



## BellaDati 系统平台安装实施文档

## 目 录

一、	安装前准备.....	3
(一)	服务器建议配置.....	3
(二)	操作系统推荐.....	3
(三)	检查服务器 Java 版本.....	3
二、	安装并配置 GLASSFISH .....	4
(一)	解压缩 Glassfish .....	4
(二)	配置 Glashfish 管理员密码.....	4
(三)	登录到 Glassfish 管理控制台配置 .....	5
(四)	添加链接源.....	6
三、	安装数据库 POSTGRESQL .....	8
(一)	安装数据库 postgresql9.3 组合.....	8
(二)	配置数据库相关.....	8
(三)	创建数据用户和数据库.....	11
四、	配置并上传 BELLADATI .....	12
(一)	配置 belladati.war.....	12
(二)	登录 Glassfish 管理控制台上传 War 包 .....	13
(三)	完成部署, 登录测试.....	14

## 一、 安装前准备

### (一) 服务器建议配置

CPU	8core x 2 GHz 64-bit Intel Xeon or comparable
RAM	16 GB
Disk	100 GB, HDD
Network	100 Mbps

### (二) 操作系统推荐

推荐使用 RHEL 6.0+/Cent OS 6.0 +

### (三) 检查服务器 Java 版本

1、查看 java 版本（需 1.8 版本以上，1.6、1.7 的要卸载，从高到底卸载）

```
java -version
```

2、安装 java 版本 1.8

```
rpm -ivh jdk-8u25-linux-x64.rpm
```

3、修改环境变量的配置文件

```
vim /etc/profile
```

在最后加入：

```
export JAVA_HOME=/usr/java/jdk1.8.0_25
```

```
export PATH=$JAVA_HOME/bin:$PATH
```

```
export CLASSPATH=.:$JAVA_HOME/lib/dt.jar:$JAVA_HOME/lib/tools.jar
```

4、立即使环境变量生效

```
source /etc/profile
```

5、确认 java 版本为 1.8.0\_25

```
java -version
```

## 二、 安装并配置 Glassfish

### (一) 解压缩 Glassfish

程序为压缩包，只需要解压到指定目录即可，软件源：glassfish-4.1-web.zip

1、解压 glassfish

```
unzip glassfish-4.1-web.zip -d /opt
```

2、配置环境变量

```
vim /etc/profile
```

在最后加入

```
export GLASSFISH_HOME=/opt/glassfish4
```

```
export PATH=$GLASSFISH_HOME/bin:$PATH
```

3、立即使环境变量生效

```
source /etc/profile
```

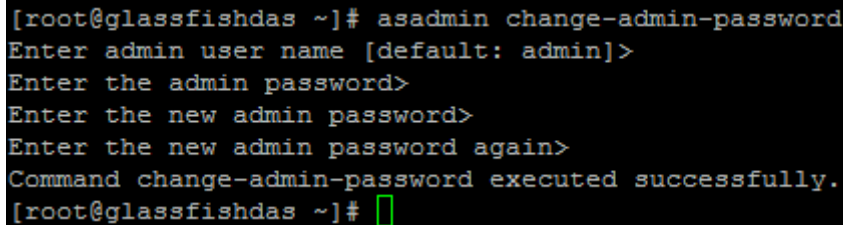
4、启动 glassfish 服务

```
asadmin start-domain
```

### (二) 配置 Glassfish 管理员密码

1、配置管理员密码(先两个回车后输入新密码 admin)如图 2.2.1 所示。

```
asadmin change-admin-password
```

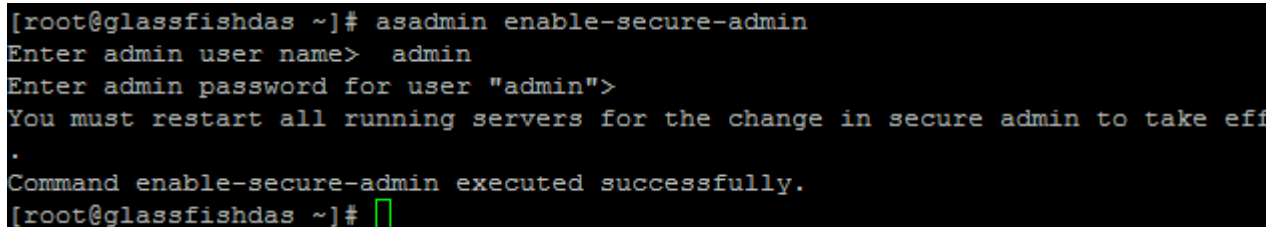


```
[root@glassfishdas ~]# asadmin change-admin-password
Enter admin user name [default: admin]>
Enter the admin password>
Enter the new admin password>
Enter the new admin password again>
Command change-admin-password executed successfully.
[root@glassfishdas ~]#
```

