Relax-and-Recover (ReaR) Mass Deployment

Gratien D’haese

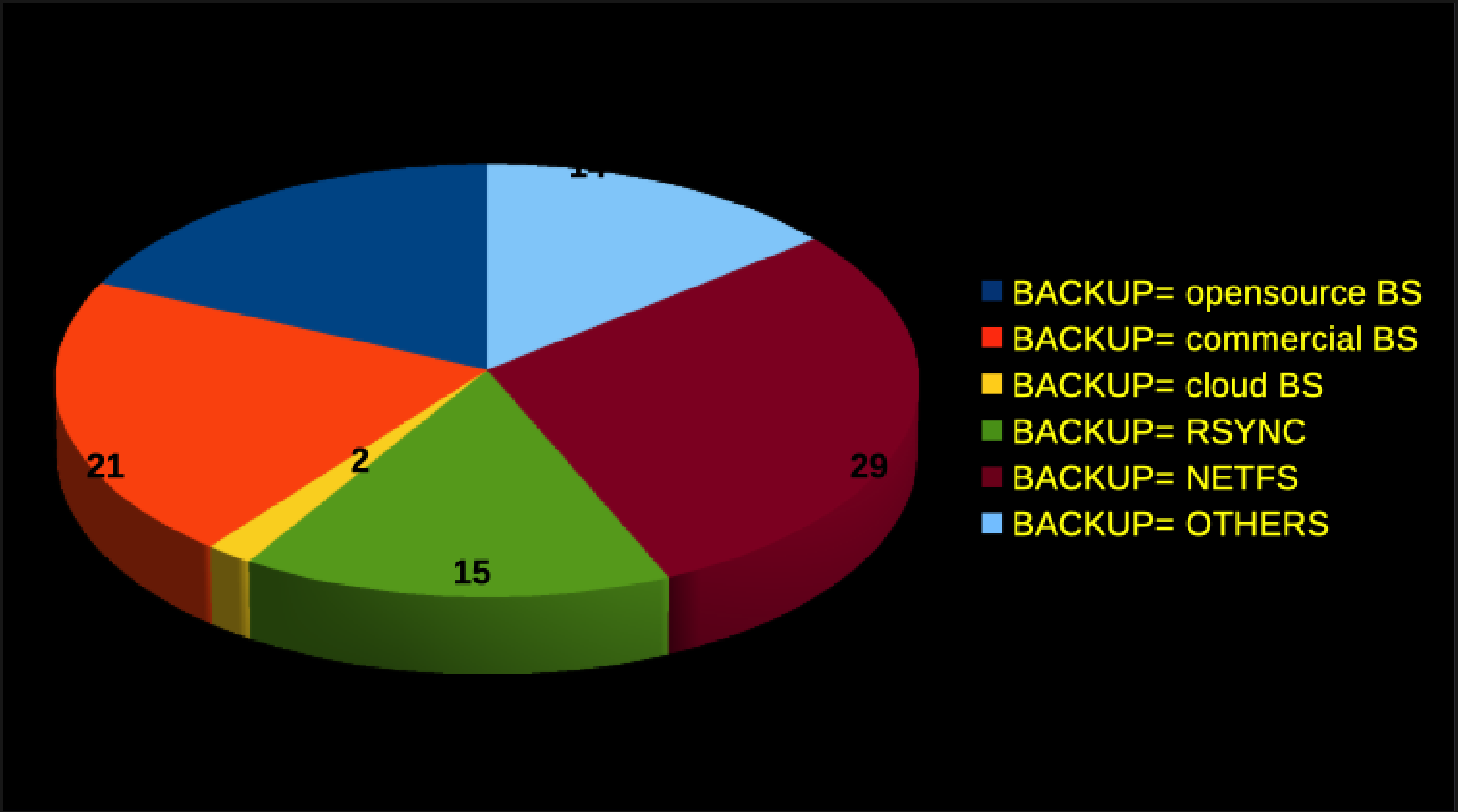
https://github.com/gdha/rear-masses

A story about

35k

Popular Backup Schemes

- From ReaR poll of 2015



Cloud vs. in-house

- Amazon
- Azure
- Google
- VMware vSphere VMs
- Physical servers
- Older Linux systems

