MOVING FAST TOGETHER: TEST DRIVEN INFRASTRUCTURE

VICTORIA D JEFFREY

TWITTER: @VICKKOALA

GITHUB: VJEFFREY

WHO AM I?

- Software Engineer @ Chef
- Over the last six months: focused on various InSpec related projects

SO, WHAT IS THIS LADY TALKING ABOUT???

- what is infrastructure testing? why should we do it?
- some awesome tooling for writing infrastructure tests
- quick demo
- a little more about InSpec and what it can do

WHAT IS INFRASTRUCTURE TESTING????

- testing your IT infrastructure
- traditionally done manually/with one-off scripts

WHAT IS INFRASTRUCTURE TESTING????

- testing your IT infrastructure
- traditionally done manually/with one-off scripts
- config mngmt tools like Chef allow infrastructure to be described as code - makes it easy to introduce tests

WHAT IS INFRASTRUCTURE TESTING????

- testing your IT infrastructure
- traditionally done manually/with one-off scripts
- config mngmt tools like Chef allow infrastructure to be described as code - makes it easy to introduce tests
- so let's treat our infrastructure like code
- test-kitchen, server spec...

WHY IS INFRASTRUCTURE TESTING IMPORTANT?

- what kind of tests do you have for your app now?

WHY IS INFRASTRUCTURE TESTING IMPORTANT?

- what kind of tests do you have for your app now?
- when do you run those tests? after you've deployed?

WHY IS INFRASTRUCTURE TESTING IMPORTANT?

- what kind of tests do you have for your app now?
- when do you run those tests? after you've deployed?
- so what about infrastructure tests?

WHY IS INFRASTRUCTURE TESTING IMPORTANT?

- what kind of tests do you have for your app now?
- when do you run those tests? after you've deployed?
- so what about infrastructure tests?
- why not integrate those into your CI system?

SO, YOU HAVE A CI PIPELINE/BUILD SYSTEM. . . .

....AND YOU'RE RUNNING YOUR UNIT AND INTEGRATION TESTS IN IT...SOMETHING LIKE THIS

Verify

Unit — Passed

Download | Include color in download

```
Starting job on builder 172.21.192.109 at 2016-09-16T18:09:53+00:00.
Command: delivery job verify unit --server delivery.chef.co --user builder --ent chef --org tech-content --project
Chef Delivery
Loading configuration from /var/opt/delivery/workspace
Starting job for chef-web-docs verify unit
Creating workspace in /var/opt/delivery/workspace/delivery.chef.co/chef/tech-content/chef-web-docs/master/verify/u
```

NOW HOW ABOUT RUNNING THOSE INFRA TESTS?

+ Smoke — Skipped

− Functional — Passed

Compliance Report — View Log Report

CIS Ubuntu 14.04 LTS Server Level 1

1.0.0

CIS Ubuntu 14.04 LTS Server Benchmark Level 1 translated from SCAP

Summary: 88 ✔

Check for Duplicate GIDs	Passed
Check for Duplicate Group Names	Passed
Check for Duplicate UIDs	Passed
Check for Duplicate User Names	Passed
Check for Presence of User .forward Files	Passed

DON'T WANNA HEAR YOUR INFRASTRUCTURE IS FAILING AT MIDNIGHT...

THAT MAKES TOWLIE SAD..

RIGHT. SO WE KNOW WHAT WE WANT TO DO WITH OUR INFRA TESTS. . . .

BUT HOW DO WE ACTUALLY WRITE INFRASTRUCTURE TESTS??

RIGHT. SO WE KNOW WHAT WE WANT TO DO WITH OUR INFRA TESTS. . . .

BUT HOW DO WE ACTUALLY WRITE INFRASTRUCTURE TESTS??

WHAT IS INSPEC??

- open-source framework for testing your infrastructure

WHAT IS INSPEC??

