发布环境：代码发布机，有些公司为堡垒机（安全屏障）

使用者：运维

管理者： 运维（有经验）

发布机：往往需要有2台（主备）

生产环境

使用者： 运维

功能： 对用户提供公司产品的服务

管理者： 只能是运维

生产环境中服务器数量： 一般比较多，而且应用非常重要，往往需要自动化工具协助部署配置应用

灰度环境（生产环境的一部分）

使用者： 运维

功能：在全量发布代码的功能面向少量精准用户的环境，珂基于主机或者用户执行灰度发布

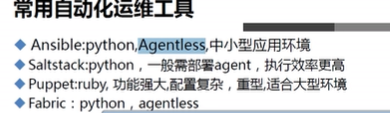
如果没问题，在全量发布至所以服务器上

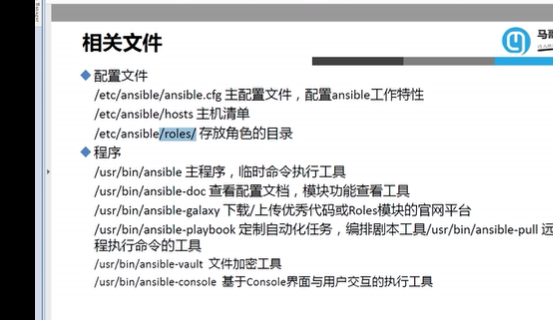
预发布验证：在测试机上测试

程序发布：

不能导致系统故障或者造成系统完全不可用

不能影响用户体验





## 一：Ansible概述

运维工具按需不需要有代理程序来划分的话分两类：

agent（需要有代理工具）：基于专用的agent程序完成管理功能，puppet, func, zabbix

agentless（无须代理工具）：基于ssh服务完成管理，ansible, fabric

## 二：简介

　　Ansible是一个简单的自动化运维管理工具，基于Python语言实现，由Paramiko和PyYAML两个关键模块构建，可用于自动化部署应用、配置、编排task(持续交付、无宕机更新等)。

　　Ansible与Saltstack最大的区别是Ansible无需在被控主机部署任何客户端代理，默认直接通过SSH通道进行远程命令执行或下发配置：相同点是都具备功能强大、灵活的系统管理、状态配置，两者都提供丰富的模板及API，对云计算平台、大数据都有很好的支持。

1：Ansible的特点：

（1）部署简单，只需在主控端部署Ansible环境，被控端无需做任何操作；

（2）默认使用SSH协议对设备进行管理；

（3）主从集中化管理；

（4）配置简单、功能强大、扩展性强；

（5）支持API及自定义模块，可通过Python轻松扩展；

（6）通过Playbooks来定制强大的配置、状态管理；

（7）对云计算平台、大数据都有很好的支持；

（8）提供一个功能强大、操作性强的Web管理界面和REST API接口——AWX平台。

（9）幂等性：一种操作重复多次结果相同

简评：

（1）轻量级，无需在客户端安装agent，更新时，只需在操作机上进行一次更新即可；  
（2）批量任务执行可以写成脚本，而且不用分发到远程就可以执行；  
（3）使用python编写，维护更简单，ruby语法过于复杂；  
（4）支持sudo。

2：Ansible架构

Ansible核心组件说明：

（1）Ansible：Ansible的核心程序

（2）Host Lnventory：记录了每一个由Ansible管理的主机信息，信息包括ssh端口，root帐号密码，ip地址等等。可以通过file来加载，可以通过CMDB加载

（3）Playbooks：YAML格式文件，多个任务定义在一个文件中，使用时可以统一调用，“剧本”用来定义那些主机需要调用那些模块来完成的功能.

（4）Core Modules：Ansible执行任何管理任务都不是由Ansible自己完成，而是由核心模块完成；Ansible管理主机之前，先调用core Modules中的模块，然后指明管理Host Lnventory中的主机，就可以完成管理主机。

（5）Custom Modules：自定义模块，完成Ansible核心模块无法完成的功能，此模块支持任何语言编写。

（6）Connection Plugins：连接插件，Ansible和Host通信使用

