# 环境准备

## 中间件版本

jdk 1.8

kafka 0.10.2.1

## windows指标

windows采用wmi(Windows Management Instrumentation)采集，确保wmi服务开启。

## oracle指标

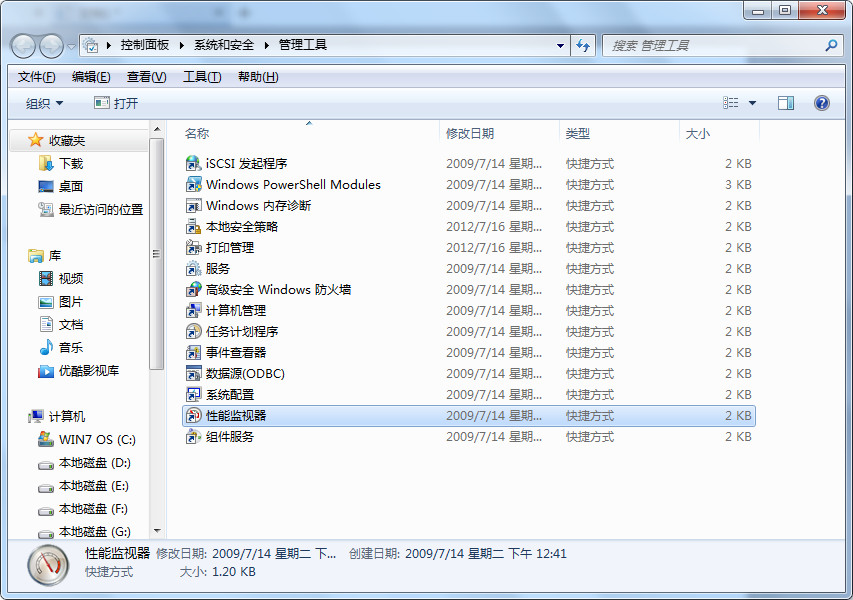
使用sql语句以及命令行，确保指标查询用户拥有相关查询权限。

## sqlserver指标

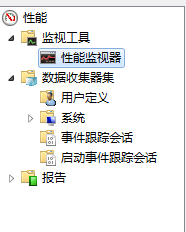
使用windows的性能监视器，开启步骤：

### 打开性能计数器

开始菜单——控制面板——管理工具——性能监视器

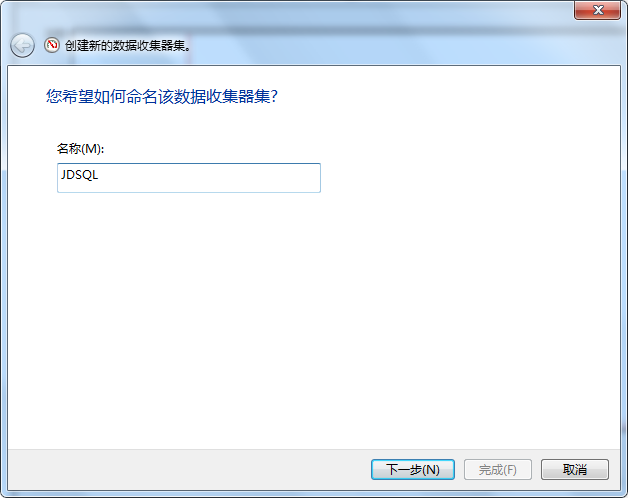


### 添加性能计数器

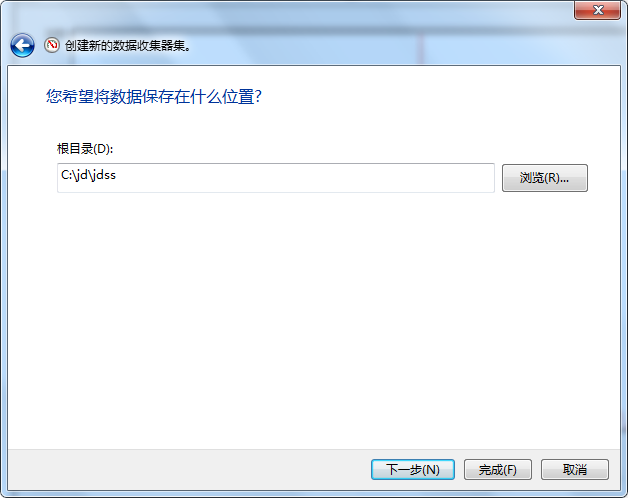


右键“性能监视器”——新建——数据收集器

名称为“JDSQL”



