Tedu.cn
达内教育

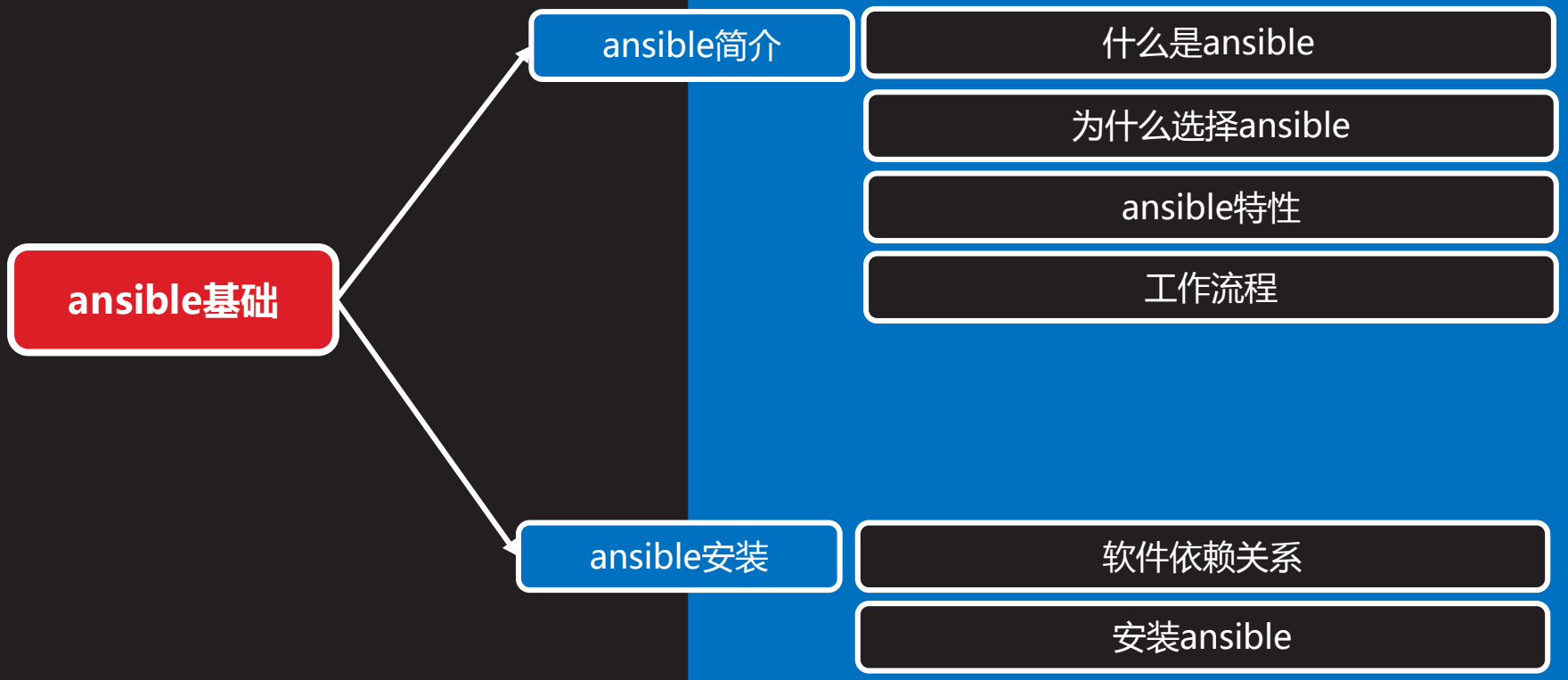
大型架构及配置技术

NSD ARCHITECTURE DAY01

内容

上午	09:00 ~ 09:30	ansible基础
	09:30 ~ 10:20	
	10:30 ~ 11:20	
	11:30 ~ 12:00	ad-hoc
下午	14:00 ~ 14:50	
	15:00 ~ 15:50	批量配置管理
	16:10 ~ 17:10	
	17:20 ~ 18:00	总结和答疑

ansible基础


ansible基础
ansible简介
什么是ansible
为什么选择ansible
ansible特性
工作流程
ansible安装
软件依赖关系
安装ansible

ansible简介

什么是ansible

- ansible是2013年推出的一款IT自动化和DevOps软件，2015年被RedHat收购。是基于Python研发，糅合很多老运维工具的优点，实现了批量操作系统配置，批量程序部署，批量运行命令等功能
- ansible可以实现：
  - 自动化部署APP
  - 自动化管理配置项
  - 自动化持续交付
  - 自动化（AWS）云服务管理