- open-source framework for testing your infrastructure
- command-line tool

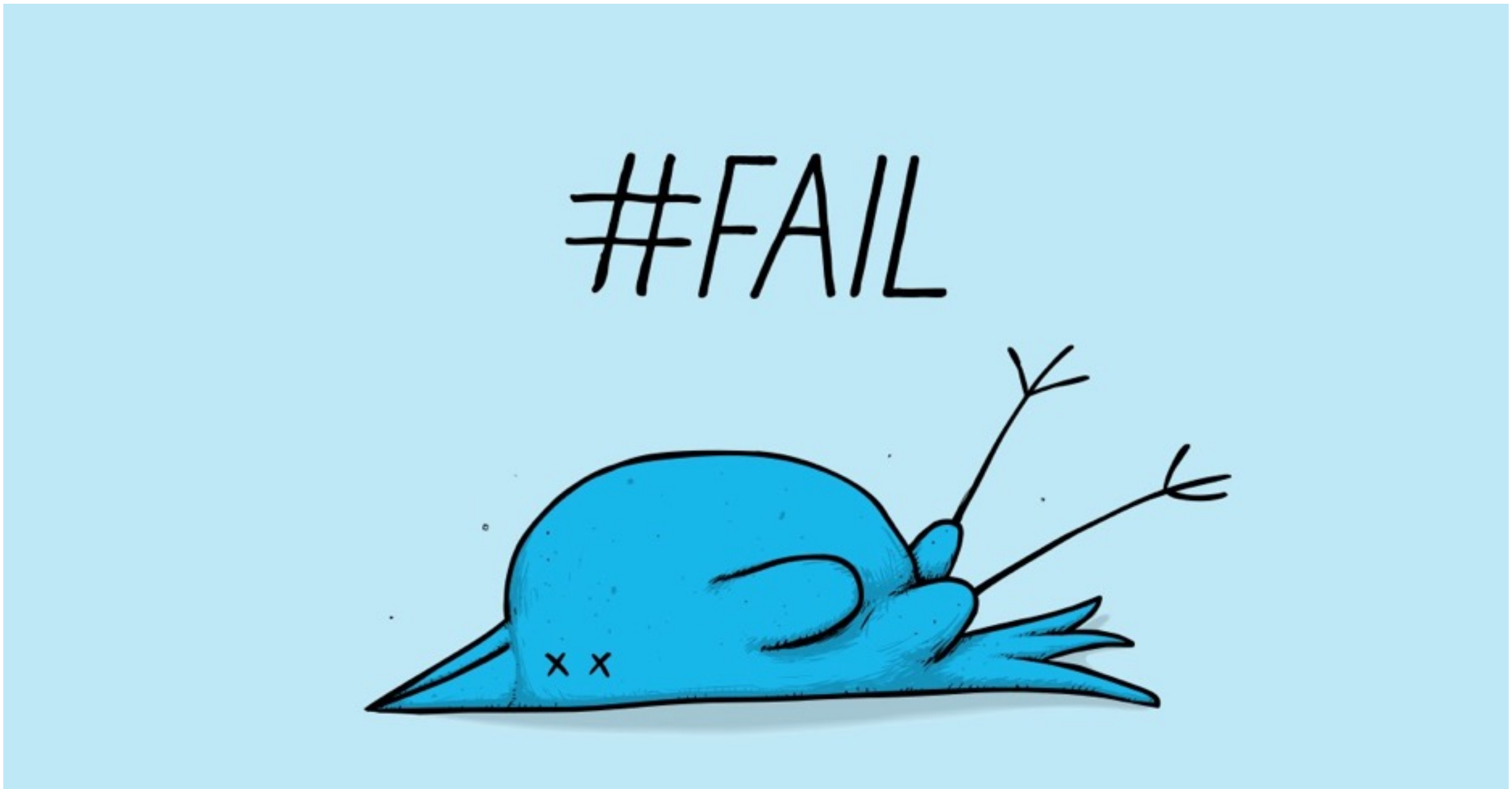
WHAT IS INSPEC??

- open-source framework for testing your infrastructure
- command-line tool
- platform agnostic: you’re only limited to what you want to do

WHAT IS INSPEC??

- open-source framework for testing your infrastructure
- command-line tool
- platform agnostic: you're only limited to what you want to do (CentOS, Debian, Oracle, Solaris, Windows, FreeBSD, AIX, RedHat, SUSE, HP-UX)
- run InSpec locally to test against any target — InSpec doesn't care if you use Chef, Ansible, Puppet - whatever.
- test your infrastructure via ssh, winrm, docker

OpenSSL


#FAIL

NOW LET'S USE INSPEC....

```
describe ssl(port: 443).protocols('ssl2') do
  it { should_not be_enabled }
end
```

NICE, HUH?

SO HOW DO WE USE INSPEC??

IT'S A GEM!

YA, COOL, K. HOW DO I RUN IT??

$ inspec exec test.rb

```
Target:  local://


  SSL/TLS on
     ✔  localhost:443 with protocol == "ssl2" should not be enabled

Test Summary: 1 successful, 0 failures, 0 skipped
```

SO, THAT'S COOL, BUT…MY PROD ENV != MY LAPTOP

- well, did you see that little target thing in the output??

We can run the tests against other targets…

```
# run test locally
inspec exec test.rb

# run test on remote host on SSH
inspec exec test.rb -t ssh://user@hostname -i /path/to/key

# run test on remote windows host on WinRM
inspec exec test.rb -t winrm://Administrator@windowshost --password 'your-password'

# run test on docker container
inspec exec test.rb -t docker://container_id
```

D-D-DEMO TIME!!!!

to the terminal we go!

