bak			2014/11/11 10:2...	rwXr-xr-x
Smartbi_new			2016/4/8 20:20:10	rwXr-xr-x
SNX			2016/2/9 14:39:04	rwXr-xr-x
test			2017/1/25 17:25:...	rwXr-xr-x
tomcat			2016/9/23 1:23:58	rwXr-xr-x
weblogic			2015/6/30 23:40:...	rwXr-xr-x
weblogic1033			2015/8/24 23:04:...	rwXr-xr-x

在这个截图是进入到 /home/admin/jdk1.6.0_26/bin 目录下

```

root@smartbiserver20:/home/admin/jdk1.6.0_26/bin
login as: root
root@192.168.1.14's password:
[root@smartbiserver20 ~]# cd /home/admin/jdk1.6.0_26/bin
[root@smartbiserver20 bin]# jps
41118 Jps
40507 Server
[root@smartbiserver20 bin]# jstack 40507
2017-02-18 03:43:22
Full thread dump Java HotSpot(TM) 64-Bit Server VM (20.1-b02 mixed mode):

"Attach Listener" daemon prio=10 tid=0x00007f7a40001000 nid=0xa0ea waiting on co
ndition [0x0000000000000000]
    java.lang.Thread.State: RUNNABLE

"Timer-6" daemon prio=10 tid=0x00007f7a1c1d0000 nid=0x9fa6 in Object.wait() [0x0
0007f79fd7c2000]
    java.lang.Thread.State: TIMED_WAITING (on object monitor)
        at java.lang.Object.wait(Native Method)
        - waiting on <0x00000000fd010098> (a java.util.TaskQueue)
        at java.util.TimerThread.mainLoop(Timer.java:509)
        - locked <0x00000000fd010098> (a java.util.TaskQueue)
        at java.util.TimerThread.run(Timer.java:462)

"DefaultQuartzScheduler_QuartzSchedulerThread" prio=10 tid=0x00007f7a1c1ce000 ni

```

2) 输入 jps 命令，Server 代表的是weblogic的线程号

```
root@smartbiserver20:/home/admin/jdk1.6.0_26/bin
login as: root
root@192.168.1.14's password:
[root@smartbiserver20 ~]# cd /home/admin/jdk1.6.0_26/bin
[root@smartbiserver20 bin]# jps
41118 Jps
40507 Server
[root@smartbiserver20 bin]# jstack 40507
2017-02-18 03:43:22
Full thread dump Java HotSpot(TM) 64-Bit Server VM (20.1-b02 mixed mode):

"Attach Listener" daemon prio=10 tid=0x00007f7a40001000 nid=0xa0ea waiting on co
ndition [0x0000000000000000]
    java.lang.Thread.State: RUNNABLE

"Timer-6" daemon prio=10 tid=0x00007f7a1c1d0000 nid=0x9fa6 in Object.wait() [0x0
0007f79fd7c2000]
    java.lang.Thread.State: TIMED_WAITING (on object monitor)
        at java.lang.Object.wait(Native Method)
        - waiting on <0x00000000fd010098> (a java.util.TaskQueue)
        at java.util.TimerThread.mainLoop(Timer.java:509)
        - locked <0x00000000fd010098> (a java.util.TaskQueue)
        at java.util.TimerThread.run(Timer.java:462)

"DefaultQuartzScheduler_QuartzSchedulerThread" prio=10 tid=0x00007f7a1c1ce000 ni
```

3) 在窗口继续输入 `jstack 40507` (就是输入第二步命令获取到的weblogic的线程号), 开始打印线程 (这一步也可以直接跳过进行第四步操作)

```
root@smartbiserver20:/home/admin/jdk1.6.0_26/bin
login as: root
root@192.168.1.14's password:
[root@smartbiserver20 ~]# cd /home/admin/jdk1.6.0_26/bin
[root@smartbiserver20 bin]# jps
41118 Jps
40507 Server
[root@smartbiserver20 bin]# jstack 40507
2017-02-18 03:43:22
Full thread dump Java HotSpot(TM) 64-Bit Server VM (20.1-b02 mixed mode):

"Attach Listener" daemon prio=10 tid=0x00007f7a40001000 nid=0xa0ea waiting on co
ndition [0x0000000000000000]
    java.lang.Thread.State: RUNNABLE

"Timer-6" daemon prio=10 tid=0x00007f7a1c1d0000 nid=0x9fa6 in Object.wait() [0x0
0007f79fd7c2000]
    java.lang.Thread.State: TIMED_WAITING (on object monitor)
        at java.lang.Object.wait(Native Method)
        - waiting on <0x00000000fd010098> (a java.util.TaskQueue)
        at java.util.TimerThread.mainLoop(Timer.java:509)
        - locked <0x00000000fd010098> (a java.util.TaskQueue)
        at java.util.TimerThread.run(Timer.java:462)

"DefaultQuartzScheduler_QuartzSchedulerThread" prio=10 tid=0x00007f7a1c1ce000 ni
```