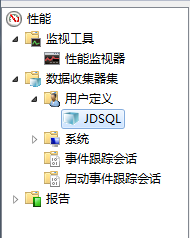
填写根目录



下一步，直到完成

### 设置性能计数器

此时可以在“数据收集器——用户自定义”目录下看到刚才新建的“JDSQL”数据收集器



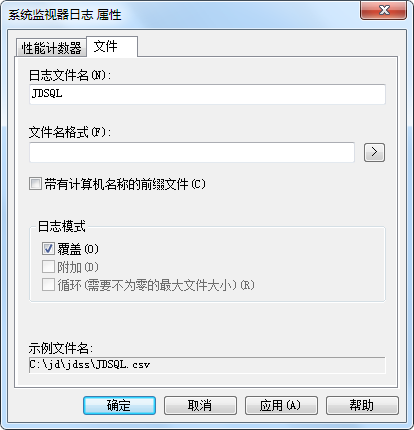
右键性能监视器日志——属性



日志格式改为“逗号分隔”

实例间隔为“1秒”

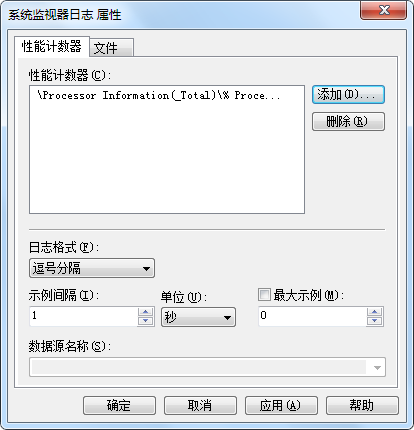
在“文件”里，日志文件名为“JDSQL”，日志模式选择“覆盖”



注意：conf.yaml中sqlserver下perfmon.path的值即为该示例文件名的值。

在“性能计数器”里，根据策略表，按顺序依次添加相关的SQL SERVER性能计数器。

具体的监控指标SQL SERVER性能计数器的值可参见“监控策略表”。



添加完成后，点“应用”，然后是“确定”。

### 启动性能计数器

右键“JDSQL”，点击“开始”

此时，在“C:jd/jdss/”目录下可以看到以下文件



说明计数器能够运行

右键“JDSQL”，点击“停止”。

## tomcat指标

### windows版本

#### 需要身份认证

* 将jmxremote.access上传到tomcat\conf\目录下
* 将jmxremote.password上传到tomcat\conf\目录下

密码文件应该是只读的，只能由Tomcat运行的操作系统用户访问。

如果tomcat是以Administrators用户启动的，可执行如下命令更改密码文件权限。以其他用户启动的，则修改“Administrators”为相应的用户名即可

* 执行命令：

cacls $CATALINA\_HOME/conf/jmxremote.password /P Administrators:R

cacls $CATALINA\_HOME/conf/jmxremote.access /P Administrators:R

如tomcat是通过startup.bat启动的修改catalina.bat文件。

如tomcat是通过tomcat.exe启动的修改tomcatw.exe中的JAVA参数。

* 修改catalina.bat文件，将以下内容插入catalina.bat中

|  |
| --- |
| set JMX\_REMOTE\_CONFIG=-Dcom.sun.management.jmxremote -Dcom.sun.management.jmxremote.port=10001 -Dcom.sun.management.jmxremote.ssl=false -Dcom.sun.management.jmxremote.authenticate=true  Dcom.sun.management.jmxremote.password.file=../conf/jmxremote.password  Dcom.sun.management.jmxremote.access.file=../conf/jmxremote.access  set CATALINA\_OPTS=%CATALINA\_OPTS% %JMX\_REMOTE\_CONFIG% |

* 双击tomcatw.exe，在java参数tab中加入以下内容

|  |
| --- |
| -Dcom.sun.management.jmxremote  -Dcom.sun.management.jmxremote.port=10001  -Dcom.sun.management.jmxremote.ssl=false  -Dcom.sun.management.jmxremote.authenticate=true  -Dcom.sun.management.jmxremote.password.file=../conf/jmxremote.password  -Dcom.sun.management.jmxremote.access.file=../conf/jmxremote.access |

