

# Error: undefined method downcase

I've got multiple versions of puppet installed on my machine:

- 1) 3.x for regular use
- 2) 2.7 to generate puppetdoc. Puppetdoc is buggy in 3.x so I had to install an older version

I installed puppet 2.7 in my user's .gem folder under the home dir (~/.gem). When I run puppet, I get an error:

```
Error: undefined method `downcase' for # at  
/vagrant/dev_puppet/manifests/site.pp:1 on node
```

It is either a bug in puppet 2.7 or I am missing some gems in my 2.7 .gem installation. The temporary fix is to rename .gem folder when running puppet runs on 3.x, and move it back when i need to run puppetdoc from the .gem folder.

A more elegant solution would have been to install puppet into it's own environment using rvm or something similar.

---

## create\_resources and hiera

When using create\_resources to instantiate defined resources with hiera, it is a good practice to include the last parameter to the create\_resources function – the defaults parameter. The easiest thing is to set the default to an empty hash '{}'. If you don't do this AND you don't list the data item in hiera, you will see an error like:

```
Error: Could not find data item etc_host in any Hiera data
file      and      no      default      supplied      at
/vagrant/dev_puppet/modules/hosts/manifests/hiera.pp:4 on node
foo.foo.com
```

The fix is to set the default as an empty hash:

```
class hosts::hiera {

$etc_host = hiera('etc_host', {}) )
create_resources('hosts::etchosts', $etc_host)

}
```

---

## [Use FPM to create java jdk RPM package](#)

Creating packages from redhat or debian systems can be a royal pain. [FPM](#) takes some of the pain away. Refer to the FPM wiki for install and usage instructions.

### **To create an RPM for Oracle's (sun) java jdk 7:**

1. Download jdk7 from Oracle site. (example, jdk1.7.0\_65)
2. untar into /opt folder, (result: /opt/jdk1.7.0\_65)
3. create alternatives.sh script in the exploded folder
4. create uninstall.sh script in the same location
5. run fpm command
6. (recommended) upload rpm to your custom yum repo

```
#alternatives.sh
#!/bin/bash
alternatives      --install      /usr/bin/java      java
/opt/jdk1.7.0_65/bin/java 1
```

```
#uninstall.sh
```

```
#!/bin/bash
```

```
alternatives --remove java /opt/jdk1.7.0_65/bin/java
```

```
rm -rf /opt/jdk1.7.0_65
```

Run the fpm command to generate an rpm

```
fpm -s dir -t rpm --after-install  
/opt/jdk1.7.0_65/alternatives.sh --after-remove  
/opt/jdk1.7.0_65/uninstall.sh --name jdk7 -v 1.7.0.65  
/opt/jdk1.7.0_65
```

---