WebSphere打印线程信息和堆Dump文件

利用wsadmin生成javacore,heapdump文件



windows:

一、生成javacore文件

1、启动wsadmin.bat ,路径为: /opt/IBM/WebSphere/AppServer/profiles/AppSrv01/bin/

🕌 bin				
G → ↓ + 计算机 + 本地磁盘 (D:) + IBM ·	• WebSphere • AppServer • profiles • AppSrvO1	→ bin	▼ 🚱 搜索 bin	
组织 ▼ 📾 打开 打印 新建文件夹				:= - 🗐 😧
🕌 ChDw 🔟	名称 🔺	修改日期	类型	大小
🍶 daochu	👒 waslogbr. bat	2015/6/9 10:40	Windows 批处埋	1 KB
🎉 DbVisualizer=8.0.7	🚳 WASPostUpgrade. bat	2015/6/9 10:40	Windows 批处理	1 KB
JBM	🚳 WASPreUpgrade. bat	2015/6/9 10:40	Windows 批处理	1 KB
🕌 WebSphere	🚳 wasprofile. bat	2015/6/9 10:40	Windows 批处理	1 KB
7.0.0.31-WS-UPDI-WinIA32	🚳 weslogbr. bat	2015/6/9 10:40	Windows 批处理	1 KB
🎍 AppServer	🚳 WebSphereConnectJDBCDriverConversi	2015/6/9 10:40	Windows 批处理	1 KB
bin 🖉	🍇 wrd. bat	2015/6/9 10:40	Windows 批处理	1 KB
b configuration	🚳 wrd-config. bat	2015/6/9 10:40	Windows 批处理	1 KB
deploytool	🚳 ws_ant. bat	2015/6/9 10:40	Windows 批处理	1 KB
je derby	🚳 wsadmin. bat	2015/6/9 10:40	Windows 批处理	1 KB
	sappid. bat	2015/6/9 10:40	Windows 批处理	1 KB
ja eturas	wsdb2gen, bat	2015/6/9 10:40	Windows 批外理	1 KB
Tratatens	💽 wsdenlov bet	2015/6/9 10:40	Windows 批处理	1 KB
installableAnns	WSDI 2 Torre bet	2015/6/9 10:40	Windows 脱处理	1 KB
installedConnectors		2015/6/0 10:40		1 170
java	wsennancer. bat	2015/6/9 10.40	WINGOWS HUXUTE	1 KB
lafiles	wsgen. bat	2015/6/9 10:40	mindows 祝知理	1 KB
🚡 1ів	wsimport.bat	2015/6/9 10:40	Windows 批处埋	1 KB
linke 🔽	👒 wsjpaversion. bat	2015/6/9 10:40	Windows 批处理	1 KB 💌
wsadmin.bat 修改日期: 20 Windows 批处理文件 大小: 210 创建日期: 20	15/6/9 10:40 3 字节 15/6/12 19:49			

输入命令行: cd /opt/IBM/WebSphere/AppServer/profiles/AppSrv01/bin

./wsadmin.bat

管理员:C:\Tindovs\system32\cmd. exe - vsadmin. bat	
Microsoft Windows [版本 6.1.7601] 版权所有 <c> 2009 Microsoft Corporation。保留所有权利。</c>	
C:\Users\Administrator>d:	
D:\>cd D:\IBM\WebSphere\AppServer\profiles\AppSrv01\bin	
D:\IBM\WebSphere\AppServer\profiles\AppSrv01\bin <mark></mark> >wsadmin.bat WASX7209I: Connected to process "server1" on node HUANGPINGNode01 using SOAP connector; T WASX7029I: For help, enter: "\$Help help" wsadmin>_	he type of proces
	• •

2、输入命令 wsadmin>set jvm [\$AdminControl completeObjectName type=JVM,process=<server1>,*]

说明:上面中的<server1>要换成现场was服务的真实名称

如果websphere是集群部署环境,通过process查找时,会有多个java进程匹配,此时可以增加条件判断条件过滤出需要打印线程的进程。

wsadmin>set jvm [\$AdminControl completeObjectName type=JVM, process=<服务器>, platform=proxy.node=<节点>,*]

如下图,可以匹配服务器和节点名称来获取需要打印线程的进程。

 長和跟踪					2
记录和限算 使用此页面来指定服务器	如何处理日志记录。可以选择应用程序	服务器以启用或禁用该服	务器的系统日志,指定将日志制	取据存储在何处,并选择日志 和	9客的格式。还可以指定组件和组件组的
志详细信息级别。 ④ 首选项					
***					Marrie Barris
服务器 \$	节点◇	主机名 🗘	版本 💲	类型 🗘	状态 🛛
您可以管理以下资源:	TERRITORIA CONTRACTORIA				
Nodeserver1	WEB01Node01	CWEB01	ND 8.5.5.0	servers	+
Nodeserver1	Gus CWEB02Node01	CWEB02	ND 8.5.5.0	servers	•
Nodeserver1	WEB03Node01	IN THE CWEBO3	ND 8.5.5.0	servers	•
dmar	SCWEB01CellManager01	CWEB01	ND 8.5.5.0	servers	
nodeagent	SCWEB01Node01	JULCWEB01	ND 8.5.5.0	servers	1111 + 1 (S) 1 (S) 1
nodeagent	SCWEB02Node01	CWEB02	ND 8.5.5.0	servers	+
A REAL PROPERTY OF A REAL PROPER			NDOCEO	servers	+