#### 无需身份认证

如tomcat是通过startup.bat启动的修改catalina.bat文件。

如tomcat是通过tomcat.exe启动的修改tomcatw.exe中的JAVA参数。

* 修改catalina.bat文件，将以下内容插入catalina.bat中

|  |
| --- |
| set JMX\_REMOTE\_CONFIG=-Dcom.sun.management.jmxremote -Dcom.sun.management.jmxremote.port=10001 -Dcom.sun.management.jmxremote.ssl=false -Dcom.sun.management.jmxremote.authenticate=false  set CATALINA\_OPTS=%CATALINA\_OPTS% %JMX\_REMOTE\_CONFIG% |

* 双击tomcatw.exe，在java参数tab中加入以下内容

|  |
| --- |
| -Dcom.sun.management.jmxremote  -Dcom.sun.management.jmxremote.port=10001  -Dcom.sun.management.jmxremote.ssl=false -Dcom.sun.management.jmxremote.authenticate=false |

### linux版本

#### 需要身份认证

* 将jmxremote.access上传到tomcat/conf/目录下
* 将jmxremote.password上传到tomcat/conf/目录下

如tomcat是通过startup.sh启动的修改catalina.sh文件。

* 修改catalina.sh文件，将以下内容插入catalina.sh中

|  |
| --- |
| set JMX\_REMOTE\_CONFIG=-Dcom.sun.management.jmxremote -Dcom.sun.management.jmxremote.port=10001 -Dcom.sun.management.jmxremote.ssl=false -Dcom.sun.management.jmxremote.authenticate=true  Dcom.sun.management.jmxremote.password.file=../conf/jmxremote.password  Dcom.sun.management.jmxremote.access.file=../conf/jmxremote.access  set CATALINA\_OPTS=%CATALINA\_OPTS% %JMX\_REMOTE\_CONFIG% |

进入bin目录，上传setenv.sh文件，并授予可执行的权限

     文件内容如下：

|  |
| --- |
| #!/bin/sh  CATALINA\_OPTS="$CATALINA\_OPTS -Dcom.sun.management.jmxremote  -Dcom.sun.management.jmxremote.authenticate=true  -Dcom.sun.management.jmxremote.password.file=$CATALINA\_BASE/conf/jmxremote.password  -Dcom.sun.management.jmxremote.access.file=$CATALINA\_BASE/conf/jmxremote.access  -Dcom.sun.management.jmxremote.ssl=false" |

#### 无需身份认证

如tomcat是通过startup.sh启动的修改catalina.sh文件。

* 修改catalina.sh文件，将以下内容插入catalina.sh中

|  |
| --- |
| set JMX\_REMOTE\_CONFIG=-Dcom.sun.management.jmxremote -Dcom.sun.management.jmxremote.port=10001 -Dcom.sun.management.jmxremote.ssl=false -Dcom.sun.management.jmxremote.authenticate=false  set CATALINA\_OPTS=%CATALINA\_OPTS% %JMX\_REMOTE\_CONFIG% |

进入bin目录，上传setenv.sh文件，并授予可执行的权限

     文件内容如下：

|  |
| --- |
| #!/bin/sh  CATALINA\_OPTS="$CATALINA\_OPTS -Dcom.sun.management.jmxremote  -Dcom.sun.management.jmxremote.authenticate=false  -Dcom.sun.management.jmxremote.ssl=false" |

## apache指标

提取http://{ip}:{port}/server-status?auto里性能指标,确保agent可访问该地址。

## weblogic指标

### 说明

本例是以Weblogic9.2和 Weblogic10为例进行安装，其他如为9之前的版本可参考weblogic9,如后为10之后的版本请参考weblogic10

### 检查端口

本次安装Weblogic，需要开放SNMP协议端口，建议SNMP端口固定为udp 8161，同时SNMP AgentX的端口为tcp 7050，若端口被占用请更换。