图 2.2.1

2、启用安全配置（默认情况下只有本机可以访问，启用安全之后其他主机也可以访问）如图：2.2.2 所示。

```
asadmin enable-secure-admin
```



```
[root@glassfishdas ~]# asadmin enable-secure-admin
Enter admin user name> admin
Enter admin password for user "admin">
You must restart all running servers for the change in secure admin to take eff
.
Command enable-secure-admin executed successfully.
[root@glassfishdas ~]#
```

图 2.2.2

3、重启 glassfish 服务，使以上配置生效。

```
asadmin restart-domain
```

### (三) 登录到 Glassfish 管理控制台配置

登录方式为 IP+端口 4848，例如：<http://10.X.X.X:4848>

#### 1、修改 JVM 配置

依次点击：Configurations → server-config → JVM Settings → JVM Options

-client 修改为 -server

-XX:MaxPermSize=192m 修改为 -XX:MaxPermSize=512m （内存的 1/8 到 1/4 大小）

-Xmx512m 修改为 -Xmx2048m （内存的 1/2 到 3/4 大小）

如图 2.3.1 所示

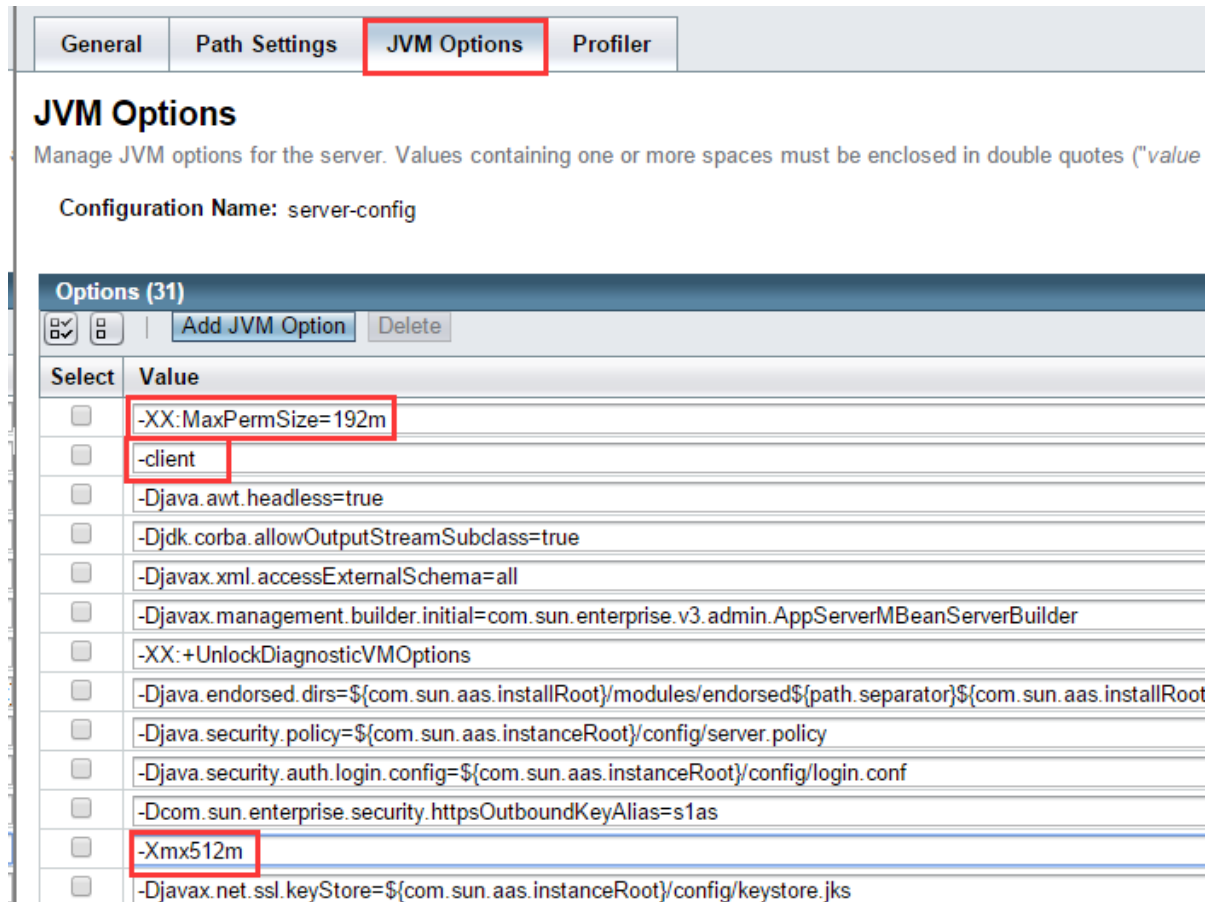


图 2.3.1

2、修改好后，点击保，按提示重启服务。如图 2.3.2 所示

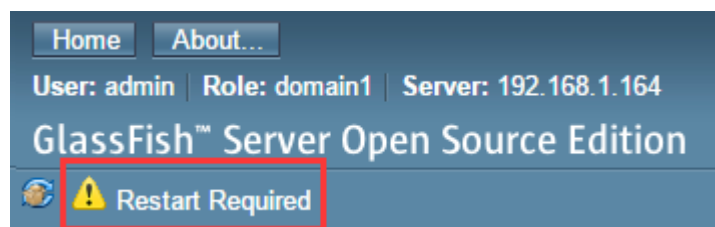


图 2.3.2

#### (四) 添加链接源

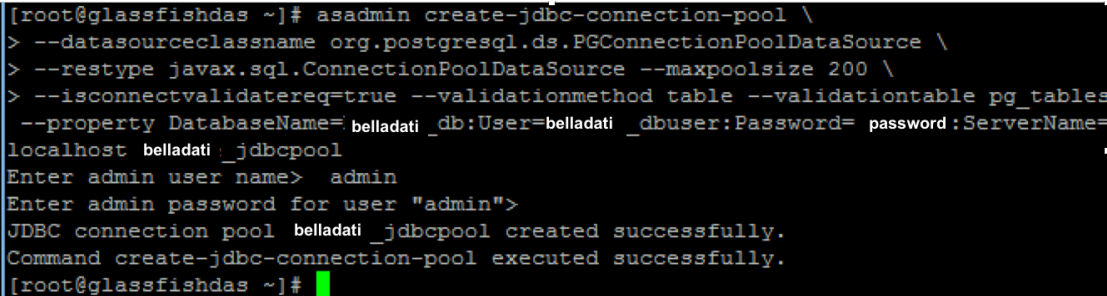
1、复制 JDBC 连接文件到指定目录 (源 postgresql-9.3-1103.jdbc4.jar)  
cp postgresql-9.3-1103.jdbc4.jar /opt/glassfish4/glassfish/lib/

2、配置 JDBC 连接池

注意:

1. 输入这个“\”符号后表示要回车换行;
2. 数据库连接账号要按实际情况输入, 如图 2.4.2 所示。

```
asadmin create-jdbc-connection-pool \  
--datasourceclassname org.postgresql.ds.PGConnectionPoolDataSource \  
--restype javax.sql.ConnectionPoolDataSource --maxpoolsize 200 \  