4) 将线程信息输出, 如输出到home 路径下, 输入命令为 `jstack 40507 >> /home/2.txt`

```
root@smartbiserver20:/home/admin/jdk1.6.0_26/bin

le

"GC task thread#8 (ParallelGC)" prio=10 tid=0x00007f7a90029800 nid=0x9e45 runnab
le

"GC task thread#9 (ParallelGC)" prio=10 tid=0x00007f7a9002b800 nid=0x9e46 runnab
le

"GC task thread#10 (ParallelGC)" prio=10 tid=0x00007f7a9002d800 nid=0x9e47 runna
ble

"GC task thread#11 (ParallelGC)" prio=10 tid=0x00007f7a9002f000 nid=0x9e48 runna
ble

"GC task thread#12 (ParallelGC)" prio=10 tid=0x00007f7a90031000 nid=0x9e49 runna
ble

"VM Periodic Task Thread" prio=10 tid=0x00007f7a900bb000 nid=0x9e51 waiting on c
ondition

JNI global references: 1796

[root@smartbiserver20 bin]# jstack 40507 >> /home/2.txt
[root@smartbiserver20 bin]#
```

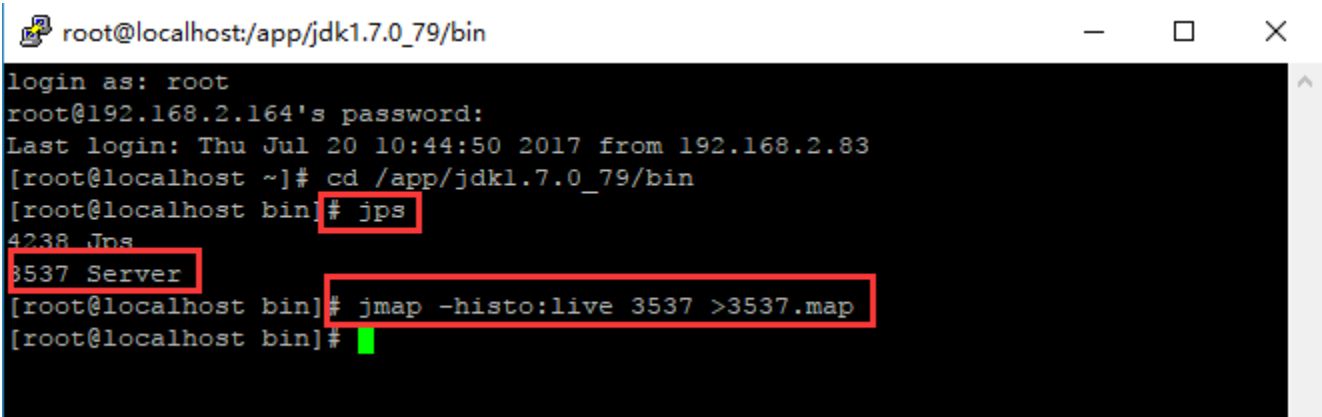
这时候就会在服务器上home目录下生成对应的文件了，如下图，将这个文件发回分析。

home				
/home				
名字	扩展	大小	已改变	权限
..			2017/2/13 19:36:...	r-xr-xr-x
.Trash-0			2016/2/9 14:38:06	rwX-----
aaaBBB			2017/1/24 23:06:...	rwX-----
admin			2017/2/13 20:57:...	rwXrwXrwX
gpdb			2016/9/27 18:26:...	rwX-----
huangping			2017/1/24 23:02:...	rwX-----
lost+found			2014/12/30 1:31:...	rwX-----
OpenOffice			2015/6/5 19:03:38	rwXr-xr-x
oracle			2016/4/7 17:55:14	rwX-----
1.txt		0 B	2016/9/22 2:46:59	rw-r--r--
2.txt		34,806 B	2017/2/18 3:49:23	rw-r--r--
hello.sh		28 B	2015/7/10 22:50:...	rw-r--r--