如下图,执行set命令,返回多个进程信息,说明匹配不成功。新增判断条件,最终返回一条我们指定的java进程信息,此时就可以打印堆或线程了

wsadminsset ivm [Sadmincontrol] completeshile average and a second se
WASX7026W: 字符串"type=JVM,platform=proxy,node=GLJSCWEB03Node01,"」 WebSphere:name=JVM,process=nodeagent,platform=proxy.node=GLJSCWEB03Node01, #T与 2 不同的 Weban,相对应,已差回第一个 MBean,
webSpheres Town (Administration of the state
wsadmin>set jum [Sadmincontrol completobjectName type>Jum process-NameUsNodeu.j2eeType>JM.j2EEServer-Modeserver1.version=8.5.5.0.type>JM.mbeanidentifier=JM.cell=alscwebolce101,spec=1.0 webSpherstmme=JM.process=Nadeserver1.platform=proxy.node=GLJSCWeb03Nodeu.j2eeType=JM,J2EEServer=Nodeserver1.version=8.5.5.0.type=JM.mbeanidentifier=JM.cell=alscweb01ce1101,spec=1.0 wsadmin>\$Admincontrol invoke \$jum dumpThreads
vsadmin:set objectName [sadmin:control queryNames WebSphere:type=JWM,process=Nodeserver1.platform=proxy.node=GLJSCWEB03Node01.*] 返回一个进程信息是正常的 webSphere:name=JVM,process=Nodeserver1.platform=proxy.node=GLJSCWEB03Node01.j2eeType=JVM,JZEEServer=Nodeserver1.version=8.5.5.0.type=JVM,mbeanIdentifier=JVM,cell=qLJSCWEB02Node01.e] wsadmin:Sadmin:control invoke sobjectName generateHeapOuxplateListISSLMEB03Node01.j2eeType=JVM,JZEEServer=Nodeserver1.version=8.5.5.0.type=JVM,mbeanIdentifier=JVM,cell=qLJSCWEB02Node02.e] /opt/ISM/WebSphere/AppServer/profiles/AppSrv01/./heapdump.2023L124.153123.109383.6042.phd wsadmin:

比如在服务器上对应的服务名是sever1 , 命令行输入是: set jvm [\$AdminControl completeObjectName type=JVM, process=server1,*]

管理员:C:\Tindows\system32\cmd.exe - wsadmin.bat	_ 🗆 🗵
Microsoft Windows [版本 6.1.7601] 版权的方式 -> 2000 Microsoft Company Line - 保密的方力利	^
加州 (C) 2009 Hicrosoft Corporation。休田所有权利。	
C:\Users\Administrator>d:	
D:\>cd D:\IBM\WebSphere\AppServer\profiles\AppSrv01\bin	
D: \IBM\WebSphere\AppServer\profiles\AppSrv01\bin>wsadmin.bat	
WASX72091: Connected to process "server1" on node HUANGPINGNode01 using SOAP connector; The type of p	roces
WARX/M291: For helm, enter: "SHelm helm"	
wsaamin/set jum lyhamintontroi completeunjectname type=jun,process=serveri,*j WebSnbene:name=JUM nuocess=senueni nlatfonm=nuovu node=HUANGPINGNodeØ1 j2eeTune=JUM J2FFSenuen=senueni	110100
wsadmin)_	,0013
4	

3、输入命令 wsadmin>\$AdminControl invoke \$jvm dumpThreads



4、安装目录D:\IBM\WebSphere\AppServer\profiles\AppSrv01下生成javacore文件(例: javacore. 20090804. 111947. 2632. txt)

hppSrv01						
	WebSphere • AppServer • profiles • AppSrv01	•	▼ 🔯 搜索 AppSr	v01		2
组织 👻 🧾 打开 👻 打印 新建文件夹					- 🔲 (?
🌗 bin 🗾	名称 🔺	修改日期	类型	大小		
🍌 configuration	🏨 firststeps	2015/6/9 10:40	又件夹			
🍑 deploytool	🕌 Font-smartbi	2015/6/9 10:58	文件夹			
🍑 derby	퉬 Index-smartbi	2016/3/4 17:30	文件夹			
🎉 dev 💳	퉬 installableApps	2015/6/9 10:40	文件夹			
it etc	퉬 installedApps	2015/6/9 10:40	文件夹			
🎍 features	퉬 installedConnectors	2015/6/9 10:40	文件夹			
🎍 firststeps	퉬 installedFilters	2015/6/9 10:42	文件夹			
🕌 installableApps	퉬 logs	2017/7/19 11:20	文件夹			
installedConnectors	properties	2015/6/9 10:40	文件夹			
java	Lemp	2017/7/19 11:22	文件夹			
Lafiles	tranlog	2015/6/12 18:08	文件卖			
		2017/7/19 11:22	文件文			
logs	iavacore 20170719 114117 10952 000	2017/7/19 11:41	· · · · · · · · · · · · · ·	1.771 KB		
 optionalLibraries	smarthi log	2015/10/9 15:00	マホマ档	66 KB		
plugins	smarthi properties	2015/10/9 14:47	ス本入口 PROPERTIES 文仕	1 KB		
profiles	smartbi war	2015/10/5 14:41	un 文件	177 070 20		
AppSrv01		2015/0/12 11.40		111,019 10		
📔 bin	smartbi-config.xml	2015/6/12 19:57	XML 又相	3 KB		
🛄 config 📃 🗖	🔄 Smartbi-License.xml	2016/12/20 16:22	XML 义相	19 KB		-
javacore. 20170719. 114117. 10	952.0 修改日期: 2017/7/19 11:41					
文本文档	大小: 1.72 MB					
	创建日期: 2017/7/19 11:41					

- 5、获取到javacore文件发回分析。
- 6、如何解析这个Javacore文件?