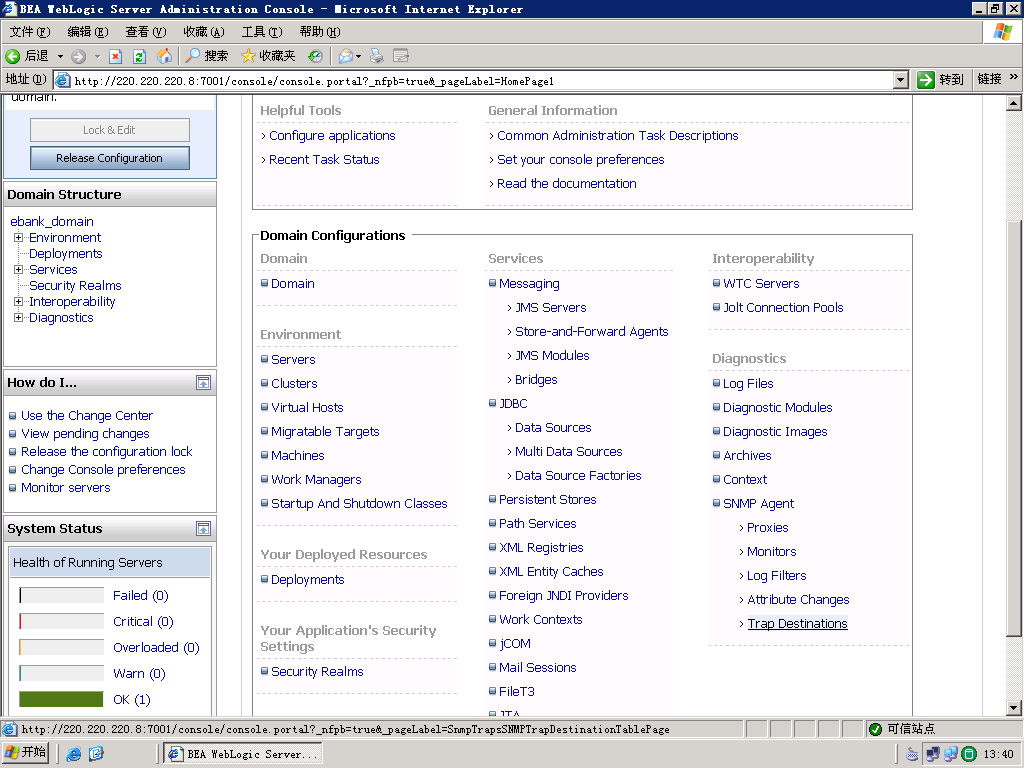


### WebLogic9配置过程

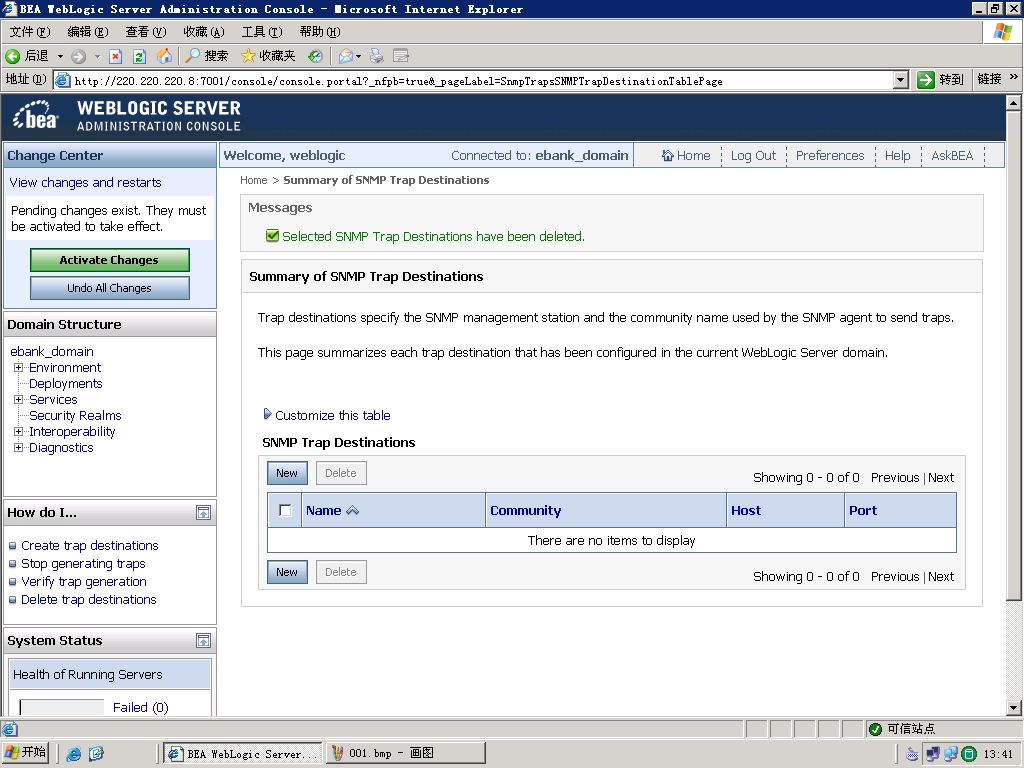
**如果服务器上安装有多个AdminServer，那么每个AdminServer都需要配置snmp，并且这台服务器上每个AdminServer在配置snmp port（SNMP 端口），snmp trap port（snmp陷阱端口），AgentX port（主要AgentX端口）需要不同。**