知识讲解

为什么选择ansible

- 选择一款配置管理软件，无外乎从以下几点来权衡利弊
  - 活跃度（社区）
  - 学习成本
  - 使用成本
  - 编码语言
  - 性能
  - 使用是否广泛

知识讲解

为什么选择ansible（续2）

- ansible优点
  - 只需要SSH和Python即可使用
  - 无客户端
  - ansible功能强大，模块丰富
  - 上手容易，门槛低
  - 基于Python开发，做二次开发更容易
  - 使用公司比较多，社区活跃

知识讲解

ansible特性

- 模块化设计，调用特定的模块完成特定任务
- 基于Python语言实现
  - paramiko
  - PyYAML (半结构化语言)
  - Jinja2
- 其模块支持JSON等标准输出格式，可以采用任何编程语言重写

知识讲解

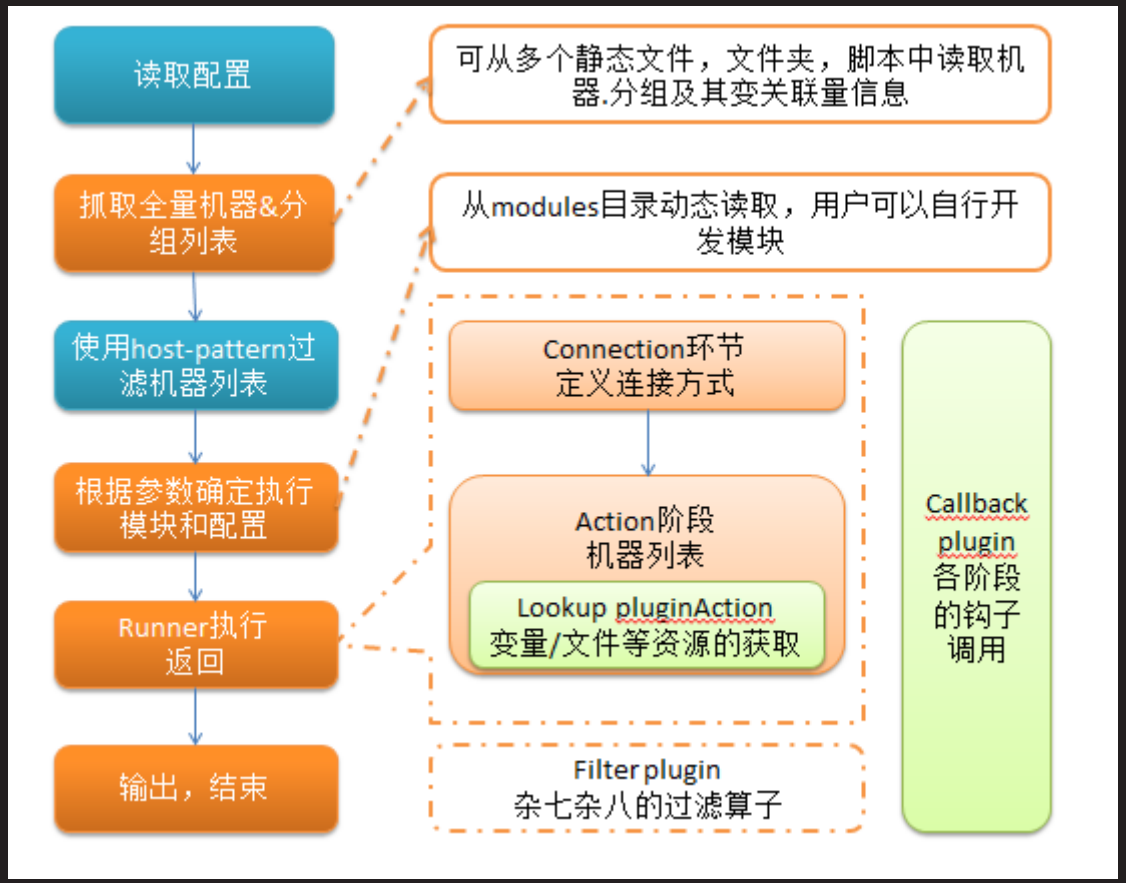
ansible特性（续1）

- 部署简单
- 主从模式工作
- 支持自定义模块
- 支持playbook
- 易于使用
- 支持多层部署
- 支持异构IT环境

知识讲解

工作流程

- ansible大体执行过程



ansible安装

软件依赖关系

- 对管理主机
  - 要求Python 2.6 或Python 2.7
- ansible 使用以下模块，都需要安装
  - paramiko
  - PyYAML
  - Jinja2
  - httplib2
  - six