用javacore文件分析器对javacore文件进行分析

cd /d javacore文件分析器放置路径>java -Xmx1000m -jar jca31.jar

二、生成heapdump文件

注:具体步骤和生成Javacore文件类似,只是在命令行有差异:

1、启动wsadmin.bat ,路径为: D:\IBM\WebSphere\AppServer\profiles\AppSrv01\bin

输入命令行: cd D:\IBM\WebSphere\AppServer\profiles\AppSrv01\bin

./wsadmin.bat

2、输入命令 wsadmin>set objectName [\$AdminControl queryNames WebSphere:type=JVM, process=<server1>,*]

说明:上面中的<server1>要换成现场was服务的真实名称

比如在服务器上对应的服务名是server1 , 命令行输入是: set objectName [\$AdminControl queryNames WebSphere:type=JVM, process=server1, *]

3、输入命令 wsadmin> \$AdminControl invoke \$objectName generateHeapDump



4、安装目录D:\IBM\WebSphere\AppServer\profiles\AppSrv01下生成HeapDump文件(例: heapdump.20170719.141911.10952.0003.phd)

NppSrv01					
	▼ WebSphere → AppServer → profiles → AppSrv01	•	▼ 🛂 🛛 搜索 AppSr	v01	2
组织 ▼ 🕞 打开 包含到库中 ▼ 共享 ▼	新建文件夹				- 🗌 🕐
🛨 🌗 configuration 📃	名称 🔺	修改日期	类型	大小	▲
🕀 🍌 deploytool	🖐 Font-smartbi	2015/6/9 10:58	又件夹		
🕀 🌗 derby	🕌 Index-smartbi	2016/3/4 17:30	文件夹		
🛨 🍌 dev	퉬 installableApps	2015/6/9 10:40	文件夹		
🛨 퉲 etc 🥣	퉬 installedApps	2015/6/9 10:40	文件夹		
🕀 퉲 features	퉬 installedConnectors	2015/6/9 10:40	文件夹		
🕀 🌗 firststeps	퉬 installedFilters	2015/6/9 10:42	文件夹		
🌗 installableApps	logs	2017/7/19 11:20	文件夹		
🕀 🎍 installedConnectors	properties	2015/6/9 10:40	文件夹		
🛨 🎍 java	temp	2017/7/19 14:18	文件文		
🕌 lafiles	tranlar	2015/6/12 18:08	立件束		
🕀 🕌 lib		2013/0/12 10:00	文件天		
jinks		2011/1/19 14.10		10, 175, 175	
🕂 🎍 logs	heapdump. 20170719. 141911. 10952. 000	2017/7/19 14:19	PHD X14	16,415 KB	
🕀 👑 optionalLibraries	javacore. 20170719. 114117. 10952. 000	2017/7/19 11:41	文本文档	1,771 KB	
🕀 👑 plugins	smartbi.log	2015/10/9 15:00	文本文档	66 KB	
🗆 🎽 profiles	📄 smartbi.properties	2015/10/9 14:47	PROPERTIES 文件	1 KB	
E 🍌 AppSrv01	🔳 smartbi.war	2015/6/12 17:46	WAR 文件	177,879 KB	
📕 bin	🖭 smartbi-config.xml	2015/6/12 19:57	XML 文档	3 KB	
🗄 👑 config	🔮 Smartbi-License. xml	2016/12/20 16:22	XML 文档	19 KB	-
hin					

修改日期: 2015/6/15 11:10

5、将获取都到的heapdump文件发回分析即可。

文件夹

6、如何分析heapdump文件?

用heapdump文件分析器对heapdump文件进行分析

cd /d heapdump文件分析器放置路径>java -Xmx1000m -jar ha33.jar

三、GC日志

在Websphere 中,如果想让GC服务器自动GC,可以做如下设置:

在Websphere的控制台, server-->server1--->Java and Process Management --->Process definition-->Java Virtual Machine,

在这个页面把 "Verbose garbage collection " 勾上,

然后,在"Generic JVM arguments"那里添加如下参数: -verbose:gc -XX:+PrintGCDetails -Xverbosegclog:gc.log,

GC日志就会自动生成在你的server1 目录下了,可以通过一些工具来分析Websphere服务器的GC情况了。

Linux

一、生成javacore文件

1、启动wsadmin.sh, 路径为: /opt/IBM/WebSphere/AppServer/profiles/AppSrv01/bin/wsadmin.sh