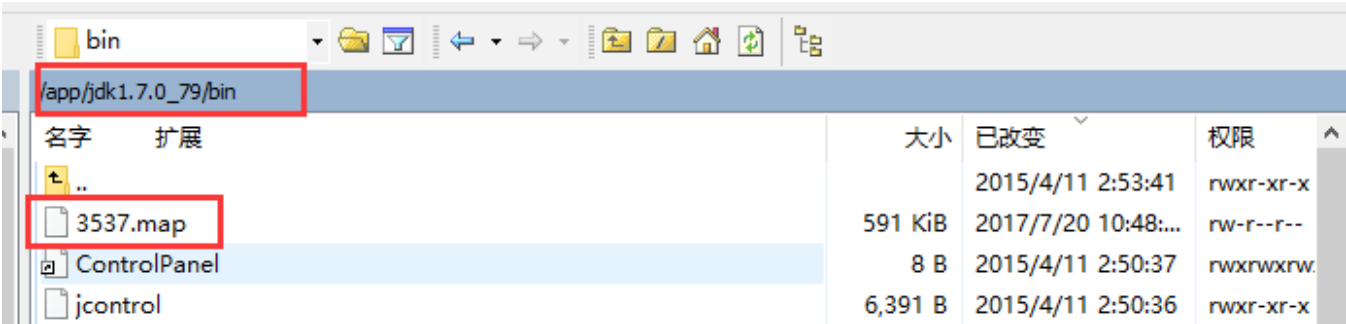
二、堆Dump文件

- 1、同“一、线程信息”打印线程信息步骤一样，需要先进入对应的JDK的bin目录下，通过jps获取Java进程号，Server 代表的是weblogic的线程号；
- 2、在JDK的bin目录运行【jmap -histo:live 进程号 >进程号.map】

假设获取到的进程号是3537，则输入命令行为：jmap -histo:live 3537 >3537.map

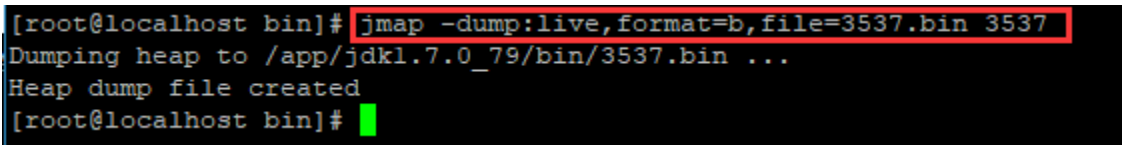


- 3、反馈文件以跟踪问题

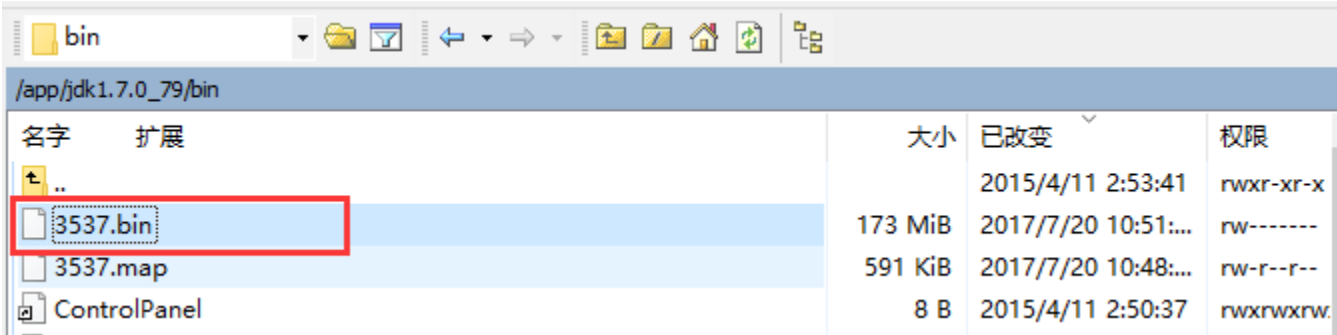


- 4、若条件允许，生成整个堆Dump文件【jmap -dump:live,format=b,file=进程号.bin 进程号】

如输入命令行：jmap -dump:live,format=b,file=3537.bin 3537



- 5、该文件非常大，与Java进程的运行内存一样的大，必须压缩以后再反馈



注：反馈以上的信息有助于解决系统无响应的问题。

三、GC参数

- 1、在JDK的bin目录中运行【jstat -gcutil 进程号 5000 1000】

```
C:\jdk1.8.0_40\bin>jstat -gcutil 720 5000 1000
```