--isconnectvalidatereq=true --validationmethod table --validationtable  
pg_tables --property  
DatabaseName=belladati_db:User=belladati_dbuser:Password=password:ServerName=db  
_host belladati_jdbcpool
```

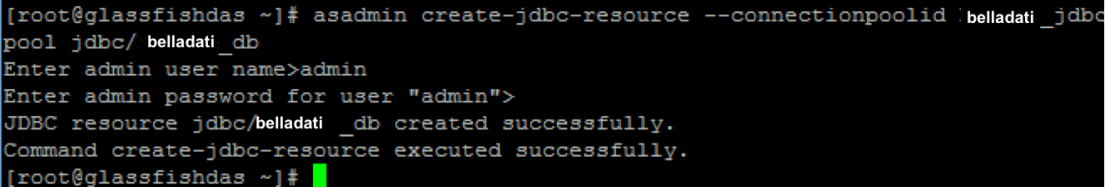


```
[root@glassfishdas ~]# asadmin create-jdbc-connection-pool \  
> --datasourceclassname org.postgresql.ds.PGConnectionPoolDataSource \  
> --restype javax.sql.ConnectionPoolDataSource --maxpoolsize 200 \  
> --isconnectvalidatereq=true --validationmethod table --validationtable pg_tables  
--property DatabaseName=belladati_db:User=belladati_dbuser:Password= password:ServerName=  
localhost belladati_jdbcpool  
Enter admin user name> admin  
Enter admin password for user "admin">  
JDBC connection pool belladati_jdbcpool created successfully.  
Command create-jdbc-connection-pool executed successfully.  
[root@glassfishdas ~]#
```

图 2.4.2

3、配置 JDBC 资源, 如图 2.4.3 所示。

```
asadmin create-jdbc-resource --connectionpoolid belladati_jdbcpool  
jdbc/belladati_db
```



```
[root@glassfishdas ~]# asadmin create-jdbc-resource --connectionpoolid belladati_jdbc  
pool jdbc/ belladati_db  
Enter admin user name>admin  
Enter admin password for user "admin">  
JDBC resource jdbc/belladati_db created successfully.  
Command create-jdbc-resource executed successfully.  
[root@glassfishdas ~]#
```

