Managing Files via Puppet

Let Me Count The Ways

Mike Arnold (razorsedge)

Puppet Camp Phoenix 2015

Agenda

- Intro
- Managing Files
- Managing Lines
- Various Patterns
- Questions
- Links

Intro

Who is Mike Arnold?

- Unix Systems Administrator
- Puppet Certified Professional
- 15 years in IT
- Presently doing:
  - Hadoop Infrastructure Engineering
  - Building Puppet modules

What Is This Topic?

- Puppet can manage files. (This may be obvious.)
    - entire files
    - or just lines
    - via static content
    - or templates

Lets see all the ways.

Managing Files

Basic File resource:

```
file { '/tmp/testFile':
  ensure => present,
  mode   => '0644',
  owner  => 'root',
  group  => 'root',
}
```

With source attribute:

```
file { '/tmp/testFileA':
  ensure => present,
  mode   => '0644',
  source => 'puppet:///modules/example/fileA',
}
```

With multiple source attributes:

```
file { '/tmp/testFileB':
  ensure => present,
  mode   => '0644',
  source => [
    "puppet:///modules/example/fileB.${::operatingsystem}",
    'puppet:///modules/example/fileB',
  ],
}
```

With content attribute:

```
file { '/tmp/testFileC':
  ensure  => present,
  mode    => '0644',
  content => 'Some fancy string.',
}
```

Note: no carriage return

With content attribute string variables:

```
file { '/tmp/testFileD':
  ensure  => present,
  mode    => '0644',
  content => "Your operating system is: ${::operatingsystem}\
${::operatingsystemrelease}\nYour CPU architecture\
is: ${::architecture}\n",
}
```

With content attribute template():

```
$variableA = 'good'
$variableB = 'horrible'
file { '/tmp/testFileE':
  ensure  => present,
  mode    => '0644',
  content => template('example/templateA.erb'),
}
```

```
This is a <%= @variableA %> day.
There will be a <%= @variableB %> calamity.
```

With mutiple content attribute template()s:

```
$variableA = 'good'
$variableB = 'horrible'
file { '/tmp/testFileF':
  ensure  => present,
  mode    => '0644',
  content => template('example/templateA.erb','example/templateB.erb'),
}
```

```
We are using <%= scope['::osfamily'] %> osfamily.
```

With content attribute inline_template():

```
$options = [ 'blue', 'heavy', 'yummy' ]
$item = 'The sky is'
file { '/tmp/testFileG':
  ensure  => present,
  mode    => '0644',
  content => inline_template("${item}: <%= @options.join ' ' %>\n"),
}
```

With content attribute file():

```
file { '/tmp/testFileH':
  ensure  => present,
  mode    => '0644',
  content => file('/etc/hosts'),
}
```

With content attribute epp():

```
$variableA = 'good'
$variableB = 'horrible'
file { '/tmp/testFileI':
  ensure  => present,
  mode    => '0644',
  content => epp('example/templateA.epp'),
}
```

```
This is a <%= $variableA %> day.
There will be a <%= $variableB %> calamity.
```

puppet apply --parser=future

Difference between source and content?

Unlike `content`, the `source` attribute can be used to recursively copy directories if the `recurse` attribute is set to `true` or `remote`.

concat

The concat module constructs files from multiple fragments in an ordered way.

```
concat { '/tmp/testFileJ':
  ensure => present,
  mode   => '0644'
}
concat::fragment { 'testfileJ-01':
  target  => '/tmp/testFileJ',
  content => "This is a concat line 01.\n",
  order   => '01',
}
concat::fragment { 'testfileJ-02':
  target => '/tmp/testFileJ',
  source => 'puppet:///modules/example/fileA',
  order  => '02',
}
```

Managing Lines

augeas

Apply a change or an array of changes to the filesystem using the augeas tool.

```
file { '/tmp/testFileK':
  ensure  => present,
  mode    => '0644',
  content => file('/etc/resolv.conf'),
} ->
augeas { 'testFileK' :
  incl    => '/tmp/testFileK',
  lens    => 'Resolv.lns',
  changes => 'set domain example.net',
}
```

file_line

The file_line resource ensures that a given line, including whitespace at the beginning and end, is contained within a file. If the line is not present, Puppet will add the line. Multiple resources can be declared to manage multiple lines in the same file.

```
file { '/tmp/testFileL':
  ensure => present,
  mode   => '0644',
} ->
file_line { 'testFileL':
  ensure => present,
  path   => '/tmp/testFileL',
  line   => 'This line shall be present.',
}
```

file_line with line replacement:

You can also use `match` to replace existing lines.

```
file { '/tmp/testFileM':
  ensure  => present,
  mode    => '0644',
  content => file('/etc/resolv.conf'),
} ->
file_line { 'testFileM#search':
  ensure => present,
  path   => '/tmp/testFileM',
  line   => 'search localdomain',
  match  => '^search .*',
}
```

inifile

The inifile module allows you to manage settings and subsettings in INI-style configuration files. This module tries hard not to manipulate your file any more than it needs to. In most cases, it should leave the original whitespace, comments, ordering, etc. intact.

```
ini_setting { 'testFileN#foo#bar':
  ensure  => present,
  path    => '/tmp/testFileN',
  section => 'foo',
  setting => 'bar',
  value   => 'GIBBERISH',
}
```

inifile adding to an existing section:

```
file { '/tmp/testFileO':
  ensure  => present,
  mode    => '0644',
  content => file('/usr/share/perl5/vendor_perl/XML/SAX/ParserDetails.ini'),
} ->
ini_setting { 'testFileO#foo#bar':
  ensure  => present,
  path    => '/tmp/testFileO',
  section => 'Build',
  setting => 'testN',
  value   => 'WeDidIt',
}
```

datacat

The datacat module constructs a file by stitching line fragments together into the same line in one or multiple files.

```
datacat { '/tmp/testFileP':
  ensure   => present,
  mode     => '0644',
  template => 'example/templateP.erb',
}
datacat_fragment { "${::fqdn} in device hostgroup":
  target => '/tmp/testFileP',
  data   => { myhostgroup => [ $::fqdn ], },
  order  => '01',
}
$ilo_fqdn = regsubst($::fqdn, '\.', '-ilo.')
datacat_fragment { "${ilo_fqdn} in device hostgroup":
  target => '/tmp/testFileP',
  data   => { myhostgroup => [ $ilo_fqdn ], },
  order  => '02',
}
```

The File and Concat resource can make backups of the file being modified into the Puppet filebucket. File_line, inifile, etc do not.

Various Patterns

sudo

```
class { 'sudo':
  purge               => false,
  config_file_replace => false,
}
sudo::conf { 'web':
  source => 'puppet:///modules/example/etc/sudoers.d/web',
}
sudo::conf { 'admins':
  priority => 10,
  content  => "%admins ALL=(ALL) NOPASSWD: ALL",
}
sudo::conf { 'joe':
  priority => 60,
  source   => 'puppet:///modules/example/etc/sudoers.d/joe',
}
```

augeasproviders

```
sshd_config { 'PermitRootLogin':
  ensure => present,
  value  => 'no',
}
```

grep PermitRootLogin /etc/ssh/sshd_config

augeasproviders

```
kernel_parameter { 'elevator':
  ensure => present,
  value  => 'deadline',
}
```

grep linux16 /boot/grub2/grub.cfg

augeasproviders

```
shellvar { 'HOSTNAME':
  ensure => present,
  target => '/etc/sysconfig/network',
  value  => 'host.example.com',
}
```

cat /etc/sysconfig/network

augeasproviders

```
sysctl { 'net.ipv4.ip_forward':
  ensure  => present,
  value   => '1',
  comment => 'This is a routing test.',
}
```

sysctl net.ipv4.ip_forward
cat /etc/sysctl.conf

Apache vhost

```
apache::vhost { 'first.example.com':
  port    => '80',
  docroot => '/var/www/first',
}
```

cat /etc/httpd/conf.d/25*

Questions?

Links

- https://docs.puppetlabs.com/references/latest/type.html#file-attribute-content
- https://docs.puppetlabs.com/references/latest/type.html#file-attribute-source
- https://docs.puppetlabs.com/references/latest/function.html#template
- https://docs.puppetlabs.com/references/latest/function.html#file
- https://docs.puppetlabs.com/references/latest/function.html#epp
- https://forge.puppetlabs.com/puppetlabs/concat

Links

- https://docs.puppetlabs.com/references/latest/type.html#augeas
- https://puppetlabs.com/blog/module-of-the-week-puppetlabsstdlib-puppet-labs-standard-library
- https://forge.puppetlabs.com/puppetlabs/stdlib
- https://forge.puppetlabs.com/puppetlabs/inifile
- https://forge.puppetlabs.com/richardc/datacat

Links

- https://forge.puppetlabs.com/saz/sudo
- https://forge.puppetlabs.com/herculesteam/augeasproviders
- https://forge.puppetlabs.com/puppetlabs/apache

Contact

- puppet@razorsedge.org
- https://intelligentsysadmin.wordpress.com/
- https://github.com/razorsedge
- https://forge.puppetlabs.com/razorsedge

This presentation can be found at: https://github.com/razorsedge/presentation-managing-files-via-puppet