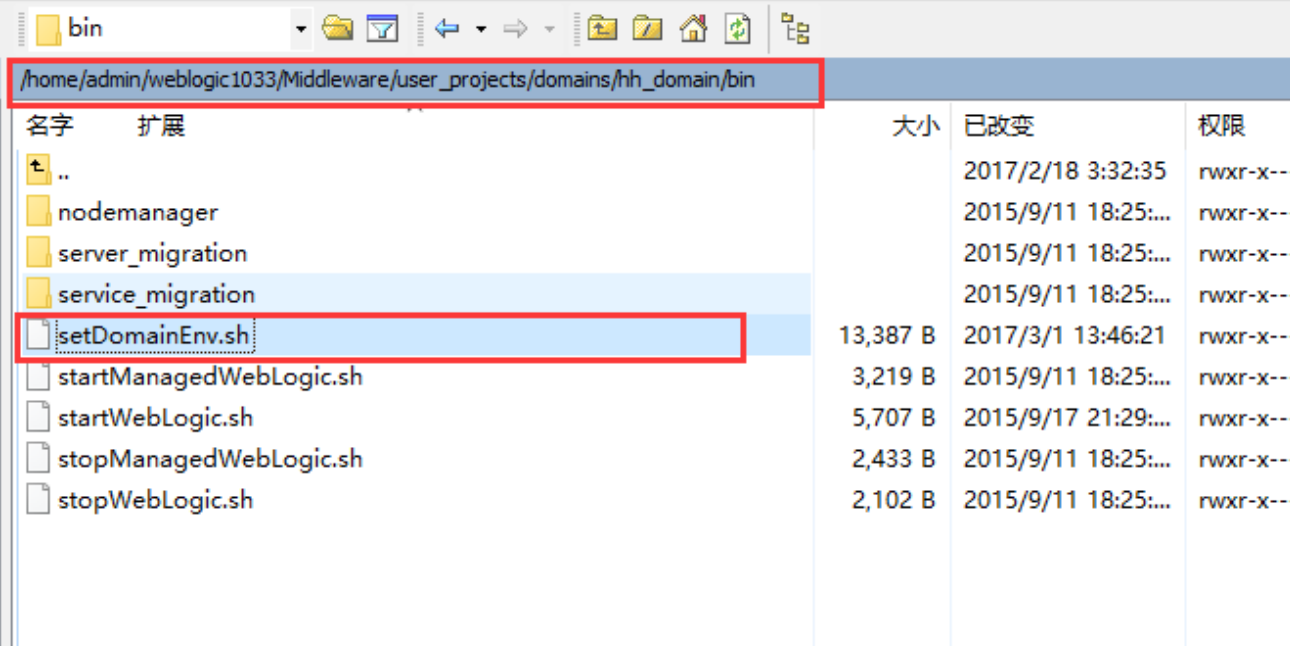
2、若最后一段GCT段（GC所花费的总时间）在持续快速增加则表示服务器的内存已经不足，若只是少量增加是正常现象：

[illegible]