图 2.4.3

4、登录 UI 控制台查看配置结果。如图 2.4.4 所示

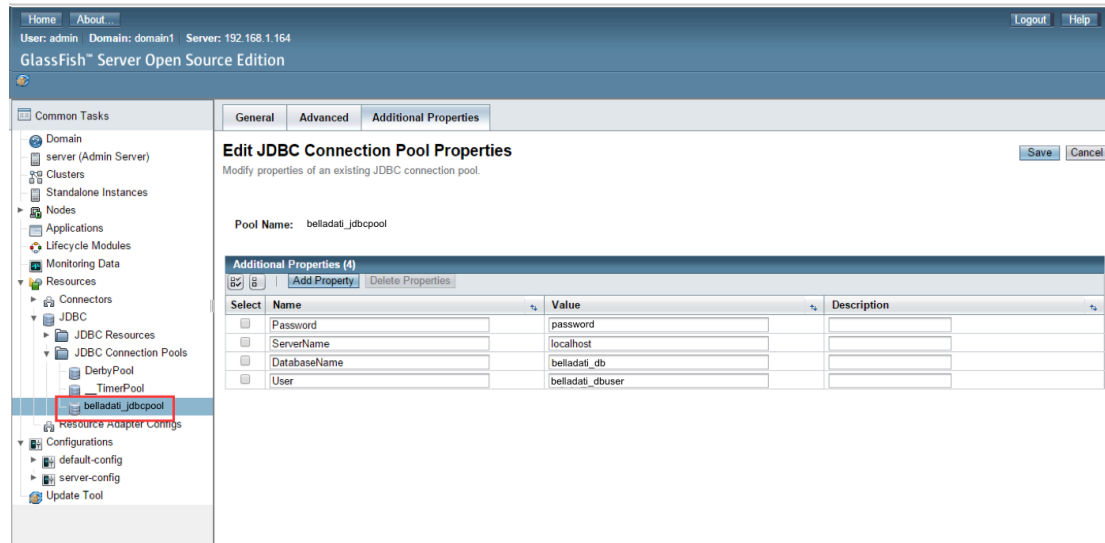


图 2.4.4

5、修改 HTTP Service 配置，点击 http listeners 下面的 http-listener-1, 将侦听端口从 8080 改为 80 端口，如图 2.4.5 所示。