To ReaR or not to ReaR

- Million dollar question
  - **why using ReaR at all?**
- AWS, Azure: no ReaR (using the disk snapshots)
- Vsphere VMs: backup snapshot → no ReaR
- Vsphere VMs: regular backups → use ReaR
- All others: use ReaR

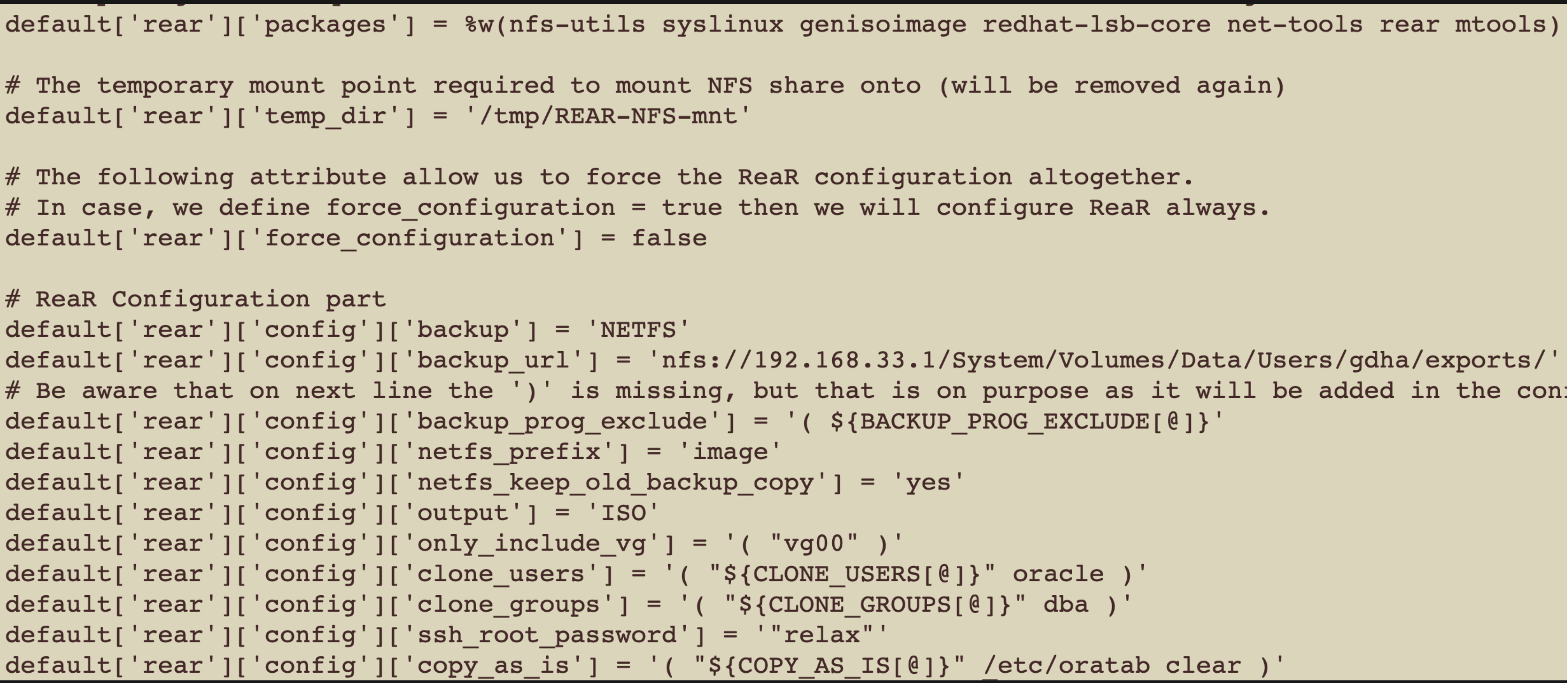
Automation is key

- Configuration management by Chef
- Cookbook “rear” capabilities:
  - Install rear
  - Configure rear
  - Unconfigure rear
  - Remove rear
- End-user ability to control attributes

Rear Cookbook

- https://github.com/gdha/rear-masses/tree/master/cookbooks/rear
- Cookbook is tested on RHEL/CentOS 6, 7 & 8
- ReaR versions 1.17.x, 2.0 and 2.4 are recognized by recipes

