

 Recent

 Topics

“ Highlights

 Settings

Feedback

Sign Out

Settings

10 days left in your trial. **Subscribe.**

Feedback

Sign Out

Running Puppet fr...

We'd like to make it as quick and easy as possible to apply Puppet on machine, so I usually write a little script that wraps the `puppet apply` command with the parameters it needs. And to deploy the script where needed, what better tool than Puppet itself?

Follow these steps:

1. In your Puppet repo, create the directories needed for a puppet module:

```
ubuntu@cookbook:~/puppet$ mkdir modules
ubuntu@cookbook:~/puppet$ mkdir modules/puppet
ubuntu@cookbook:~/puppet$ mkdir modules/puppet/manifests
ubuntu@cookbook:~/puppet$ mkdir modules/puppet/files
```

2. Create the file `modules/puppet/files/papply.sh` with the following contents (change the path `/home/ubuntu/puppet` to where your Puppet repo is located). The `sudo puppet apply` command should all be on one line:

```
#!/bin/sh
sudo puppet apply /home/ubuntu/puppet/manifests/site.pp
--modulepath=/home/ubuntu/puppet/modules/ $*
```

3. Create the file `modules/puppet/manifests/init.pp` with the following contents:

```
class puppet {
  file { ['/usr/local/bin/papply':
    source => 'puppet:///modules/puppet/papply.sh',
    mode   => '0755',
  ]
}
```

4. Modify your `manifests/nodes.pp` file as follows:

```
node 'cookbook' {
  include puppet
}
```

5. Apply your changes:

```
ubuntu@cookbook:~/puppet$ sudo puppet apply manifests/site.  
--modulepath=/home/ubuntu/puppet/modules  
Notice: /Stage[main]/Puppet/File[/usr/local/bin/papply]  
/ensure: defined content as '{md5}  
171896840d39664c00909eb8c47a753c'  
Notice: Finished catalog run in 0.07 seconds
```

6. Test that the script works:

```
ubuntu@cookbook:~/puppet$ puppet
Notice: Finished catalog run in 0.07 seconds
```

Now whenever you need to run Puppet, you can simply run `papply`. In future, when we apply Puppet changes, I'll ask you to run `papply` instead of the full `puppet apply` command.

As you've seen, to run Puppet on a machine and apply a specified manifest file, we use the `puppet apply` command:

```
puppet apply manifests/site.pp
```

When you're using modules (such as the puppet module we just created) you also need to tell Puppet where to search for modules, using the `modulepath` argument:

```
puppet apply manifests/nodes.pp --  
  modulepath=/home/ubuntu/puppet/modules
```

In order to run Puppet with the root privileges it needs, we have to put `sudo` before everything:

```
sudo puppet apply manifests/nodes.pp --  
  modulepath=/home/ubuntu/puppet/modules
```

Finally, any additional arguments passed to `papply` will be passed through to Puppet itself, by adding the `*$` parameter:

```
sudo puppet apply manifests/nodes.pp --  
  modulepath=/home/ubuntu/puppet/modules $*
```

That's a lot of typing, so putting this in a script makes sense. We've added a Puppet `file` resource that will deploy the script to `/usr/local/bin` and make it executable:

```
file { ['/usr/local/bin/papply']:  
  source => 'puppet:///modules/puppet/papply.sh',  
  mode   => '0755',  
}
```

Finally, we include the `puppet` module in our node declaration for `cookbook`:

```
node 'cookbook' {  
  include puppet  
}
```

You can do the same for any other nodes managed by Puppet.

◀ PREVIOUS
Creating a decent...

NEXT ▶
Running Puppet fr...

[Recommended](#) / [Queue](#) / [Recent](#) / [Topics](#) / [Settings](#) / [Blog](#)(<http://blog.safaribooksonline.com>) /

[Support](#)(<http://msupport.safaribooksonline.com/?prod=flow>) / [Feedback](#) / [Sign Out](#)

© 2015 Safari(<http://www.safaribooksonline.com/>).

[Terms of Service](#) / [Membership Agreement](#) / [Privacy Policy](#)