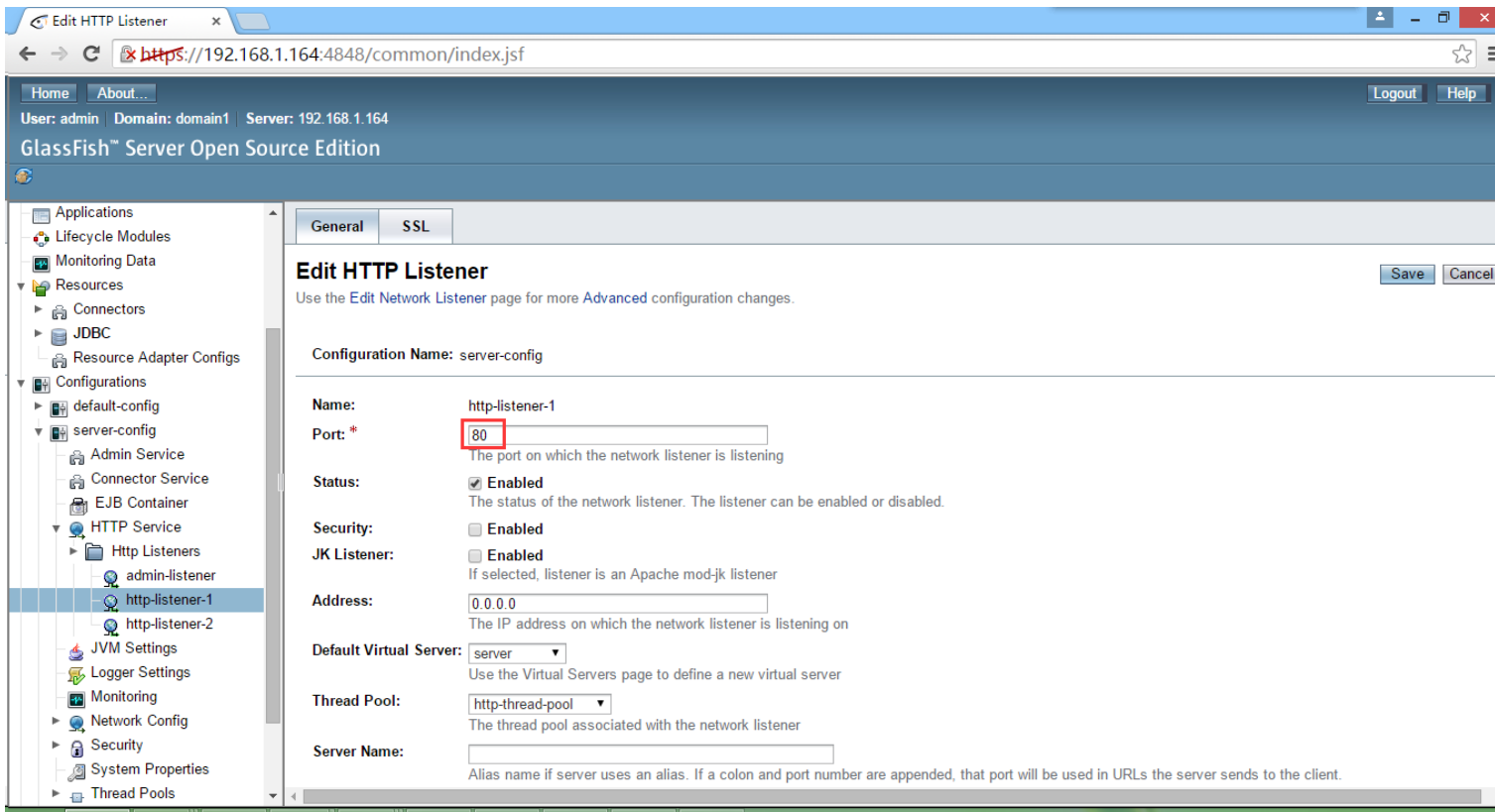


图 2.4.5

### 三、 安装数据库 Postgresql

#### (一) 安装数据库 postgresql9.3 组合

1、 有 4 个安装包，安装顺序 A-B-C-D 依次安装，如图 3.1.1 所示。

- A. postgresql93-libs-9.3.4-1PGDG.rhel6.x86\_64.rpm
- B. postgresql93-9.3.4-1PGDG.rhel6.x86\_64.rpm
- C. postgresql93-server-9.3.4-1PGDG.rhel6.x86\_64.rpm
- D. postgresql93-contrib-9.3.4-1PGDG.rhel6.x86\_64.rpm

注意：

安装出现警告而无法安装，是应为需要先安装相应的依赖包去支持，按提示先安装依赖包即可。

```
[root@glassfishnode1 RPM]# rpm -ivh postgresql93-libs-9.3.4-1PGDG.rhel6.x86_64.rpm
warning: postgresql93-libs-9.3.4-1PGDG.rhel6.x86_64.rpm: Header V4 DSA/SHA1
Signature, key ID 442df0f8: NOKEY
Preparing...                               (10)
##### [100%]
 1:postgresql93-libs                       ( 1
##### [100%]

[root@glassfishnode1 RPM]# rpm -ivh postgresql93-contrib-9.3.4-1PGDG.rhel6.x86_64.rpm
warning: postgresql93-contrib-9.3.4-1PGDG.rhel6.x86_64.rpm: Header V4 DSA/SHA1
Signature, key ID 442df0f8: NOKEY
error: Failed dependencies:
 libssp-uuid.so.16()(64bit) is needed by postgresql93-contrib-9.3.4-1PGDG.rhel6.x86_64
[root@glassfishnode1 ~]# rpm -ivh uuid-1.6.1-10.el6.x86_64.rpm
warning: uuid-1.6.1-10.el6.x86_64.rpm: Header V3 RSA/SHA256 Signature, key ID fd431d51: NOKEY
Preparing...                               (10)
##### [100%]
 1:uuid                                     ( 1
##### [100%]
```

图 3.1.1

#### (二) 配置数据库相关

1、 修改数据库数据的存放路径，修改后如图 3.2.1 所示。

vim /etc/rc.d/init.d/postgresql-9.3

```
# Set defaults for configuration variables
PGENGINE=/usr/pgsql-9.3/bin
PGPORT=5432
PGDATA=/opt/pgsql/9.3/data
PGLOG=/opt/pgsql/9.3/pgstartup.log
# Log file for pg_upgrade
PGUPLOG=/opt/pgsql/$PGMAJORVERSION/pgupgrade.log
```

图 3.2.1



2、如路径不存在，需手工创建目录

```
mkdir -p /opt/pgsql/9.3/data
```

3、修改目录的拥有人

```
chown postgres:postgres -R /opt/pgsql/
```

4、完成后，初始化数据库，如图 3.2.4 所示。

```
service postgresql-9.3 initdb
```

```
[root@glassfishdas postgresql]# service postgresql-9.3 initdb
Initializing database: [ OK ]
[root@glassfishdas postgresql]#
```

图 3.2.4

5、修改客户端访问控制文件，如图 3.2.5 所示。

```
vim /opt/pgsql/9.3/data/pg_hba.conf
```

数据库安装在本机使用 trust，安装在其他机器使用 md5

```
# TYPE      DATABASE      USER      ADDRESS      METHOD
# "local" is for Unix domain socket connections only
local      all           all              trust
# IPv4 local connections:
host      all           all          127.0.0.1/32  trust
host      all           all          192.168.1.164/32 md5
# IPv6 local connections:
host      all           all          ::1/128      ident
# Allow replication connections from localhost, by a user with the
# replication privilege.
```

