SMOOTHING TROUBLES

Custom Types and Providers

Chad Thompson
PuppetCamp Chicago
August 18, 2014

CHAD THOMPSON

"DevOps Engineer"

Dice Holdings, Inc.

Leading websites for professional communities

The IT Job Board
Your search made simple

ClearanceJobs.com
A Dice Service

Slashdot

Hcareers

BioSpace

Health Callings
JOBS THAT MATTER

Dice

RIGZONE

efc efinancialcareers

healthecareers NETWORK

oilcareers.com

http://chadthompson.me

From Puppet Manifests...

```
node default {

   file{"/etc/config.conf:
      ensure   => "present",
      owner    => "chad.thompson",
      content  => "Key Value”,
   }

}
```

... TO MODULES

```
node default {
  include custommod
}
```

```
class custommod {
  file{“/etc/config.conf”:
      ensure   => "present",
      owner    => "chad.thompson",
      content  => "Key Value”,
   }
}
```

… TO DEFINED TYPES

```
node default {
  custommod::config{“config”:
     config_dir => “/etc/“,
}
```

```
define custommod::config(
  $config_dir = “/etc/“
)
 {
  file{“/etc/config.conf: […]
  }
}
```

Encapsulation

```
 user{“chad.thompson”:
    ensure      => "present",
    managehome => true,
}
```

DEPENDENCY MANAGEMENT

```
 package{“nginx”:
    ensure => installed,
 }

 file{“/etc/nginx/nginx.conf”:
    ensure  => "present",
    content => template(“..”),
    require => Package[“nginx”],
    notify  => Service[“nginx”],
}

service{“nginx”:
  ensure  => running,
  require => Package[“nginx”],
}
```

PROCEDURAL ROUTINES

```
# Copy SSH keys into place
file{"/home/devuser/.ssh":
  ensure => "directory",
  owner  => 'dice',
}->
exec{"cp /opt/ssh-keys/* /home/devuser/.ssh/":
  path    => "/bin:/usr/bin",
  user    => 'devuser',
  creates => '/home/devuser/.ssh/id_rsa.pub',
}->
exec{"chown devuser:devuser /home/devuser/.ssh/*":
  path => "/bin:/usr/bin",
  user => "devuser",
}->
exec{"chmod -R 0400 /home/devuser/.ssh/*":
  path => "/bin:/usr/bin",
}->
```

exec{} -> exec{} -> exec{}->

DIFFICULT THINGS..

- **Executing deployment scripts.**
  - Deployment scripts (“php artisan…”)
  - Database migrations
- **Third party service integrations.** (“Send a note to HipChat…”)

A DIFFERENT WAY..

```
wpcli{“/usr/local/bin”:
  ensure => present,
}
```

http://wp-cli.org

A CUSTOM TYPE

```
# lib/puppet/type/wpcli.rb
Puppet::Type.newtype(:wpcli) do
  @doc = "Install The WP-CLI Tool"
  ensurable

  # A single parameter -
  # the path to install the wp-cli binary.
  newparam(:path, :namevar => true) do
    desc "The wp-cli install location."
  end
end
```

Defines the behavior of the ‘wpcli’ type.

A PROVIDER

```
require 'mkmf'

Puppet::Type.type(:wpcli).provide :ruby do
  desc "..."

  # Required executables for the module
  commands :curl => 'curl'
  commands :php => 'php'
…
end
```

Defines the Steps for Execution

SOME BACKGROUND..

- The Ruby Language (http://ruby-lang.org)
  - Libraries, Objects, Gem Files (oh, my!)
- There can be many providers for a type.
  - Test for features / facts.
- A convention for the 'lib' directory.

```
.
├── Modulefile
├── README
├── lib
|   └── puppet
|       ├── provider
|       |   └── wpcli
|       |       └── ruby.rb
|       └── type
|           └── wpcli.rb
├── manifests
|   └── init.pp
├── spec
|   └── spec_helper.rb
└── tests
    ├── init.pp
    └── wpcli.pp
```

THE CUSTOM TYPE

```
# lib/puppet/type/wpcli.rb
Puppet::Type.newtype(:wpcli) do
  @doc = "Install The WP-CLI Tool"
  ensurable

  # A single parameter -
  # the path to install the wp-cli binary.
  newparam(:path, :namevar => true) do
    desc "The wp-cli install location."
  end
end
```

THE PROVIDER

```
require 'mkmf'

Puppet::Type.type(:wpcli).provide :ruby do
  desc "..."

  def create
 end

  def destroy
 end

  ## This provider will create a binary called 'wp'
  def exists?
  end

end
```

```
## This provider will create a binary called 'wp'
def exists?
  executable = find_executable 'wp'
  if executable.nil?
    return false
  else
    return true
  end
end
```

```
 # Required executables for the module
  commands :curl => 'curl'
  commands :php => 'php'

  @@wp_url = 'https://raw.githubusercontent.com/wp-cli/
builds/gh-pages/phar/wp-cli.phar'

 def create
    # Using the puppet logger to output
    # information - sends information
    # to the puppet master
    debug("CREATE #{resource[:path]}")

    curl(['-o', "#{resource[:path]}/wp", @@wp_url])
    File.chmod(0755, "#{resource[:path]}/wp")
 end
```

CONCLUSION

Q: How do I mix scripting with Puppet?

A: With Custom Types/Providers!

THANK YOU!

http://chadthompson.me
@chadothompson
chad_thompson@mac.com