Hubot in the Cloud

Wes McClure
wesmcclure.com

pluralsight
hardcore dev and IT training

Reuse

- Provisioning scripts
- Upstart service
- Rsync

Additional Learning

- Vagrant multi machine
- Vagrant providers
- vagrant-aws plugin
- Vagrant plugins
- AMIs
- Dev/Prod config
- Cleanup

Why Multiple Machines?

- Development and Production environments
- Separate web and database machines
- Test replication in multi node MongoDB

Providers

- **Dev uses VirtualBox**
- **Prod uses AWS**
- **Providers**
- **OOB – VirtualBox, HyperV, Docker**
- **Provider plugins**
  - AWS
  - VMWare (HashiCorp)

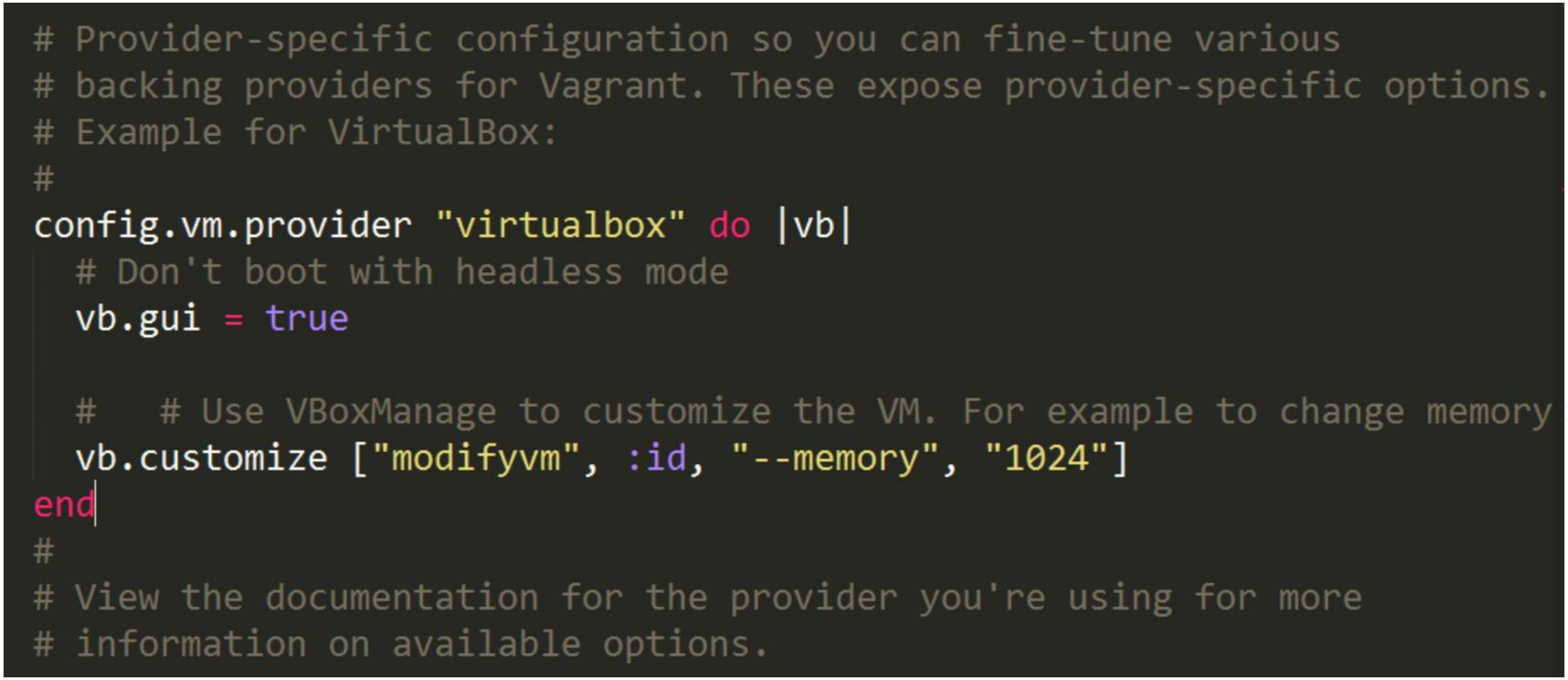
Boxes and Providers

- Base image specific to virtualization system
- Historically, 1 provider per box
- Vagrant 1.5, multiple providers per box

Changing Providers

- vagrant up --provider aws
- OOB defaults VirtualBox
- VAGRANT_DEFAULT_PROVIDER environment variable