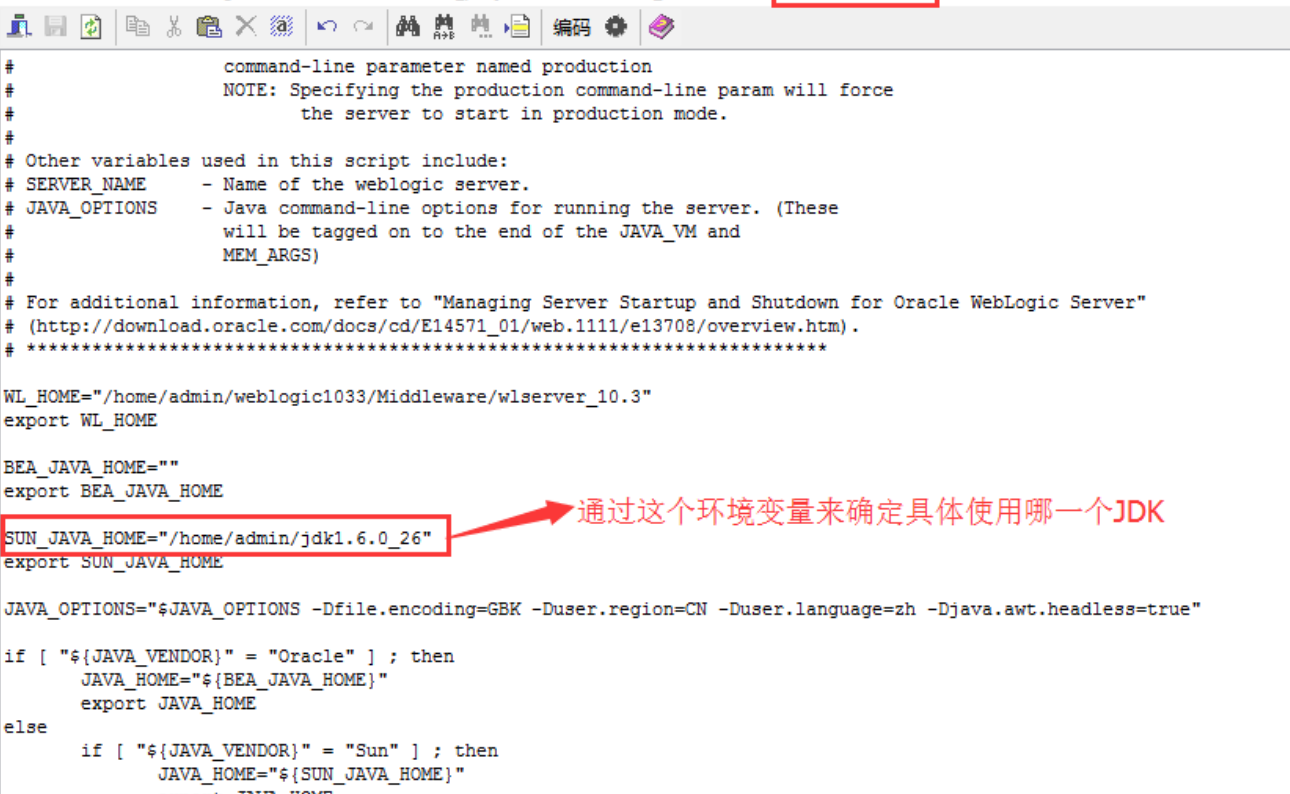
备注

如果没有进行修改过JDK的环境变量操作，就任何JDK都可以；如果不确定是否修改过，就直接使用正在运行的JDK，具体可通过smartbi应用所在域里边找到setDomainEnv.sh 文件查看，如下边截图是把smartbi部署在hh_domain域下的：/home/admin/weblogic1033/Middleware/user_projects/domains/hh_domain/bin



名字	扩展	大小	已改变	权限
..			2017/2/18 3:32:35	rwxr-x---
nodemanager			2015/9/11 18:25:...	rwxr-x---
server_migration			2015/9/11 18:25:...	rwxr-x---
service_migration			2015/9/11 18:25:...	rwxr-x---
setDomainEnv.sh		13,387 B	2017/3/1 13:46:21	rwxr-x---
startManagedWebLogic.sh		3,219 B	2015/9/11 18:25:...	rwxr-x---
startWebLogic.sh		5,707 B	2015/9/17 21:29:...	rwxr-x---
stopManagedWebLogic.sh		2,433 B	2015/9/11 18:25:...	rwxr-x---
stopWebLogic.sh		2,102 B	2015/9/11 18:25:...	rwxr-x---

/home/admin/weblogic1033/Middleware/user_projects/domains/hh_domain/bin/setDomainEnv.sh root@192.168.1.14



```
#
#      command-line parameter named production
#      NOTE: Specifying the production command-line param will force
#            the server to start in production mode.
#
# Other variables used in this script include:
# SERVER_NAME      - Name of the weblogic server.
# JAVA_OPTIONS     - Java command-line options for running the server. (These
#                    will be tagged on to the end of the JAVA_VM and
#                    MEM_ARGS)
#
# For additional information, refer to "Managing Server Startup and Shutdown for Oracle WebLogic Server"
# (http://download.oracle.com/docs/cd/E14571_01/web.1111/e13708/overview.htm).
# *****

WL_HOME="/home/admin/weblogic1033/Middleware/wlserver_10.3"
export WL_HOME

BEA_JAVA_HOME=""
export BEA_JAVA_HOME

SUN_JAVA_HOME="/home/admin/jdk1.6.0_26"
export SUN_JAVA_HOME

JAVA_OPTIONS="$JAVA_OPTIONS -Dfile.encoding=GBK -Duser.region=CN -Duser.language=zh -Djava.awt.headless=true"

if [ "${JAVA_VENDOR}" = "Oracle" ] ; then
    JAVA_HOME="${BEA_JAVA_HOME}"
    export JAVA_HOME
else
    if [ "${JAVA_VENDOR}" = "Sun" ] ; then
        JAVA_HOME="${SUN_JAVA_HOME}"
        export JAVA_HOME
    fi
fi
```