ReaR attributes



```
default['rear']['packages'] = %w(nfs-utils syslinux genisoimage redhat-lsb-core net-tools rear mtools)

# The temporary mount point required to mount NFS share onto (will be removed again)
default['rear']['temp_dir'] = '/tmp/REAR-NFS-mnt'

# The following attribute allow us to force the ReaR configuration altogether.
# In case, we define force_configuration = true then we will configure ReaR always.
default['rear']['force_configuration'] = false

# ReaR Configuration part
default['rear']['config']['backup'] = 'NETFS'
default['rear']['config']['backup_url'] = 'nfs://192.168.33.1/System/Volumes/Data/Users/gdha/exports/'
# Be aware that on next line the ')' is missing, but that is on purpose as it will be added in the con
default['rear']['config']['backup_prog_exclude'] = '( ${BACKUP_PROG_EXCLUDE[@]}'
default['rear']['config']['netfs_prefix'] = 'image'
default['rear']['config']['netfs_keep_old_backup_copy'] = 'yes'
default['rear']['config']['output'] = 'ISO'
default['rear']['config']['only_include_vg'] = '( "vg00" )'
default['rear']['config']['clone_users'] = '( "${CLONE_USERS[@]}" oracle )'
default['rear']['config']['clone_groups'] = '( "${CLONE_GROUPS[@]}" dba )'
default['rear']['config']['ssh_root_password'] = '"relax"'
default['rear']['config']['copy_as_is'] = '( "${COPY_AS_IS[@]}" /etc/oratab clear )'
```

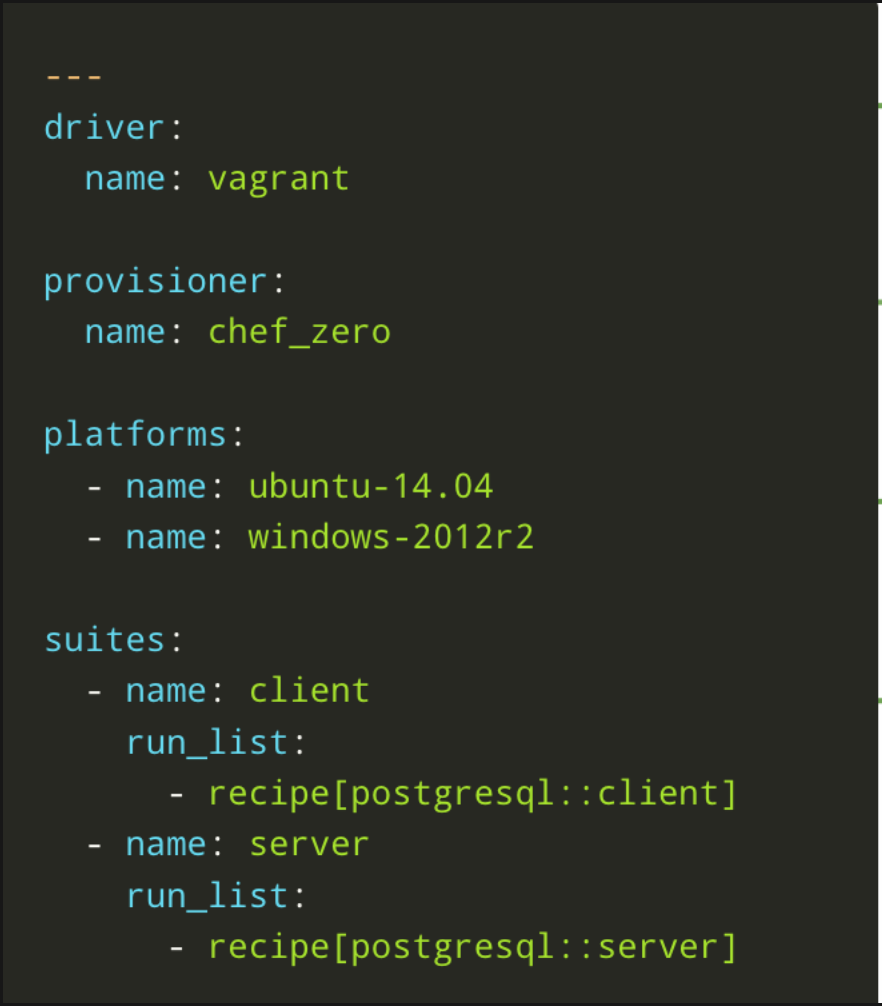
ReaR backup_url

- default['rear']['config']['backup_url']
- if ::File.exist?('/etc/install/config')
  - rear_netfs_url=
  - rear_netfs_url=nfs-server:/path
- include_recipe 'rear::configure'