* + 1. 通过weblogic的console添加相关snmp配置。

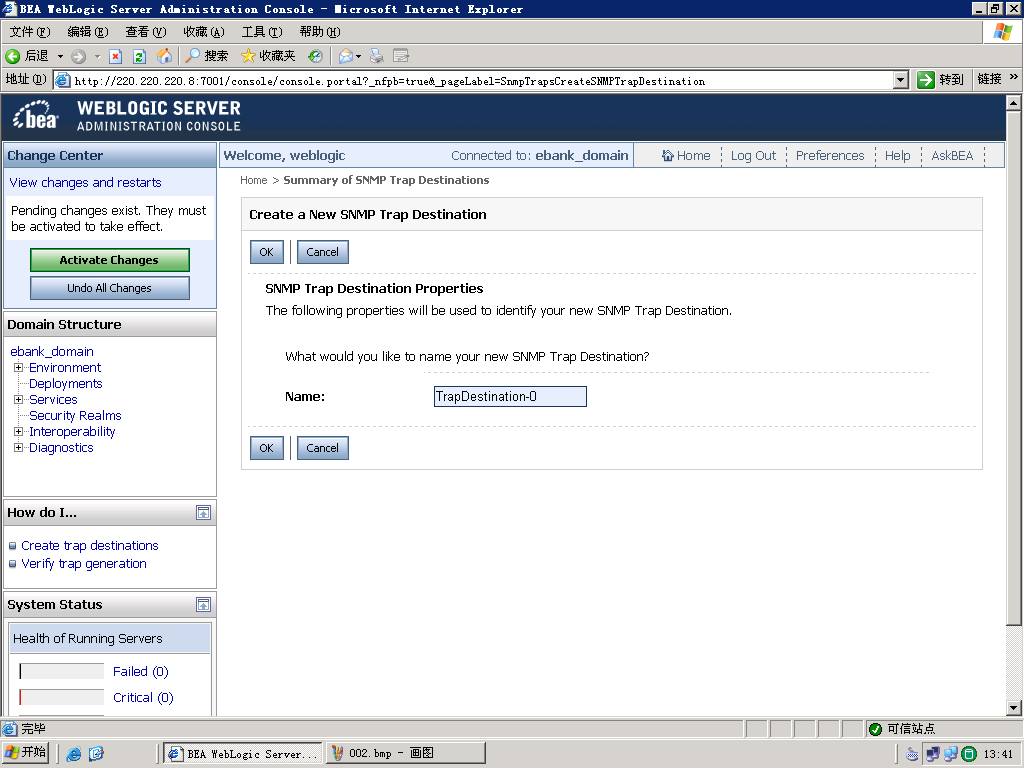
1. 通过<http://ip>：7001/console地址输入用户名和密码后登陆weblogic控制台