图 3.2.5

6、修改数据库的主配置文件，按配置文档修改，如图 3.2.6 所示。

```
vim /opt/pgsql/9.3/data/postgresql.conf
```

将侦听主机 localhost 更改为所有主机 \*

```
listen_addresses = '*'          # what IP address(es) to listen on;
                                # comma-separated list of addresses;
                                # defaults to 'localhost'; use '*' for
                                # (change requires restart)
port = 5432                     # (change requires restart)
max_connections = 100           # (change requires restart)
```

图 3.2.6

按照对应的配置文件（如 dbconfiguration8GB RAM）修改对应的参数

```
shared_buffers = 4096MB
temp_buffers = 4MB
max_connections = 250
work_mem = 8MB
maintenance_work_mem = 16MB
max_stack_depth = 4MB

fsync = on

wal_buffers = 1024kB
effective_cache_size = 6144MB

#-----
# WRITE AHEAD LOG
#-----

checkpoint_segments = 128
checkpoint_timeout = 15min
checkpoint_completion_target = 0.8

#-----
# RUNTIME STATISTICS
#-----
# - Query/Index Statistics Collector -

track_activities = on
track_counts = on

#-----
# AUTOVACUUM PARAMETERS
#-----

autovacuum = on

log_autovacuum_min_duration = 0

autovacuum_max_workers = 3

autovacuum_naptime = 15s
autovacuum_vacuum_threshold = 25

autovacuum_analyze_threshold = 25

autovacuum_vacuum_scale_factor = 0.01
autovacuum_analyze_scale_factor = 0.05

autovacuum_vacuum_cost_delay = 10ms

autovacuum_vacuum_cost_limit = 1000
```

7、启动数据库服务 postgresql-9.3，如图 3.2.7 所示。

```
service postgresql-9.3 start
```

```
[root@glassfishdas pgsq]# service postgresql-9.3 start
Starting postgresql-9.3 service: [ OK ]
[root@glassfishdas pgsq]#
[root@glassfishdas pgsq]#
```

图 3.2.7

### (三) 创建数据用户和数据库

这里创建的用户，密码是连接数据库池时的用户名和密码。

1、切换到 postgres 用户下

```
su - postgres
```

2、创建用户

```
createuser -SDRP belladati_dbuser
```

3、创建数据库

```
createdb -E UNICODE -O belladati_dbuser belladati_db -T template0
```

如图 3.3.3 所示。

```
[root@glassfishdas pgsq]# su - postgres
-bash-4.1$ createuser -SDRP belladati_dbuser
Enter password for new role:
Enter it again:
-bash-4.1$ createdb -E UNICODE -O belladati_dbuser belladati_db -T template0
-bash-4.1$
```

图 3.3.3

## 四、配置并上传 Belladati

### (一) 配置 belladati.war

1、从压缩文件中抽取 war 文件，如图 4.1.1 所示。

```
unzip BellaDati-2_7_13_4.war.zip belladati.war
```

```
[root@glassfishdas ~]#  
[root@glassfishdas ~]# unzip BellaDati-2_7_13_4.war.zip belladati.war  
Archive:  BellaDati-2_7_13_4.war.zip  
  inflating: belladati.war  
[root@glassfishdas ~]#
```

图 4.1.1

2、抽取 war 包的配置文件，如图 4.1.2 所示。

```
unzip belladati.war WEB-INF/classes/conf/application.properties
```

```
[root@glassfishdas ~]# unzip belladati.war WEB-INF/classes/conf/application.properties  
Archive:  belladati.war  
  inflating: WEB-INF/classes/conf/application.properties  
[root@glassfishdas ~]#
```

图 4.1.2

3、修改 war 包配置文件，如图 4.1.3 所示。

```
vi WEB-INF/classes/conf/application.properties
```

localhost 更改为 本机 ip

jdbc 资源 更改为 belladati\_db

下面的都加上 # 成为注释

```
# APPLICATION properties^M  
#application.homeDir=^M  
application.url=http://192.168.1.164^M  
application.production-mode=true^M  
application.secure-mode=false^M  
^M  
# JDBC properties^M  
jdbc.resource=jdbc/belladati_db^M  
#when running on Tomcat:^^M  
#jdbc.resource=java:/comp/env/jdbc/belladati_db^M  
#when connecting directly to the db^M  
#jdbc.driverClassName=org.postgresql.Driver^M  
#jdbc.url=jdbc:postgresql://localhost:5432/belladati_db^M  
#jdbc.username=belladati^M  
#jdbc.password=belladati^M  
#jdbc.pool.maxActive^M  
#jdbc.pool.minIdle^M  
#jdbc.pool.maxIdle^M
```

图 4.1.3

4、删除 war 中原有的配置文件，如图 4.1.4 所示。

```
zip -d belladati.war WEB-INF/classes/conf/application.properties
```

```
[root@glassfishdas ~]# zip -d belladati.war WEB-INF/classes/conf/application.properties
deleting: WEB-INF/classes/conf/application.properties
```