Configuring Providers



```
# Provider-specific configuration so you can fine-tune various
# backing providers for Vagrant. These expose provider-specific options.
# Example for VirtualBox:
#
config.vm.provider "virtualbox" do |vb|
  # Don't boot with headless mode
  vb.gui = true

  #   # Use VBoxManage to customize the VM. For example to change memory
  vb.customize ["modifyvm", :id, "--memory", "1024"]
end
#
# View the documentation for the provider you're using for more
# information on available options.
```

Configuring AWS Provider

- Dummy box
- Region specific AMI
- Name EC2 instance
- SSH key pair
- AWS credentials

The Dummy Box

- Local VM uses local image
- Cloud VM uses cloud image
- AWS uses AMI id
- Dummy placeholder

SSH and Security

- **Private Dev Machine – convenience over security**
  - Insecure vagrant ssh key
  - Default vagrant user and password
- **Production – security is serious**
  - AWS key pair
  - ubuntu user for remote access (not vagrant)

Application Configuration

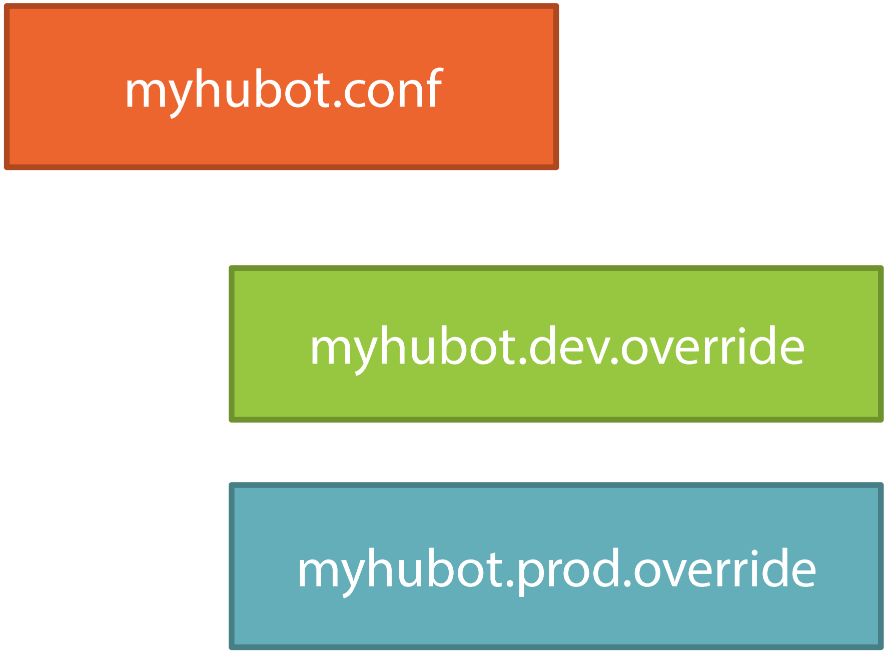
- **dev config != prod config**
  - Connection strings
  - User names
  - Passwords
- **dev HipChat != prod HipChat**

Separate Job Configuration Files?

myhubot.dev.conf

myhubot.prod.conf

Upstart Job Override File



myhubot.conf

myhubot.dev.override

myhubot.prod.override

Development Workflow

- **Checkout repository**
- **vagrant up dev**
- **vagrant rsync-auto dev**
- **New/Modify script**
  - Write on host
  - rsync to guest
  - vagrant ssh dev
  - sudo service myhubot restart
  - hubot die

Development Workflow

- **vagrant provision dev**
  - new npm package
  - Change upstart job config
- **Testing**
  - grunt/gulp
  - watch for changes
- **Commit**

Cleanup

- **vagrant destroy prod**
  - Check EC2 console instances and volumes
- **Delete or inactivate security keys**
  - Account -> Security Credentials
- **Revert security group changes**
- **Revert key pairs**
  - ~/.ssh folder locally hubot.pem

Vagrant / AWS Considerations

- **IAM security**
- **Automate deploys**
  - Store keys in secure environment
  - Consistent provisioning
  - Avoid releasing changes in dev
  - vagrant EC2 instance state
- **Box wrappers**
  - Ami & region config
- **ELB load balancer support**

Benefits

- Model multiple machines
- Provisioning reuse
- Self documenting provisioning
- Dev workflow meshes w/ prod hosting
- Confidence
- Prod environment history
- Environment rollback

Different Needs

- Don’t use service in dev?
- Ops staff -> prod change testing?
- Don’t use ubuntu?
- Don’t use upstart services?

- Key: morph safely with VCS