知识讲解

软件依赖关系（续1）

- 对于被托管主机
  - ansible默认通过SSH协议管理机器
  - 被管理主机要开启ssh服务，允许ansible主机登录
  - 在托管节点上也需要安装Python2.5或以上的版本
  - 如果托管节点上开启了SElinux，需要安装libselinux-python

知识讲解

安装ansible（续2）

知识讲解

- Yum安装
  - 把软件包拷贝到（真机） ftp 共享目录中
  - 更新索引文件

    createrepo --update .
- 安装及验证
  - 在 ansible 托管主机上配置 yum 配置文件
  - 安装 yum install ansible
  - 验证 ansible --version

案例1：环境准备

课堂练习

启动6台虚拟机

2cpu，1.5G 以上内存，10G 以上硬盘，1块网卡

主机名	Ip地址	角色
ansible	192.168.1.40	管理主机
web1	192.168.1.41	托管主机
web2	192.168.1.42	托管主机
db1	192.168.1.43	托管主机
db2	192.168.1.44	托管主机
cache	192.168.1.45	托管主机

ad-hoc

Tedu.cn
达内教育

主机管理

主机定义与分组

知识讲解

- 安装ansible之后可以做一些简单的任务
- ansible配置文件查找顺序
  - 首先检测ANSIBLE_CONFIG变量定义的配置文件
  - 其次检查当前目录下的 ./ansible.cfg 文件
  - 再次检查当前用户家目录下 ~/ansible.cfg 文件
  - 最后检查/etc/ansible/ansible.cfg文件
- /etc/ansible/ansible.cfg是ansible的默认配置文件路径

主机定义与分组（续1）

知识讲解

- ansible.cfg 配置文件
  - inventory 定义托管主机地址配置文件路径名
  - inventory 指定的配置文件，写入远程主机的地址。

- 格式
  - # 表示注释

```
[组名称]
主机名称或ip地址，其他参数
```

主机定义与分组（续2）

- ansible.cfg 配置文件
  - ssh 主机 key 验证配置参数
  - host_key_checking = False

  - 如果为 False，不需要输入 yes
  - 如果为 True，等待输入 yes

知识讲解

主机定义与分组（续3）

- 分组定义、范围定义样例
  - [web]
  - web1
  - web2

  - [db]
  - db[1:2]

  - [other]
  - cache

知识讲解

ansible命令基础

- ansible 主机集合 -m 模块名称 -a 模块参数
  - 主机集合　主机名或分组名，多个使用"逗号"分隔
  - -m　模块名称，默认 command 模块
  - -a　or　--args 模块参数
- 其他参数
  - -i　inventory文件路径，或可执行脚本
  - -k　使用交互式登录密码
  - -e　定义变量
  - -v　显示详细信息

知识讲解

ansible命令基础（续1）

知识讲解

- 列出要执行的主机
  - ansible all --list-hosts
- 批量检测主机
  - ansible all -m ping -k

部署证书文件

知识讲解

- ansible 是通过 SSH 在远程执行命令的
- ssh 远程执行命令必须要通过认证才行
- 密码写入配置文件安全性很差
- 使用key方式认证
- 给所有主机部署公钥
  - 没有秘钥命令执行会出错
  - ansible web -a 'uptime'

案例2：部署证书文件

1. 创建一对密钥

```
cd  /root/.ssh
ssh-keygen -t rsa -b 2048 -N '' -f  key
```

2. 给所有主机部署密钥

```
ssh-copy-id -i key.pub  主机名称
```

课堂练习

inventory 扩展参数

- inventory 参数说明
  - ansible_ssh_port
  - ssh端口号：如果不是默认的端口号，通过此变量设置
  - ansible_ssh_user
  - 默认的ssh用户名