/opt/IBM/WebSphere/AppServer/profiles/AppSrv01/bin				
名字 扩展 (1997)	大小	已改变	权限	^
wsenhancer.sh	95 B	2017/5/17 16:04:	rwxr-xr-x	
🗋 WSDL2Java.sh	94 B	2017/5/17 16:04:	rwxr-xr-x	
🗋 wsdeploy.sh	93 B	2017/5/17 16:04:	rwxr-xr-x	
🗋 wsdbgen.sh	92 B	2017/5/17 16:04:	rwxr-xr-x	
🗋 wsdb2gen.sh	93 B	2017/5/17 16:04:	rwxr-xr-x	
🗋 WSBatchPackager.sh	100 B	2017/5/17 16:04:	rwxr-xr-x	
🗋 wsappid.sh	92 B	2017/5/17 16:04:	rwxr-xr-x	
wsadmin.sh	92 B	2017/5/17 16:04:	rwxr-xr-x	
🗋 ws_ant.sh	91 B	2017/5/17 16:04:	rwxr-xr-x	
🗋 wrd-config.sh	95 B	2017/5/17 16:04:	rwxr-xr-x	
🗋 wrd.sh	88 B	2017/5/17 16:04:	rwxr-xr-x	
WebSphereConnectJDBCDriverConversion.sh	121 B	2017/5/17 16:04:	rwxr-xr-x	
🗋 wct.sh	106 B	2017/5/17 16:04:	rwxr-xr-x	
🗋 wasservice.sh	95 B	2017/5/17 16:04:	rwxr-xr-x	
🗋 wasprofile.sh	95 B	2017/5/17 16:04:	rwxr-xr-x	
🗋 WASPreUpgrade.sh	98 B	2017/5/17 16:04:	rwxr-xr-x	
WASPostUpgradeBLAHelper.sh	108 B	2017/5/17 16:04:	rwxr-xr-x	

输入命令行: cd /opt/IBM/WebSphere/AppServer/profiles/AppSrv01/bin

./wsadmin.sh

注: 输入命令行 ./wsadmin.sh 后, 需要输入WebSphere控制台的用户名和密码。



- 2、输入命令 wsadmin>set jvm [\$AdminControl completeObjectName type=JVM,process=<server1>,*]
 - 说明:上面中的<server1>要换成现场was服务的真实名称



比如在服务器上对应的服务名是sever1,命令行输入是: set jvm [\$AdminControl completeObjectName type=JVM, process=server1,*]

Proot@localhost:/opt/IBM/WebSphere/AppServer/profiles/AppSrv01/bin	—		\times
login as: root root@192.168.2.164's password: Last login: Tue Jul 18 18:03:20 2017 from 192.168.2.138 [root@localhost ~]# cd /opt/IBM/WebSphere/AppServer/profiles/AppSr [root@localhost bin]# ./wsadmin.sh 区域/单元名: <default></default>	v01/b	oin	~
Username: admin Password: WASX7209I: 已使用 SOAP 连接器连接到节点 localhostNode01 上的进程"s 的类型为: UnManagedProcess WASX7029I· 要获取帮助。请输入· <u>"SHelp belp"</u>	erver	:1″; 进	程
<pre>vsadmin>set jvm [\$AdminControl completeObjectName type=JVM,process wepSphere:name=JvM,process=server1,platform=proxy,node=localnostNo JVM,J2EEServer=server1,version=8.5.0.0,type=JVM,mbeanIdentifier=JV stNode01Cell,spec=1.0 wsadmin></pre>	=serv aeuı, M,cel	verl,*) j2eelj 1=loca	ype= alho

Proot@localhost:/opt/IBM/WebSphere/AppServer/profiles/AppSrv01/bin	_		\times
login as: root			~
root@192.168.2.164's password:			
Last login: Tue Jul 18 18:03:20 2017 from 192.168.2.138			
<pre>[root@localhost ~]# cd /opt/IBM/WebSphere/AppServer/profiles/App</pre>	pSrv01/k	oin	
[root@localhost bin]# ./wsadmin.sh 区域/单元名: <default></default>			
Username: admin			
Password:			
WASX7209I: 已使用 SOAP 连接器连接到节点 localhostNode01 上的进程 的类型为: UnManagedProcess	e"server	:1″;进	程
WASX7029I: 要获取帮助,请输入: `\$Help help″			
wsadmin>set jvm [\$AdminControl completeObjectName type=JVM,proc	ess=serv	verl,*]	
WebSphere:name=JVM,process=serverl,platform=proxy,node=localhos	tNode01,	j2eeTy	/pe=
JVM,J2EEServer=server1,version=8.5.0.0,type=JVM,mbeanIdentifier	=JVM,cel	ll=loca	lho
stNode01Cell.spec=1.0			
wsadmin>\$AdminControl invoke \$jvm dumpThreads			
wsadmin>			

4、获取生成的javacore文件路径为: /opt/IBM/WebSphere/AppServer/profiles/AppSrv01下生成javacore文件(例: javacore. 20090804. 111947. 2632. txt)