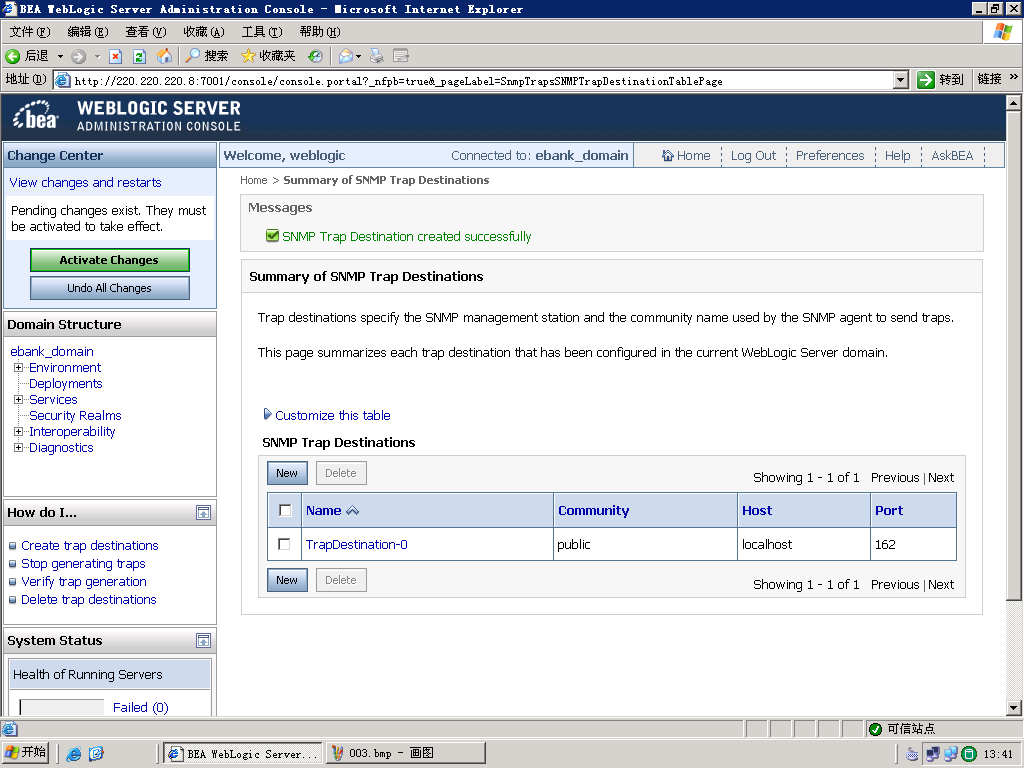
1. 点击Trap Destinations出现下面页面



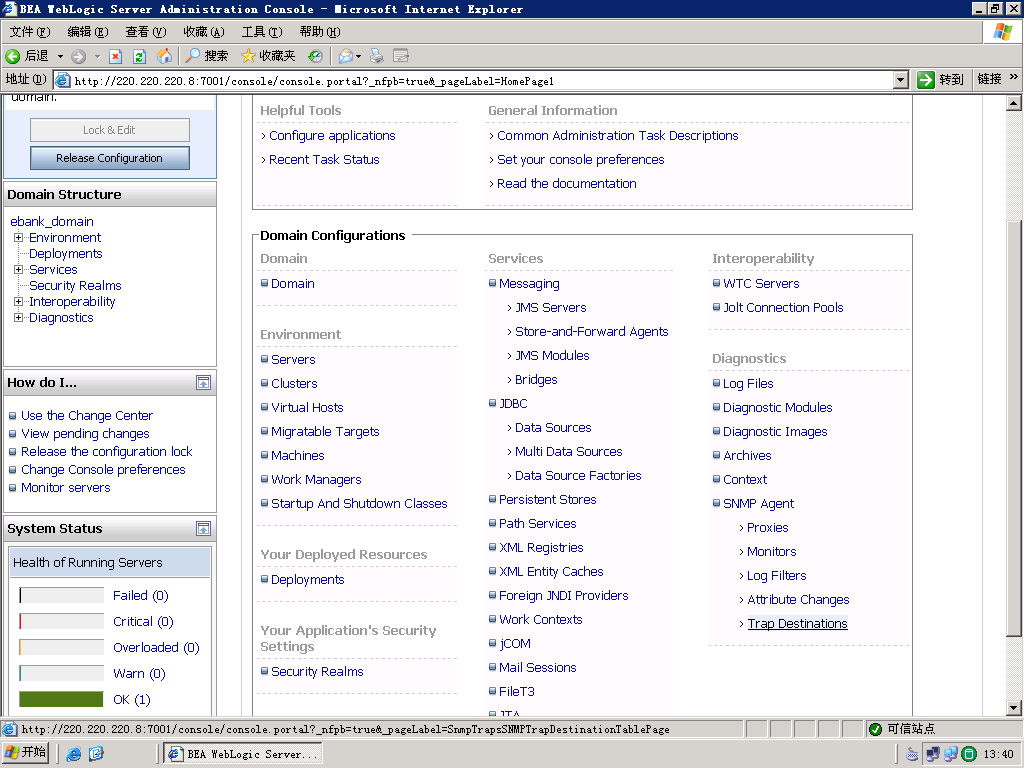
1. 通过点new按钮出现下面页面



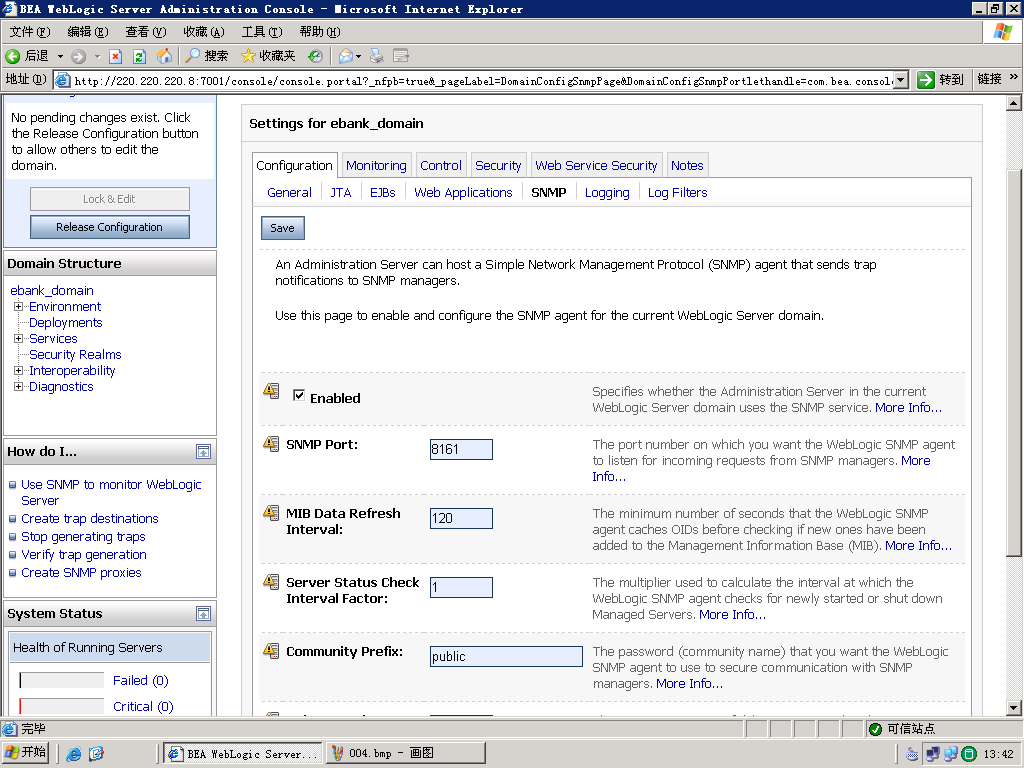
1. 通过ok按钮，进入下面页面



1. 添加成功，通过active change保存修改。回到主也面。



1. 点SNMP Agent进入下面页面，选中Enable，Sava保存后，退出。



这里的SNMP Port是可以修改的，但不要和其他端口冲突。

Weblogic的console端配置结束。

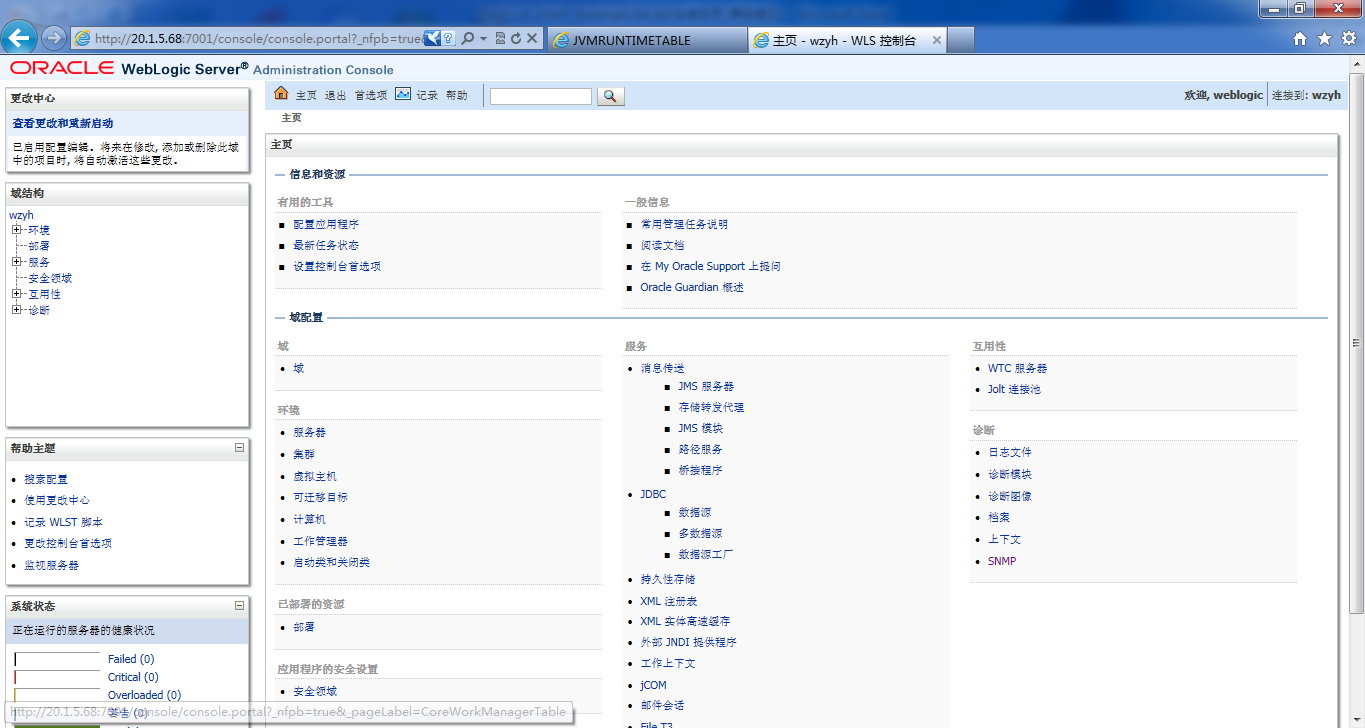
注意：在 weblogic9和之前的版本中，配置snmp服务后，只有在weblogic服务器从新启动后才可以生效。