知识讲解

inventory 扩展参数（续1）

- inventory 参数说明

  - ansible_ssh_pass
  - ssh密码（这种方式并不安全,我们强烈建议使用--ask-pass或SSH密钥）

  - ansible_ssh_private_key_file
  - ssh使用的私钥文件，适用于有多个密钥，而你不想使用SSH代理的情况

知识讲解

inventory 扩展参数（续2）

- inventory 参数说明
- vars 变量定义，用于组名后面
  - 例如
  - [all:vars]
  - ansible_ssh_private_key_file="/root/.ssh/key"
- children 子组定义，用于引用其他组名称
  - 例如
  - [app:children]
  - web
  - db

知识讲解

inventory 扩展参数（续3）

- 分组定义、范围定义样例
  - 子组定义

```
[app:children]
web
db
```

  - 变量引用

```
[other]
cache ansible_ssh_port=222

[all:vars]
ansible_ssh_private_key_file="/root/.ssh/key"
```

知识讲解

inventory 扩展参数（续4）

知识讲解

- 自定义配置文件
  - 创建文件夹myansible
  - 创建配置文件ansible.cfg

```
[defaults]
inventory = myhost
host_key_checking = False
```

  - 配置主机文件

```
[app1]
web1
db1
```

  - ansible app1 --list-hosts

案例3：主机定义与分组

1. 给所有主机部署 key
2. 在 inventory 文件中指定 key 的位置
3. 配置主机分组，自定义文件，在重新定义一个新的 ansible.cfg
4. 在自定义的文件夹中完成之前的配置