Kitchen test

- To test the cookbook – kitchen test
- Kitchen provides a test harness to execute infrastructure code on one or more platforms in isolation
- https://docs.chef.io/kitchen.html
- https://kitchen.ci/
- kitchen.yml

Kitchen.yml

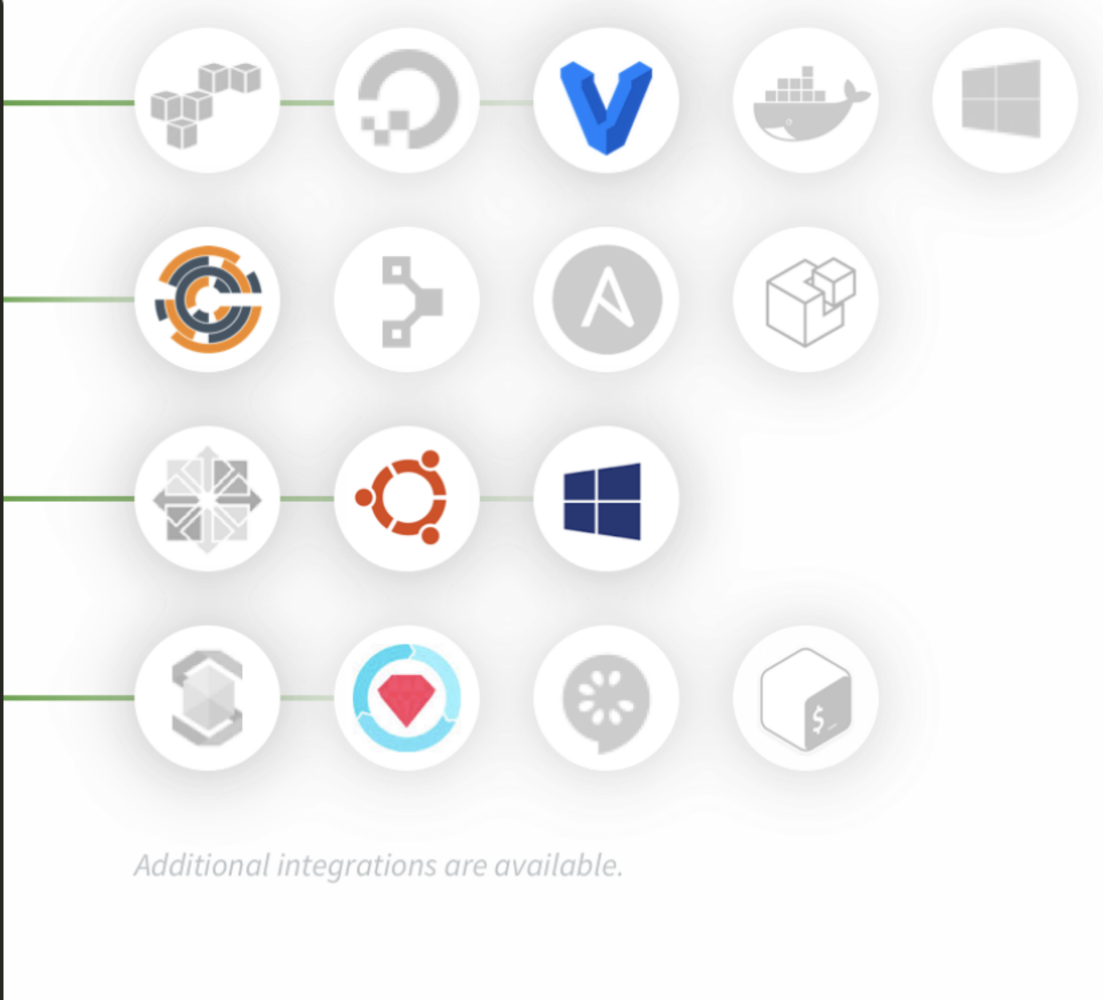


```
---
driver:
  name: vagrant

provisioner:
  name: chef_zero

platforms:
  - name: ubuntu-14.04
  - name: windows-2012r2

suites:
  - name: client
    run_list:
      - recipe[postgresql::client]
  - name: server
    run_list:
      - recipe[postgresql::server]
```