/opt/IBM/WebSphere/AppServer/profiles/AppSrv01				
名字 扩展 🎽	大小	已改变	权限	^
servers		2017/5/17 16:04:	rwxr-xr-x	
		2017/7/18 15:11:	rwxr-xr-x	
properties		2017/5/17 16:04:	rwxr-xr-x	
nlogs-smartbi		2017/5/17 18:25:	rwxr-xr-x	
logs		2017/5/17 17:37:	rwxr-xr-x	
installedFilters		2017/5/17 16:06:	rwxr-xr-x	
installedConnectors		2017/5/17 16:03:	rwxr-xr-x	
installedApps		2017/5/17 16:04:	rwxr-xr-x	
installableApps		2017/5/17 16:03:	rwxr-xr-x	
Index-smartbi		2017/7/18 18:00:	rwxr-xr-x	
Font-smartbi		2017/5/17 18:24:	rwxr-xr-x	
		2017/5/17 16:04:	rwxr-xr-x	
exts-smartbi		2017/7/18 18:00:	rwxr-xr-x	
expandedBundles		2017/5/17 16:06:	rwxr-xr-x	
etc		2017/5/17 16:04:	rwxr-xr-x	
- consolepreferences		2017/5/17 16:08:	rwxr-xr-x	
configuration		2017/5/17 16:06:	rwxr-xr-x	
config		2017/5/17 16:06:	rwxr-xr-x	
bin		2017/5/17 18:24:	rwxr-xr-x	
smartbi-config.xml	1,994 B	2017/5/17 18:22:	rw-rr	
smartbi.properties	240 B	2017/5/17 18:25:	rw-rr	
javacore.20170718.181350.13083.0083.txt	3,164 KiB	2017/7/18 18:13:	rw-rr	¥
<			>	

5、将javacore文件发回分析即可。

```
6、如何分析Javacore文件?
```

用javacore文件分析器对javacore文件进行分析

cd /d javacore文件分析器放置路径>java -Xmx1000m -jar jca31.jar

注: 若通过以上方法无法生成 javacore文件,可通过以下两行命令生成:

[root@smartbiserver20 ~]# ps -ef|grep websphere

找到需要打印线程的WebSphere线程号:

🗬 root@localhost:~

login as: root

root@192.168.2.164's password: Last login: Tue Jul <u>18 18:32:43 2017 from 192.1</u>68.2.138 lhos ~]**#** ps -ef|grep websphere p[root 🔒 root 13083 1 14 17:59 pts/0 00:05:29 /opt/IBM/WebSphere/AppServer/j ava/bin/java -peclipse.security -Dwas.status.socket=38172 -Dosgi.install.area=/o pt/IBM/WebSphere/AppServer -Dosgi.configuration.area=/opt/IBM/WebSphere/AppServe r/profiles/AppSrv01/servers/server1/configuration -Dosgi.framework.extensions=co m.ibm.cds,com.ibm.ws.eclipse.adaptors -Xshareclasses:name=websphe v85 1.6 64 %g ,nonFatal -Xbootclasspath/p:/opt/IBM/WebSphere/AppServer/java/jre/lib/ibmorb.jar -classpath /opt/IBM/WebSphere/AppServer/profiles/AppSrv01/properties:/opt/IBM/W ebSphere/AppServer/properties:/opt/IBM/WebSphere/AppServer/lib/startup.jar:/opt/ IBM/WebSphere/AppServer/lib/bootstrap.jar:/opt/IBM/WebSphere/AppServer/lib/jsf-n ls.jar:/opt/IBM/WebSphere/AppServer/lib/lmproxy.jar:/opt/IBM/WebSphere/AppServer /lib/urlprotocols.jar:/opt/IBM/WebSphere/AppServer/deploytool/itp/batchboot.jar: /opt/IBM/WebSphere/AppServer/deploytool/itp/batch2.jar:/opt/IBM/WebSphere/AppSer bsphere.internalClassAccessMode=allow -Xms1024m ver/java/lib/tools.jar -Dibm.we Xmx2048m -Xcompressedrefs -Xscmaxaot4M -Xscmx60M -Dws.ext.dirs=/opt/IBM/WebSpher e/AppServer/java/lib:/opt/IBM/WebSphere/AppServer/profiles/AppSrv01/classes:/opt /IBM/WebSphere/AppServer/classes:/opt/IBM/WebSphere/AppServer/lib:/opt/IBM/WebSp here/AppServer/installedChannels:/opt/IBM/WebSphere/AppServer/lib/ext:/opt/IBM/W ebSphere/AppServer/web/help:/opt/IBM/WebSphere/AppServer/deploytool/itp/plugins/ com.ibm.etools.ejbdeploy/runtime -Dderby.system.home=/opt/IBM/WebSphere/AppServe r/derby -Dcom.ibm.itp.location=/opt/IBM/WebSphere/AppServer/bin -Djava.util.logg 🗸