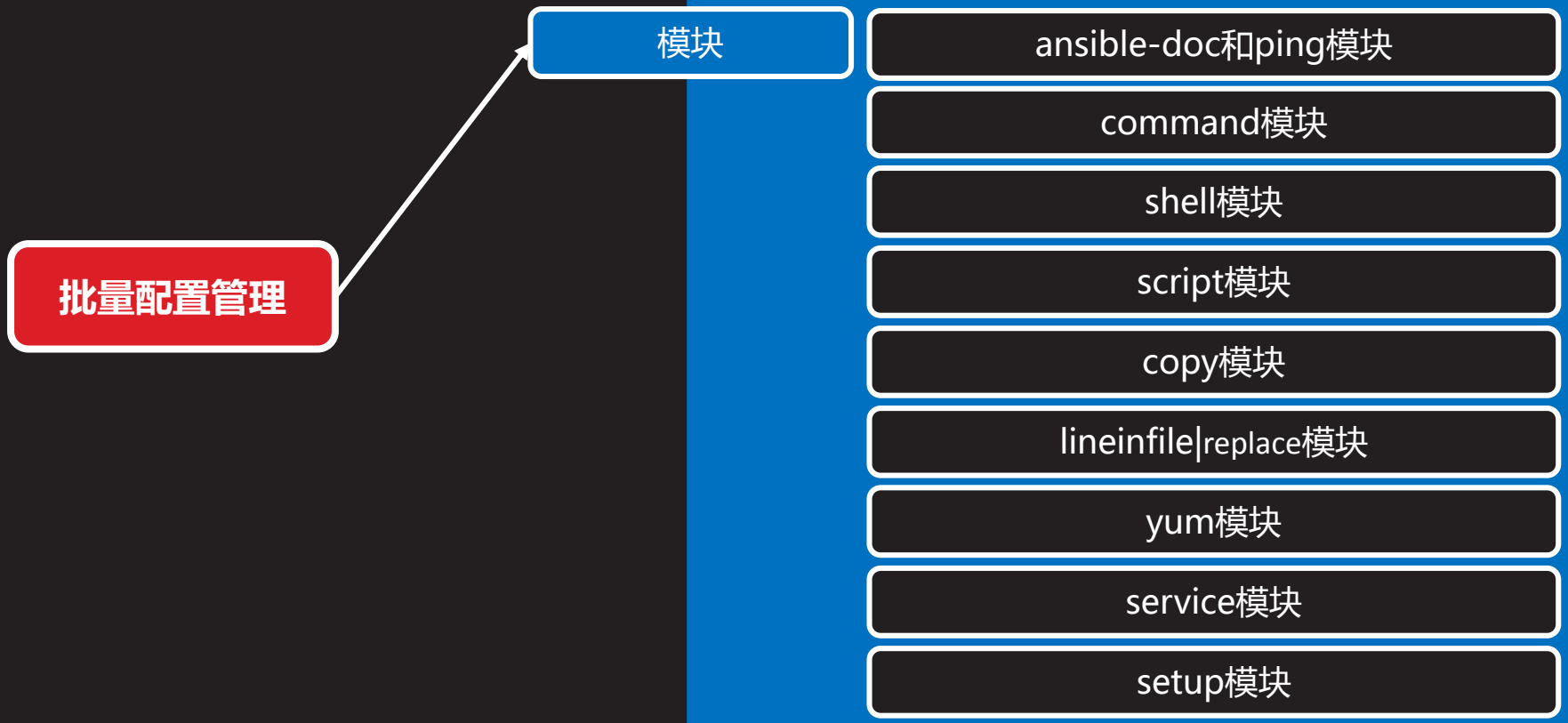
课堂练习

动态主机

- 无限可能
  - ansible Inventory包含静态和动态的Inventory，静态Inventory指在文件/etc/ansible/hosts中指定的主机和组，动态Inventory指通过外部脚本获取主机列表，按照其要求格式返回给ansilbe命令
- Json
  - JSON（JavaScript Object Notation，JavaScript对象表示法），一种基于文本独立于语言的轻量级数据交换格式

知识讲解

批量配置管理


批量配置管理
模块
ansible-doc和ping模块
command模块
shell模块
script模块
copy模块
lineinfile|replace模块
yum模块
service模块
setup模块

Tedu.cn
达内教育

模块

ansible-doc和ping模块

- ansible-doc
  - 模块的手册相当与shell的man，很重要
  - ansible-doc -l 列出所有模块
  - ansible-doc modulename 查看帮助
- ping 模块
  - 测试网络连通性, ping模块没有参数
  - 注：测试ssh的连通性

    ansible host-pattern -m ping

知识讲解

command模块

- command模块
  - 默认模块，远程执行命令
  - 用法

    ansible host-pattern -m command -a '[args]'
  - 查看所有机器负载

    ansible all -m command -a 'uptime'
  - 查看日期和时间

    ansible all -m command -a 'date +%F_%T'

知识讲解

command模块（续1）

知识讲解

- command模块注意事项：
  - 该模块通过-a跟上要执行的命令可以直接执行，若命令里有如下字符则执行不成功
  - "<"，">"，"|"，"&"
  - command 模块不能解析系统变量
  - 该模块不启动shell直接在ssh进程中执行，所有使用到shell的命令执行都会失败
  - 下列命令执行会失败

```
ansible  all  -m command -a  'ps aux|grep ssh'
ansible  all  -m command -a  'set'
```

shell模块

- shell
  - shell 模块用法基本和command一样，区别是shell模块是通过/bin/sh进行执行命令，可以执行任意命令
  - 不能执行交互式的命令，例如 vim top 等
  - 查看所有机器的负载

```
ansible all -m shell -a 'uptime'
```

知识讲解

案例4：练习理解批量执行

- shell
  - 执行以下命令查看结果，并说明原因

```
ansible  web  -m shell -a "echo ${HOSTNAME}"
ansible  web  -m shell -a 'echo ${HOSTNAME}'
```

  - testfile 文件在哪里

```
ansible  cache  -m shell -a 'cd /tmp'
ansible  cache  -m shell -a 'touch  testfile'
```

课堂练习

问题解答（续案例4）

知识讲解

- 变量解析
  - ansible 执行命令是二次解析
  - 第一次在本机解析, 第二次在执行机器解析
  - 需要第二次解析的变量要转移（\）
- 文件在哪里
  - 文件在 用户家目录
  - ansible 是使用 ssh 多次连接执行
  - 连接退出以后之前的状态就全部失效了
  - 解决方法：使用 chdir 代替 cd 命令

```
ansible cache -m shell -a 'chdir=/tmp touch testfile'
```

案例5：创建用户

- 添加用户
  - 给 web1 db2 添加用户 nb
  - 设置 nb 的密码为 123

- 思考：添加用户
  - 给所有 web 主机添加用户 wk
  - 要求 nb 用户与 wk 用户不能出现在同一台主机上
  - 设置 wk 用户的 密码是 456