图 4.1.4

5、上传修改过后的配置文件到 belladati.war 包里面，如图 4.1.5 所示。

```
zip -u belladati.war WEB-INF/classes/conf/application.properties
```

```
[root@glassfishdas ~]# zip -u belladati.war WEB-INF/classes/conf/application.properties
adding: WEB-INF/classes/conf/application.properties (deflated 55%)
[root@glassfishdas ~]#
```

图 4.1.5

## (二) 登录 Glassfish 管理控制台上传 War 包

1、登录 Glassfish 管理控制台，依次点击 Applications → Deploy ，如图 4.2.1 所示。

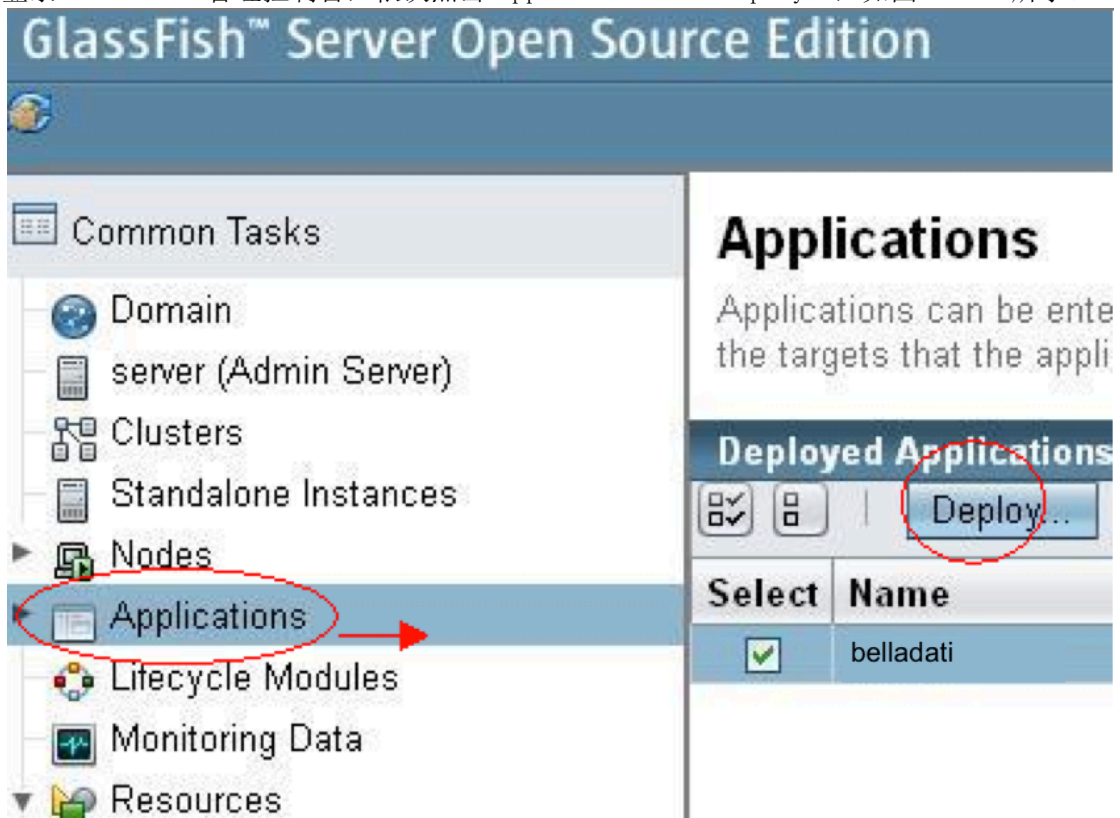
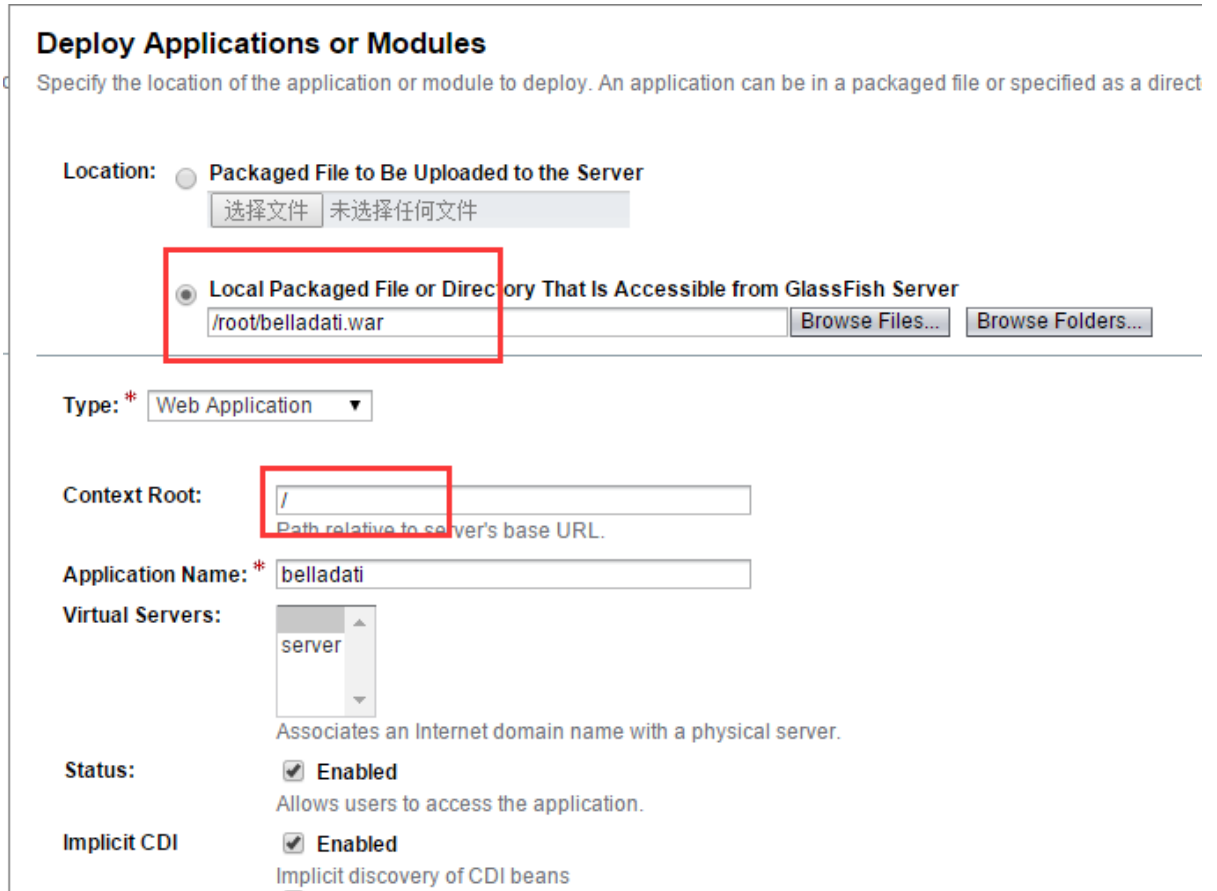