[root@smartbiserver20 ~]# kill -3 13083

🗬 root@localhost:~

– 🗆 🗙

 \times

Δ

r/derby -Dcom.ibm.itp.location=/opt/IBM/WebSphere/AppServer/bin -Djava.util.logg ∧ ing.configureByServer=true -Duser.install.root=/opt/IBM/WebSphere/AppServer/prof iles/AppSrv01 -Djava.ext.dirs=/opt/IBM/WebSphere/AppServer/tivoli/tam:/opt/IBM/W ebSphere/AppServer/java/jre/lib/ext -Djavax.management.builder.initial=com.ibm.w s.management.PlatformMBeanServerBuilder -Dpython.cachedir=/opt/IBM/WebSphere/App Server/profiles/AppSrv01/temp/cachedir -Dwas.install.root=/opt/IBM/WebSphere/App Server -Djava.util.logging.manager=com.ibm.ws.bootstrap.WsLogManager -Dserver.ro ot=/opt/IBM/WebSphere/AppServer/profiles/AppSrv01 -Dcom.ibm.security.jgss.debug= off -Dcom.ibm.security.krb5.Krb5Debug=off -Dfile.encoding=GBK -Duser.region=CN -Duser.language=zh -Djava.awt.headless=true -Djava.library.path=/opt/IBM/WebSpher e/AppServer/lib/native/linux/x86 64/:/opt/IBM/WebSphere/AppServer/java/jre/lib/a md64/default:/opt/IBM/WebSphere/AppServer/java/jre/lib/amd64:/opt/IBM/WebSphere/ AppServer/bin:/usr/lib: -Djava.endorsed.dirs=/opt/IBM/WebSphere/AppServer/endors ed apis:/opt/IBM/WebSphere/AppServer/java/jre/lib/endorsed:/opt/IBM/WebSphere/Ap pServer/endorsed_apis:/opt/IBM/WebSphere/AppServer/java/jre/lib/endorsed -Djava. security.auth.login.config=/opt/IBM/WebSphere/AppServer/profiles/AppSrv01/proper ties/wsjaas.conf -Djava.security.policy=/opt/IBM/WebSphere/AppServer/profiles/Ap pSrv0l/properties/server.policy com.ibm.wsspi.bootstrap.WSPreLauncher -nosplash application com.ibm.ws.bootstrap.WSLauncher com.ibm.ws.runtime.WsServer /opt/IB M/WebSphere/AppServer/profiles/AppSrv01/config localhostNode01Cell localhostNode 01 serverl 14640 14558 0 18:36 pts/1 root. 00:00:00 grep --color=auto

[root@localhost ~]# kill -3 13083 [root@localhost ~]#

-5 13085

/opt/IBM/WebSphere/AppServer/profiles/AppSrv01					
名字 分展 [×]		大小	已改变	权限	^
- properties			2017/5/17 16:04:	rwxr-xr-x	
nlogs-smartbi			2017/5/17 18:25:	rwxr-xr-x	
logs			2017/5/17 17:37:	rwxr-xr-x	
installedFilters			2017/5/17 16:06:	rwxr-xr-x	
installedConnectors			2017/5/17 16:03:	rwxr-xr-x	
installedApps			2017/5/17 16:04:	rwxr-xr-x	
installableApps			2017/5/17 16:03:	rwxr-xr-x	
Index-smartbi			2017/7/18 18:00:	rwxr-xr-x	
Font-smartbi			2017/5/17 18:24:	rwxr-xr-x	
firststeps			2017/5/17 16:04:	rwxr-xr-x	
exts-smartbi			2017/7/18 18:00:	rwxr-xr-x	
expandedBundles			2017/5/17 16:06:	rwxr-xr-x	
etc .			2017/5/17 16:04:	rwxr-xr-x	
consolepreferences			2017/5/17 16:08:	rwxr-xr-x	
configuration			2017/5/17 16:06:	rwxr-xr-x	
- config			2017/5/17 16:06:	rwxr-xr-x	
📙 bin			2017/5/17 18:24:	rwxr-xr-x	
🗋 smartbi-config.xml		1,994 B	2017/5/17 18:22:	rw-rr	
smartbi.properties	_	240 B	2017/5/17 18:25:	rw-rr	
iavacore.20170718.183841.13083.0235.txt		3,222 KiB	2017/7/18 18:38:	rw-rr	
javacore.20170718.182813.13083.0171.txt		3,222 KiB	2017/7/18 18:28:	rw-rr	
📄 javacore.20170718.182719.13083.0165.txt		3,222 KiB	2017/7/18 18:27:	rw-rr	¥
<				>	

二、生成heapdump文件

方法一:

注:具体步骤和Javacore文件生成差不多,只是差别在于命令:

1、 启动wsadmin.sh, 路径为: /opt/IBM/WebSphere/AppServer/profiles/AppSrv01/bin/wsadmin.sh

输入命令行: cd /opt/IBM/WebSphere/AppServer/profiles/AppSrv01/bin

./wsadmin.sh

注: 输入命令行 ./wsadmin.sh 后,需要输入WebSphere控制台的用户名和密码

2、输入命令 wsadmin>set objectName [\$AdminControl queryNames WebSphere:type=JVM, process=<server1>,*]

说明:上面中的<server1>要换成现场was服务的真实名称

比如在服务器上对应的服务名是sever1 ,命令行输入是: set objectName [\$AdminControl queryNames WebSphere:type=JVM,process=server1,*]