课堂练习

script模块

- script模块
  - 命令太复杂？
  - 在本地写脚本，然后使用script模块批量执行

    ansible t1 -m script -a 'urscript'

  - 注意：该脚本包含但不限于shell脚本，只要指定Sha-bang解释器的脚本都可运行

知识讲解

案例6：练习模块

- 添加用户
  - 给所有 web 主机添加用户 wk
  - 要求 nb 用户与 wk 用户不能出现在同一台主机上
  - 设置 wk 用户的 密码是 456

课堂练习

yum模块

知识讲解

- yum模块
  - 使用yum包管理器来管理软件包
  - name：要进行操作的软件包名字
  - state： 动作（installed， removed）

  - install === installed
  - remove === removed

yum模块（续1）

- yum模块
  - 给所有db 主机安装 mariadb

```
ansible  db -m yum -a 'name="mariadb-server" state=installed'
```

  - cache 主机删除lrzsz

```
ansible  cache -m yum -a 'name="lrzsz" state=removed'
```

知识讲解

service模块

- service模块
  - name：必选项，服务名称
  - enabled：是否开机启动 yes|no
  - sleep：执行restarted，会在stop和start之间沉睡几秒钟
  - state：对当前服务执行启动，停止、重启、重新加载等操作（started，stopped，restarted，reloaded）

```
ansible  t1 -m service -a 'name="sshd" enabled="yes"
state="started"'
```

知识讲解

copy模块

- copy 模块
  - 复制文件到远程主机
  - src：复制本地文件到远程主机，绝对路径和相对路径都可，路径为目录时会递归复制。若路径以"/"结尾，只复制目录里的内容，若不以"/"结尾，则复制包含目录在内的整个内容，类似于rsync
  - dest：必选项。远程主机的绝对路径，如果源文件是一个目录，那该路径必须是目录

知识讲解

copy模块（续1）

知识讲解

- copy 模块
  - backup：覆盖前先备份原文件，备份文件包含时间信息。有两个选项：yes|no
  - force：若目标主机包含该文件，但内容不同，如果设置为yes，则强制覆盖，设为no，则只有当目标主机的目标位置不存在该文件时才复制。默认为yes
  - 复制文件

    ```
    ansible  all -m copy -a 'src=/etc/resolv.conf
    dest=/etc/resolv.conf'
    ```

  - 复制目录

    ```
    ansible all -m copy -a 'src=/etc/yum.repos.d/
    dest=/etc/yum.repos.d/'
    ```

案例7：练习模块

- 批量修改配置文件
  - 批量修改所有机器的 dns 配置 /etc/resolv.conf
  - 批量同步所有机器的 yum 配置文件
  - 给所有 db 主机开启 binlog 日志
  - log_bin = mysql-bin
  - binlog-format = mixed

课堂练习

lineinfile模块

- lineinfile 模块
  - 类似sed的一种行编辑替换模块
  - path　　目标文件文件
  - regexp　正则表达式，要修改的行
  - line　　最终修改的结果
  - 例如修改 my.cnf，中 bin-log 的格式
  - mixed --> row

```
ansible db -m lineinfile -a '
    path="/etc/my.cnf"
    regexp="^binlog-format"
    line="binlog-format = row" '
```

知识讲解

replace模块

- replace 模块
  - 类似sed的一种行编辑替换模块
  - path 目的文件
  - regexp 正则表达式
  - replace 替换后的结果
  - 替换指定字符 row --> mixed

```
ansible db -m replace -a '
    path="/etc/my.cnf"
    regexp="row"
    replace="mixed" '
```

知识讲解

案例8：模块练习

1. 使用copy模块同步 my.cnf 配置文件
2. 使用 lineinfile 模块 修改 binlog 格式
3. 使用 replace 模块修改 binlog 格式

课堂练习

setup模块

- setup模块
  - 主要用于获取主机信息，playbooks里经常会用的另一个参数gather_facts与该模块相关，setup模块下经常用的是filter参数
  - filter过滤所需信息

```
ansible cache -m setup -a 'filter=ansible_distribution'
```

知识讲解

Tedu.cn
达内教育

总结和答疑