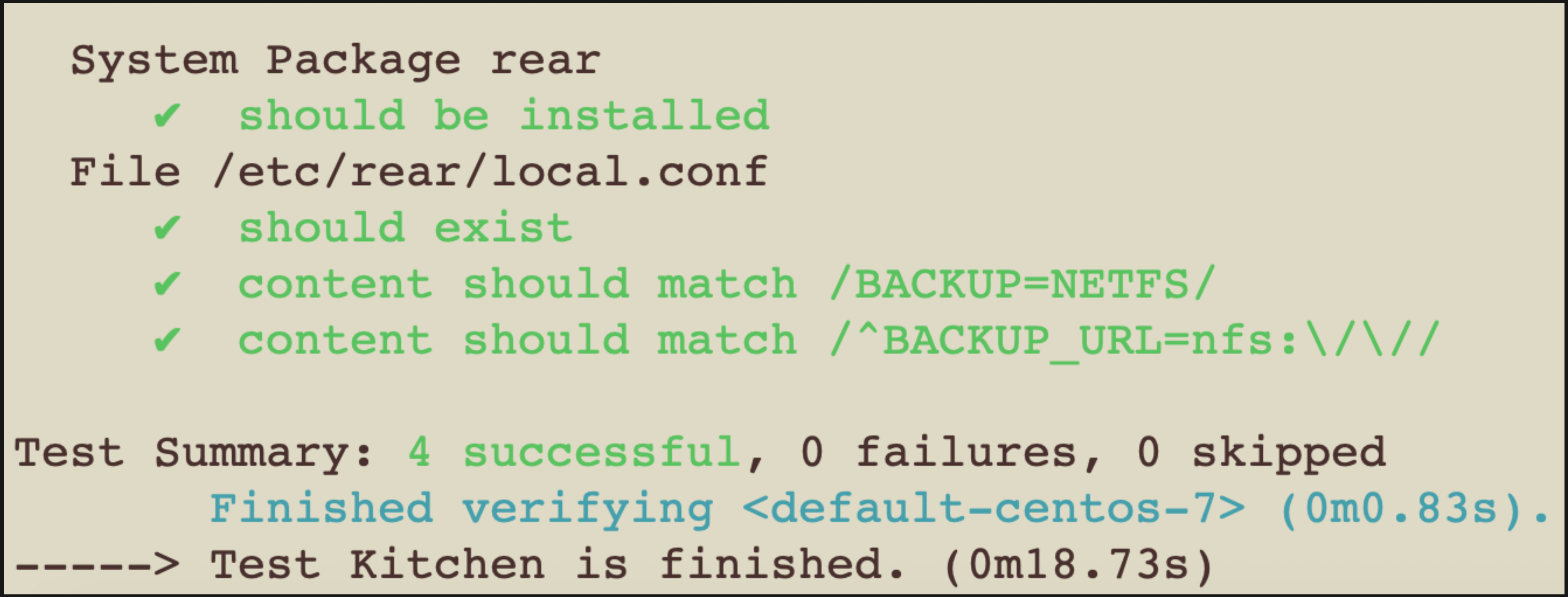
Additional integrations are available.

Time for demo’s

- Run ‘kitchen create’ to launch Ubuntu VM
- Run ‘kitchen converge’ to see some action
- Demo 1 – /etc/install/config contains
  rear_netfs_url=192.168.33.1:/System/Volumes/Data/Users/gdha/exports
- Demo 2 – /etc/install/config contains
  rear_netfs_url=N/A
- Demo 3 – delete /etc/install/config

InSpec test

- Demo 4 – run ‘kitchen verify’



```
  System Package rear
     ✔  should be installed
  File /etc/rear/local.conf
     ✔  should exist
     ✔  content should match /BACKUP=NETFS/
     ✔  content should match /^BACKUP_URL=nfs:\/\//

Test Summary: 4 successful, 0 failures, 0 skipped
       Finished verifying <default-centos-7> (0m0.83s).
-----> Test Kitchen is finished. (0m18.73s)
```