Enabling successful DevOps practices using Puppet

Nick Barringer
Puppetlabs
What is DevOps?

DevOps is a software development method that stresses communication, collaboration and integration between software developers and information technology (IT) operations professionals. DevOps is a response to the interdependence of software development and IT operations. It aims to help an organization rapidly produce software products and services.


11-2-27
What is DevOps?

DevOps is a software development method that stresses communication, collaboration and integration between software developers and information technology (IT) operations professionals. DevOps is a response to the interdependence of software development and IT operations. It aims to help an organization rapidly produce software products and services.
What is DevOps?

- Software development method
- Communication
- Collaboration
- Integration
- Rapidly produce software
Demo

• Use Puppet to deploy Tomcat and Jenkins onto a linux node
  • Separate modules for Tomcat and Jenkins
  • Assigned to the node via a single group
Seven essential DevOps capabilities provided by Puppet
DevOps essentials #1

Accessability...

def create
    File.open(@resource[:name], "w") { |f| f.puts "" } # Create an empty file
    # Make sure the mode is correct
    should_mode = @resource.should(:mode)
    unless self.mode == should_mode
        self.mode = should_mode
    end
end

# Return the mode as an octal string, not as an integer.
def mode
    if File.exists?(@resource[:name])
        "%o" % (File.stat(@resource[:name]).mode & 007777)
    else
        :absent
    end
end

# Set the file mode, converting from a string to an integer.
def mode=(value)
    File.chmod(Integer("0" + value), @resource[:name])
end
DevOps essentials #1
Everyone should be able to use it!

```bash
# /root/examples/break_ssh.pp
package { 'openssh-server':
    ensure => present,
    before => File['/etc/ssh/sshd_config'],
}

file { '/etc/ssh/sshd_config':
    ensure => file,
    mode   => 600,
    source => '/root/examples/sshd_config',
}

service { 'sshd':
    ensure => running,
    enable => true,
    subscribe => File['/etc/ssh/sshd_config'],
}
```
DevOps essentials #2
Separate configuration data from modules

```
DEFINE RESOURCES

“Ensure Apache is installed, configured, and running”

COMPOSE STACKS

“Ensure a LAMP stack on top of RHEL”
```
DevOps essentials #2
Separate configuration data from modules

```yaml
# In this example, $parameter's value gets set when `myclass` is eventually declared.
# Class definition:
class myclass ($parameter_one = "default text") {
    file {'/tmp/foo':
        ensure => file,
        content => $parameter_one,
    }
}
```

```yaml
# /etc/puppet/hieradata/web01.example.com.yaml
---
myclass::parameter_one: "This node is special, so we're overriding the common configuration."
```
```
# /etc/puppet/hieradata/common.yaml
---
myclass::parameter_one: "This node can use the standard configuration."
```
DevOps essentials #3
Manage environments as code

branch: dev → cd_demo_cm_repo / hiera / environments / +

Update DB address for production

root authored 4 months ago
ccaum committed 4 months ago

dev.json: Fix dynamic app configuration per environment
production.json: Update DB address for production
staging.json: Fix dynamic app configuration per environment
uat.json: Pre-existing infrastructure
DevOps essentials #3
Manage environments as code
DevOps essentials #4

Automation
DevOps essentials #5
Scale to thousands of nodes

- Define Resources
  - "Ensure Apache is installed, configured, and running"

- Compose Stacks
  - "Ensure a LAMP stack on top of RHEL"

- Automate Infrastructure
  - "Stand-up a LAMP-on-RHEL stack on 100 nodes, and then enforce configurations"
DevOps essentials #5

Scale to thousands of nodes

Joint project between University of Tennessee and Oak Ridge national laboratory

One of the world's most powerful computing complex

- 16,512 compute sockets
- 129 Tb memory
- 3.3 Pb disk space
- >700m CPU hours pa to 2000 active researchers
DevOps essentials #5
Scale to thousands of nodes
DevOps essentials #6
Use version control

puppetlabs/vcsrepo

Author: Puppet Labs
Links: Project URL | Report issues

Tags: vcs, repo, svn, subversion, git, hg, bzr, CVS

vcsrepo { "/opt/code/${repo}"
    ensure => latest,
    owner => $owner,
    group => $owner,
    provider => git,
    require => [ Package["git"] ],
    source => "http://giturl.com/proj/${repo}.git",
    revision => 'master',
}
DevOps essentials #7
Support cross-team workflows

Dev
1. App Code Check in
2. Build RPM on Jenkins
3. Upload App RPM and Manifest RPM to Dev Servers

QA
1. Promote App RPM, Manifest RPM to QA
2. Install latest manifest
3. Update Hiera file for configuration change

CS
1. Promote App RPM, Manifest RPM to Prod
2. Install Latest Manifest
3. Update Hiera file for configuration change
4. Update nodes PP to map nodes to newer version of modules
5. Check in manifests to GIT/SVN
Demo

- Use Puppet to deploy Tomcat and Jenkins onto a linux node
  - Separate modules for Tomcat and Jenkins
  - Assigned to the node via a single group
In summary...

Puppet:

- Is an easily accessible solution
- Provides abstraction of configuration data from deployment code
- Enables management of environments
- Supports automation via workflow software
- Scales to manage thousands of nodes
- Works with version control systems
- Supports cross-team workflows
In summary…

Puppet enables:
• Continuous delivery

Puppet supports:
• Continuous integration
• Continuous deployment