3、 输入命令 wsadmin>\$AdminControl invoke \$objectName generateHeapDump

注:如果复制命令不行,请手动输入命令行。

Proot@localhost:/opt/IBM/	NebSphere/AppServer/profile	es/AppSrv01/bin				-	٥	\times
[root@localhost bin]#	ps -ef grep java							
root 16263 16242	0 09:46 pts/0 00	:00:04 [java] <defunc< td=""><td>t></td><td></td><td></td><td></td><td></td><td></td></defunc<>	t>					
root 16398 1	23 09:47 pts/0 00	:03:34 /opt/IBM/WebSp	here/AppServer/java/bin,	/java -Declipse.security	-Dwas.status.socket=3450	4 -Dosgi.install.area	=/opt/IB	M/W
ebSphere/AppServer -Do	sgi.configuration.are	a=/opt/IBM/WebSphere/	AppServer/profiles/AppS:	rv01/servers/server1/con	figuration -Dosgi.framewo	rk.extensions=com.ibm	.cds,com	.ib
m.ws.eclipse.adaptors -Xshareclasses:name=webspherev85 1.6 64 %g,nonFatal -Xbootclasspath/p:/opt/IBM/WebSphere/AppServer/java/jre/lib/ibmorb.jar -classpath /opt/IBM/Web								
Sphere/AppServer/profiles/AppSrv01/properties:/opt/IBM/WebSphere/AppServer/properties:/opt/IBM/WebSphere/AppServer/lib/startup.jar:/opt/IBM/WebSphere/AppServer/lib/boot								
strap.jar:/opt/IBM/WebSphere/AppServer/lib/jsf-nls.jar:/opt/IBM/WebSphere/AppServer/lib/lmproxy.jar:/opt/IBM/WebSphere/AppServer/lib/urlprotocols.jar:/opt/IBM/WebSphere								
/AppServer/deploytool/itp/batchboot.jar:/opt/IBM/WebSphere/AppServer/deploytool/itp/batch2.jar:/opt/IBM/WebSphere/AppServer/java/lib/tools.jar -Dibm.websphere.internalC								
lassAccessMode=allow -Xms1024m -Xmx2048m -Xcompressedrefs -Xscmaxaot4M -Xscmx60M -Dws.ext.dirs=/opt/IBM/WebSphere/AppServer/java/lib:/opt/IBM/WebSphere/AppServer/profil								
es/AppSrv01/classes:/opt/IBM/WebSphere/AppServer/liBM/WebSphere/AppServer/lib:/opt/IBM/WebSphere/AppServer/li								/li
b/ext:/opt/IBM/WebSphere/AppServer/web/help:/opt/IBM/WebSphere/AppServer/deploytool/itp/plugins/com.ibm.etools.ejbdeploy/runtime -Dderby.system.home=/opt/IBM/WebSphere/								re/
AppServer/derby -Dcom.ibm.itp.location=/opt/IBM/WebSphere/AppServer/bin -Djawa.util.logging.configureByServer=true -Duser.install.root=/opt/IBM/WebSphere/AppServer/prof								
iles/AppSrv01 -D java .e	xt.dirs=/opt/IBM/WebS	phere/AppServer/tivol	i/tam:/opt/IBM/WebSphere	e/AppServer/java/jre/lib	/ext -D javax. management.b	uilder.initial=com.ib	m.ws.man	age
ment.PlatformMBeanServ	erBuilder -Dpython.ca	chedir=/opt/IBM/WebSp	here/AppServer/profiles,	AppSrv01/temp/cachedir	-Dwas.install.root=/opt/I	BM/WebSphere/AppServe		.ut
il.logging.manager=com	.ibm.ws.bootstrap.WsL	ogManager -Dserver.ro	ot=/opt/IBM/WebSphere/Ap	ppServer/profiles/AppSrv	01 -Dcom.ibm.security.jgs	s.debug=off -Dcom.ibm	.securit	y.k
rb5.Krb5Debug=off -Dfi	le.encoding=GBK -Duse:	r.region=CN -Duser.la	nguage=zh -D java .awt.hea	adless=true -D <mark>java</mark> .libra	ry.path=/opt/IBM/WebSpher	e/AppServer/lib/nativ	e/linux/	x86
64/:/opt/IBM/WebSpher	e/AppServer/java/jre/	lib/amd64/default:/op	t/IBM/WebSphere/AppServ	er <mark>/java</mark> /jre/lib/amd64:/o	pt/IBM/WebSphere/AppServe	r/bin:/usr/lib: -Djav	a.endors	ed.
dirs=/opt/IBM/WebSpher	e/AppServer/endorsed_	apis:/opt/IBM/WebSphe	re/AppServer/java/jre/1:	ib/endorsed:/opt/IBM/Web	Sphere/AppServer/endorsed	_apis:/opt/IBM/WebSph	ere/AppS	erv
er/java/jre/lib/endors	ed -Djava.security.au	th.login.config=/opt/	IBM/WebSphere/AppServer,	/profiles/AppSrv01/prope	rties/wsjaas.conf -D <mark>java</mark> .	security.policy=/opt/	IBM/WebS	phe
re/AppServer/profiles/	AppSrv01/properties/s	erver.policy com.ibm.	wsspi.bootstrap.WSPreLa	uncher -nosplash -applic	ation com.ibm.ws.bootstra	p.WSLauncher com.ibm.	ws.runti	me.
WsServer /opt/IBM/WebS	phere/AppServer/profil	les/AppSrv01/config 1	ocalhostNode01Cell loca	LhostNode01 server1				
root 17025 16539	0 10:02 pts/1 00	:00:00 grepcolor=a	uto java					
[root@localhost bin]#	./w							
wasprofile.sh	wrd-config.sh	ws_ant.sh	wsdbgen.sh	wsgen.sh	wsmapping.sh	wve_encodePassword.	sh	
wasservice.sh	wrd.sh	wsappid.sh	wsdeploy.sh	wsimport.sh	wsreversemapping.sh			
wct.sh	wsadmin.sh	wsdb2gen.sh	wsenhancer.sh	wsjpaversion.sh	wsschema.sh			
[root@localhost bin]# 区域/单元名: <default></default>	./wsadmin.sh							
Username: admin								
Password:								
WASX7209I: 已使用 SOAE	・连接器连接到节点 loca	alhostNode01 上的进程	"serverl": 进程的类型为	UnManagedProcess				
WASX7029I: 要获取帮助,	请输入: "\$Help help"							
wsadmin>set objectName	[\$AdminControl query]	Names WebSphere:type=	JVM, process=server1, *]					
WebSphere:name=JVM,pro	cess=server1,platform	=proxy,node=localhost	Node01, j2eeType=JVM, J2E	Server=server1,version=	8.5.0.0,type=JVM,mbeanIde	ntifier=JVM, cell=loca	lhostNod	e01
Cell, spec=1 0								
wsadmin>\$AdminControl invoke \$objectName generateHeapDump								
/opt/IBM/WebSphere/App	Server/profiles/AppSr	v01/./heapdump.201707	19.100404.16398.0097.ph					
wsadmin>								