图 4.2.1

2、选择刚才修改过后的 war 包，并将 Context Root 参数修改为 / 符号，如图 4.2.2 所示。



**Deploy Applications or Modules**

Specify the location of the application or module to deploy. An application can be in a packaged file or specified as a direct

**Location:**  Packaged File to Be Uploaded to the Server  
选择文件 未选择任何文件

Local Packaged File or Directory That Is Accessible from GlassFish Server  
/root/belladati.war Browse Files... Browse Folders...

**Type:** \* Web Application

**Context Root:** /  
Path relative to server's base URL.

**Application Name:** \* belladati

**Virtual Servers:** server  
Associates an Internet domain name with a physical server.

**Status:**  Enabled  
Allows users to access the application.

**Implicit CDI:**  Enabled  
Implicit discovery of CDI beans

图 4.2.2

### (三) 完成部署，登录测试

1、完成以上安装和部署后，用 IP 登录测试，不带端口号，如图 4.3.1 所示表示安装配置成功。

格式：http://10. X. X. X



图 4.3.1

2、Linux 系统的防火墙可能会阻止客户端无法正常访问 Web 界面，可以修改防火墙设置，开通这几个端口 4848、80、8181、5432，如图 4.3.2 所示。

修改防火墙的配置文件：vim /etc/sysconfig/iptables

修改后需要重启防火墙：service iptables restart

```
# Firewall configuration written by system-config-firewall
# Manual customization of this file is not recommended.
*filter
:INPUT ACCEPT [0:0]
:FORWARD ACCEPT [0:0]
:OUTPUT ACCEPT [0:0]
-A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
-A INPUT -p icmp -j ACCEPT
-A INPUT -i lo -j ACCEPT
-A INPUT -m state --state NEW -m tcp -p tcp --dport 22 -j ACCEPT
-A INPUT -m state --state NEW -m tcp -p tcp --dport 4848 -j ACCEPT
-A INPUT -m state --state NEW -m tcp -p tcp --dport 80 -j ACCEPT
-A INPUT -m state --state NEW -m tcp -p tcp --dport 8181 -j ACCEPT
-A INPUT -m state --state NEW -m tcp -p tcp --dport 5432 -j ACCEPT
-A INPUT -j REJECT --reject-with icmp-host-prohibited
-A FORWARD -j REJECT --reject-with icmp-host-prohibited
COMMIT
```

图 4.3.2