

利用 Vagrant 模拟打造多机器的分布式系统

前面这些单主机单虚拟机主要是用来自己做开发机，从这部分开始的内容主要将向大家介绍如何在单机上通过虚拟机来打造分布式集群系统。这种多机器模式特别适合以下几种人：

1. 快速建立产品网络的多机器环境，例如 web 服务器、db 服务器
2. 建立一个分布式系统，学习他们是如何交互的
3. 测试 API 和其他组件的通信
4. 容灾模拟，网络断网、机器死机、连接超时等情况

Vagrant 支持单机模拟多台机器，而且支持一个配置文件 Vagrantfile 就可以跑分布式系统。

现在我们来建立多台 VM 跑起来，并且让他们之间能够相通信，假设一台是应用服务器、一台是 DB 服务器，那么这个结构在 Vagrant 中非常简单，其实和单台的配置差不多，你只需要通过 `config.vm.define` 来定义不同的角色就可以了，现在我们打开配置文件进行如下设置：

```
Vagrant.configure("2") do |config|
  config.vm.define :web do |web|
    web.vm.provider "virtualbox" do |v|
      v.customize ["modifyvm", :id, "--name", "web", "--memory", "512"]
    end
    web.vm.box = "base"
    web.vm.hostname = "web"
    web.vm.network :private_network, ip: "11.11.1.1"
  end

  config.vm.define :db do |db|
    db.vm.provider "virtualbox" do |v|
      v.customize ["modifyvm", :id, "--name", "db", "--memory", "512"]
    end
    db.vm.box = "base"
    db.vm.hostname = "db"
  end
end
```

```
    db.vm.network :private_network, ip: "11.11.1.2"
  end
end
```

这里的设置和前面我们单机设置配置类似，只是我们使用了:web 以及:db 分别做了两个 VM 的设置，并且给每个 VM 设置了不同的 hostname 和 IP，设置好之后

再使用 `vagrant up` 将虚拟机跑起来：

```
$ vagrant up
Bringing machine 'web' up with 'virtualbox' provider...
Bringing machine 'db' up with 'virtualbox' provider...
[web] Setting the name of the VM...
[web] Clearing any previously set forwarded ports...
[web] Creating shared folders metadata...
[web] Clearing any previously set network interfaces...
[web] Preparing network interfaces based on configuration...
[web] Forwarding ports...
[web] -- 22 => 2222 (adapter 1)
[web] Running any VM customizations...
[web] Booting VM...
[web] Waiting for VM to boot. This can take a few minutes.
[web] VM booted and ready for use!
[web] Setting hostname...
[web] Configuring and enabling network interfaces...
[web] Mounting shared folders...
[web] -- /vagrant
[db] Setting the name of the VM...
[db] Clearing any previously set forwarded ports...
[db] Fixed port collision for 22 => 2222. Now on port 2200.
[db] Creating shared folders metadata...
[db] Clearing any previously set network interfaces...
[db] Preparing network interfaces based on configuration...
[db] Forwarding ports...
[db] -- 22 => 2200 (adapter 1)
[db] Running any VM customizations...
[db] Booting VM...
[db] Waiting for VM to boot. This can take a few minutes.
[db] VM booted and ready for use!
[db] Setting hostname...
[db] Configuring and enabling network interfaces...
[db] Mounting shared folders...
[db] -- /vagrant
```

看到上面的信息输出后，我们就可以通过 `vagrant ssh` 登录虚拟机了，但是这次和上次使用的不一样了，这次我们需要指定相应的角色，用来告诉 `ssh` 你期望连接的是哪一台：

```
$ vagrant ssh web
vagrant@web:~$

$ vagrant ssh db
vagrant@db:~$
```

是不是很酷！现在接下来我们再来验证一下虚拟机之间的通信，让我们先使用 `ssh` 登录 `web` 虚拟机，然后在 `web` 虚拟机上使用 `ssh` 登录 `db` 虚拟机(默认密码是 `vagrant`):

```
$ vagrant ssh web
Linux web 2.6.32-38-server #83-Ubuntu SMP Wed Jan 4 11:26:59 UTC 2012 x86_64
GNU/Linux
Ubuntu 10.04.4 LTS

Welcome to the Ubuntu Server!
 * Documentation:  http://www.ubuntu.com/server/doc
New release 'precise' available.
Run 'do-release-upgrade' to upgrade to it.

Welcome to your Vagrant-built virtual machine.
Last login: Thu Aug  8 18:55:44 2013 from 10.0.2.2
vagrant@web:~$ ssh 11.11.1.2
The authenticity of host '11.11.1.2 (11.11.1.2)' can't be established.
RSA key fingerprint is e7:8f:07:57:69:08:6e:fa:82:bc:1c:f6:53:3f:12:9e.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '11.11.1.2' (RSA) to the list of known hosts.
vagrant@11.11.1.2's password:
Linux db 2.6.32-38-server #83-Ubuntu SMP Wed Jan 4 11:26:59 UTC 2012 x86_64
GNU/Linux
Ubuntu 10.04.4 LTS

Welcome to the Ubuntu Server!
 * Documentation:  http://www.ubuntu.com/server/doc
New release 'precise' available.
Run 'do-release-upgrade' to upgrade to it.

Welcome to your Vagrant-built virtual machine.
Last login: Thu Aug  8 18:58:50 2013 from 10.0.2.2
vagrant@db:~$
```

通过上面的信息我们可以看到虚拟机之间通信是畅通的，所以现在开始你伟大的架构设计吧，你想设计怎么样的架构都可以，唯一限制你的就是你主机的硬件配置了。

转到本页之前的地址 <http://i.qwluoye.com/vagrant.html>

本 pdf 文档存储服务由 IT7 主机提供 <http://i.qwluoye.com/url/bwg>

由 Qwluoye 转载自 github.com | 2017 年 4 月 14 日 20:44:28