### WebLogic10配置过程

**如果服务器上安装有多个AdminServer，那么每个AdminServer都需要配置snmp，并且这台服务器上每个AdminServer在配置snmp port（SNMP 端口），snmp trap port（snmp陷阱端口），AgentX port（主要AgentX端口）需要不同。**

1. 通过weblogic的console添加相关snmp配置。

通过<http://ip>:7001/console地址输入用户名和密码后登陆weblogic控制台



1. 点右下角的snmp选项。



1. 点要监控的服务器名字，本例为wzyh。按下图选择相应选项，保存退出。



这里的SNMP Port是可以修改的，但这里的设置要在TEP上用到，不要和其他端口冲突。

Weblogic的console端配置结束。

注意：在 weblogic10和之后的版本中，配置snmp服务后，不需要weblogic服务器从新启动就可以生效。

**配置SNMP 陷阱：**

选择“陷阱目的地”，下一步。



**点击“锁定并编辑”后，点击“新建”陷阱。**



**配置信息如下：**

**名称：**默认

**社区：**public

**主机：**localhost

**端口：**8162

**点击保存，并激活更改。**



**配置完成：**



# 部署

## 监控对象

修改属性文件conf.yaml

在projects下列出当前部署agent监控对象，不需要监控的可注释（#）掉

监控对象列表：

windows

heartbeat

oracle

sqlserver

tomcat

apache

weblogic

## 属性文件配置

修改属性文件conf.yaml

Kafka、zookeeper、heartbeat为必配项外，其他的配置根据监控对象而定，不需要的可删除或注释掉（#）。

具体配置参数可参加配置文件内注释说明。

# 安装

## 执行命令

java –jar ${jar-path}/core.jar ${conf-path}/conf.yaml