**问题：**

Weblogic/Tomcat 通过JAVA直接远程调用或者调用本地Shell还是通过Ansible来部署新程序 + 启动停止服务 如何管理?需要哪些配置信息?

**采用方案：**

通过Ansible来部署新程序+启动停止服务；

**具体步骤：**

1. 测试准备工作：

两台Linux服务器：

主服务器（本系统需要部署该服务器上），需要的软件环境：

Python 、ansible、jdk

从服务器(灰度测试服务器)，需要的软件环境：

Python 、Tomcat、jdk

1. 配置信息：

主服务器ansible目录下的hosts文件和example.yaml文件。

1)hosts文件里面配置从服务器的ip、端口、用户名、密码等信息，例如：

129w ansible\_ssh\_host=192.168.10.129 ansible\_ssh\_port=22 ansible\_ssh\_user=root ansible\_ssh\_pass="123456" #host\_to\_replace

2)example.yaml文件是一个playbook（剧本），该文件里定义了Tomcat的整个部署流程，例如：

---

- hosts: all

vars: #下面的3个变量需要作为参数传过来

local\_war\_path: /opt/store/ #local\_war\_path 本地war包的路径

war\_file: gr\* gr2\* #war\_file 要删除的旧war包

tomcat: /usr/local/tomcat8/ #remote\_tomcat\_path 从服务器上Tomcat的路径

tasks:

- name: stop tomcat service #调用脚本shutdown\_tomcat.sh，停止Tomcat服务

script: "/etc/ansible/shutdown\_tomcat.sh"

- name: remove old war file #删除旧的war包及文件夹

shell: cd {{ tomcat }}webapps&&rm -rf {{ war\_file }}

- name: copy new war file #从主服务器拷贝新war包

copy: src={{ local\_war\_path }} dest={{ tomcat }}webapps/

- name: start tomcat service #启动Tomcat

shell: chdir={{ tomcat }}bin/ nohup ./startup.sh &

3.java测试代码：

/\*\*

\* 替换配置文件中的参数

\* **@param** request

\*/

@RequestMapping(value = "/file/change", method = RequestMethod.*GET*)

**public** **void** fileChange(HttpServletRequest request) {

String space4Str = " ";

FileHelper.*Modify*("/etc/ansible/example.yaml", "#remote\_tomcat\_path", space4Str + "tomcat: /usr/local/tomcat8/ #remote\_tomcat\_path");//替换Tomcat路径

FileHelper.*Modify*("/etc/ansible/example.yaml", "#local\_war\_path", space4Str + "war\_file: /opt/store/ #local\_war\_path");//war包路径

FileHelper.*Modify*("/etc/ansible/hosts", "#host\_to\_replace",

"129web ansible\_ssh\_host=192.168.10.129 ansible\_ssh\_pass=\"123qwe\" #host\_to\_replace");//从服务器信息

}

/\*\*

\* 执行Linux命令

\* **@param** request

\*/

@RequestMapping(value = "/file/change2", method = RequestMethod.*GET*)

**public** **void** fileChange2(HttpServletRequest request) {

String playbook= ExecLinuxCMD.*exec*("ansible-playbook /etc/ansible/example.yaml").toString();// 执行playbook脚本

System.*out*.println("==========获得值=============");

System.*out*.println(playbook);// 打印执行结果

}

1. 测试结果：

可实现预期效果：自动化部署，启动停止Tomcat服务。