《部署指南》

部署指南

Maven部署api-provider

1.检查rpc.properties中的[rpc.registry.address]地址（zookeeper服务器地址）是否正确；
2.检查api-provider项目中pom.xml文件的build-->mainClass节点是否配置正确；
3.运行（双击）api-provider项目Maven的clean命令（可选）；
4.运行（双击）api-provider项目Maven的install命令；
若api-provider依赖的模块有改动，则需要先生成依赖模块的jar包，最简单的办法就是运行kitchen(root)的install命令
5.将target中生成的【api-provider***.jar】包和【lib文件夹】上传至服务器中的某个目录（jar包和lib文件夹同级目录）；
6.运行jar文件；
6.1. Windows系统：进入jar包所存放的目录，执行命令"java -jar api-provider.jar"；
6.2. Linux系统： 进入jar包所存放的目录，执行命令"java -jar api-provider***.jar"；

Tomcat部署

版本 Tomcat 9+
1、修改Tomcat配置： conf/server.xml中增加 URIEncoding="UTF-8" 支持get方式的中文编解码
2、注意防止appBase和docBase同时配置时，启动Tomcat后，加载两次war包的问题。appBase不要指定到webapps目录下，相应的docBase使用绝对路径。例如：

```
<Host name="localhost" appBase="D:/environment/server/apache-tomcat/apache-tomcat-9-local" unpackWARs="true" autoDeploy="true">
<Context path="" docBase="D:/environment/server/apache-tomcat/apache-tomcat-9-local/webapps/api-portal-1.0.0-SNAPSHOT" deug="0" reloadable="true"/>
<!-- SingleSignOn valve, share authentication between web applications Documentation at: /docs/config/valve.html -->
```

```
<!--
<Valve className="org.apache.catalina.authenticator.SingleSignOn" />
-->

<!-- Access log processes all example.
Documentation at: /docs/config/valve.html
Note: The pattern used is equivalent to using pattern="common" -->
<Valve className="org.apache.catalina.valves.AccessLogValve" directory="logs"
prefix="localhost_access_log" suffix=".txt"
pattern="%h %l %u %t &amp;quot;%r&amp;quot; %s %b" />

</Host>
```

Redis部署

关于安全配置：

1、账号与认证（需要重启Redis服务）：

修改Redis配置文件中的requirepass为所需密码

修改kitchen-cache模块中cache.properties的kitchen.cache.redis.password=为所需密码

2、指定Redis服务使用的网卡（需要重启Redis服务）：

修改Redis配置文件中的bind，例如：bind=127.0.0.1 [and 其它本机ip]

3、设置访问源IP白名单，即防火墙策略，仅允许指定的IP来访问Redis服务：

Linux：iptables -A INPUT -s xxx.xxx.xxx.xxx -p tcp --dport 6379 -j ACCEPT

4、服务运行的权限最小化（需要重启Redis服务）：

修改Linux运行Redis服务的账号，以较低权限账号运行Redis服务，并禁用该账号的登录权限。

示例：创建一个无home目录和无法登录的普通权限账号

“useradd -M -s /sbin/nologin [username]”

5、Redis服务精细化授权（需要重启Redis服务）：

redis没有权限分离之说，无管理员账号和普通账户之分，导致攻击者登陆后可执行任意操作，因此需要隐藏重要命令，具体如下：

FLUSHDB, FLUSHALL, KEYS,PEXPIRE, DEL, CONFIG, SHUTDOWN, BGREWRITEAOF, BGSAVE, SAVE, SPOP, SREM, RENAME,DEBUG, EVAL等。

在redis2.8.1 及 redis3.x(<3.0.2) 版本下存在eval沙箱逃逸漏洞，攻击者可通过该漏洞执行任意Lua代码。

具体缓解攻击操作，供参考：下述配置将config/flushdb/flushall设置为空，即禁用该命令；也可命名

一些攻击者难以猜解的名字。

rename-command CONFIG ""

rename-command flushall ""

rename-command flushdb ""

rename-command shutdown shotdown_test

保存后，执行/etc/init.d/redis-server restart重启生效。

6、安全补丁

不定期关注最新软件版本，并升级redis到最新版，防止新漏洞被利用。

部署完Redis记得使用远程的客户端连接一下试试