4、安装目录 /opt/IBM/WebSphere/AppServer/profiles/AppSrv01 下生成HeapDump文件(例:heapdump.20170719.100404.16398.0097.phd)

/opt/IBM/WebSphere/AppServer/profiles/AppSrv01									
名字 扩展		大小	已改变	权限	^				
📙 mlogs-smartbi			2017/5/17 18:25:	rwxr-xr-x					
📙 logs		2017/5/17 17:37:	rwxr-xr-x						
installedFilters		2017/5/17 16:06:	rwxr-xr-x						
installedConnectors		2017/5/17 16:03:	rwxr-xr-x						
installedApps		2017/5/17 16:04:	rwxr-xr-x						
installableApps		2017/5/17 16:03:	rwxr-xr-x						
Index-smartbi			2017/7/19 9:47:57	rwxr-xr-x					
Font-smartbi			2017/5/17 18:24:	rwxr-xr-x					
firststeps			2017/5/17 16:04:	rwxr-xr-x					
exts-smartbi			2017/7/19 9:48:01	rwxr-xr-x					
expandedBundles			2017/5/17 16:06:	rwxr-xr-x					
etc			2017/5/17 16:04:	rwxr-xr-x					
consolepreferences			2017/5/17 16:08:	rwxr-xr-x					
- configuration			2017/5/17 16:06:	rwxr-xr-x					
- config			2017/5/17 16:06:	rwxr-xr-x					
			2017/5/17 18:24:	rwxr-xr-x					
smartbi-config.xml		1,994 B	2017/5/17 18:22:	rw-rr					
smartbi.properties		240 B	2017/5/17 18:25:	rw-rr					
javacore.20170718.183841.13083.0235.txt	3,222 KiB	2017/7/18 18:38:	rw-rr						
javacore.20170718.182813.13083.0171.txt	3,222 KiB	2017/7/18 18:28:	rw-rr						
javacore.20170718.182719.13083.0165.txt	3,222 KiB	2017/7/18 18:27:	rw-rr						
heapdump.20170719.100404.16398.0097.phd		30,643	2017/7/19 10:04:	rw-rr	¥				
<				>					

5、将heapdump文件发回分析即可。

6、如何对heapdump文件进行分析?

用heapdump文件分析器对heapdump文件进行分析

cd /d heapdump文件分析器放置路径>java -Xmx1000m -jar ha33.jar

方法二:

说明: WebSphere出现无响应的时cpu使用率为0,并会自动重启,导致无法通过"方法一"获取heapdump信息。可提前在WebSphere控制台上增加以下参数并重启服务器,这样下次进行生成Javacore文件时,会自动同时生成heapdump信息:

步骤:在WebSphere控制台, Servers(服务器)>WebSphere Application Server > server_name(如: server1)> Java和进程管理 > Process Definition (进程定义)> Environment Entries(环境条目)> New(新建) 增加以下信息

名称	值
IBM_HEAPDUMP	true
IBM_HEAP_DUMP	true
IBM_HEAPDUMPDIR	your_directory
IBM_HEAPDUMP_OUTOFMEMORY	true
IBM_JAVADUMP_OUTOFMEMORY	true

注:其中第三条参数是为了指定生成heapdump文件的路径,可不填写,这样就会默认生成在和javacore文件相同的路径下。 当进行生成Javacore后,就会同时生成heapdump文件。

三、GC日志

在Websphere 中,如果想让GC服务器自动GC,可以做如下设置:

在Websphere的控制台, server-->Server1--->Java and Process Management --->Process definition-->Java Virtual Machine,

在这个页面把 "Verbose garbage collection " 勾上,

然后,在"Generic JVM arguments"那里添加如下参数: -verbose:gc -XX:+PrintGCDetails -Xverbosegclog:gc.log,

GC日志就会自动生成在你的server1 目录下了,可以通过一些工具来分析Websphere服务器的GC情况了。