D-D-DEMO TIME!!!!

example profile:

https://github.com/dev-sec/tests-os-hardening

secured configuration for server:

https://github.com/dev-sec/chef-os-hardening

BUT WAIT, WE'VE SEEN THESE KINDS OF DESCRIBE BLOCKS BEFORE...

```
describe file('/etc/sysconfig/init') do
  its('content') { should match 'umask 027' }
end
```

"the describe block"

SO WHAT'S SO SPECIAL ABOUT INSPEC??

MAKE IT EASY TO UNDERSTAND

```
control ‘cis-3.1’ do
  impact 0.7
  title ‘Set Daemon umask’
  desc ‘Set the default umask for all processes started at boot time ’
  describe file(‘/etc/sysconfig/init’) do
    its(‘content’) { should match ‘umask 027’ }
  end
end
```

But Wait...
THERE'S
MORE!!!

SO MUCH GOOODNESSSSS...

$ INSPEC HELP

```
Commands:
  inspec archive PATH                 # archive a profile to tar.gz (default) or zip
  inspec check PATH                   # verify all tests at the specified PATH
  inspec compliance SUBCOMMAND ...    # Chef Compliance commands
  inspec detect                       # detect the target OS
  inspec env                          # Output shell-appropriate completion configuration
  inspec exec PATHS                   # run all test files at the specified PATH.
  inspec help [COMMAND]               # Describe available commands or one specific command
  inspec init TEMPLATE ...            # Scaffolds a new project
  inspec json PATH                    # read all tests in PATH and generate a JSON summary
  inspec msccm SUBCOMMAND ...         # Microsoft Systems Center Configuration Manager commands
  inspec scap SUBCOMMAND ...          # SCAP commands
  inspec shell                        # open an interactive debugging shell
  inspec supermarket SUBCOMMAND ...   # Supermarket commands
  inspec vendor                       # Download all dependencies and generate a lockfile
  inspec version                      # prints the version of this tool

Options:
  l, [--log-level=LOG_LEVEL]          # Set the log level: info (default), debug, warn, error
      [--log-location=LOG_LOCATION]   # Location to send diagnostic log messages to. (default: STDOUT or STDERR)
      [--diagnose], [--no-diagnose]   # Show diagnostics (versions, configurations)
```

INSPEC DETECT

$ INSPEC DETECT -T SSH://VAGRANT@192.168.34.102 -I .VAGRANT/MACHINES/UBUNTU1204/VIRTUALBOX/PRIVATE_KEY

```
== Operating System Details

Name:       ubuntu
Family:     debian
Release:    12.04
Arch:       x86_64
```

INSPEC SHELL: THE INTERACTIVE DEBUGGING SHELL

```
inspec> command('uname -a').stdout
=> "Darwin vjeffrey01 15.6.0 Darwin Kernel Version 15.6.0: Mon Aug 29 20:21:34 PDT 2016; root:xn
u-3248.60.11~1/RELEASE_X86_64 x86_64\n"
inspec> sshd_config.Protocol
=> "2"
inspec> help resources
apache apache_conf apt ppa audit_policy auditd_conf auditd_rules command bash file bond bridge d
irectory etc_group gem group grub_conf host iis_site inetd_conf interface iptables json kernel_m
odule kernel_parameter linux_kernel_parameter limits_conf login_defs mount mssql_session mysql m
ysql_conf mysql_session npm ntp_conf oneget os os_env package parse_config parse_config_file pas
swd pip port postgres postgres_conf postgres_session powershell script processes registry_key wi
ndows_registry_key security_policy service systemd_service upstart_service sysv_service bsd_serv
ice launchd_service runit_service shadow ssl ssh_config sshd_config sys_info users user vbscript
 windows_feature xinetd_conf wmi yum yumrepo yaml csv ini
inspec>
```

YOU JUST NEED TO CHECK IF ONE OF THOSE TESTS PASS?
NO PROBLEM! WE GOT YOU COVERED!!
DESCRIBE.ONE

```
control 'or-test' do
  impact 1.0
  title 'This is a OR test'
  describe.one do
    describe ssh_config do
      its('Protocol') { should eq('3') }
    end
    describe ssh_config do
      its('Protocol') { should eq('2') }
    end
  end
end
```

But wait!
There's more!

THE PROFILE. . . .

- what is a profile, inspec init, inspec check

```
➜  ~ inspec init profile testing
WARN: Unresolved specs during Gem::Specification.reset:
      net-ssh (< 4.0, >= 2.6.5, >= 2.9)
      winrm (~> 2.0)
      logging (< 3.0, >= 1.6.1)
      docker-api (~> 1.26)
WARN: Clearing out unresolved specs.
Please report a bug if this causes problems.
Create new profile at /Users/vjeffrey/testing
 * Create directory controls
 * Create file controls/example.rb
 * Create file inspec.yml
 * Create directory libraries
 * Create file README.md
 * Create file libraries/.gitkeep
```

$ inspec check testing

```
Profile:     testing
Controls:    2
Timestamp:   2016-09-22T23:46:27-04:00
Valid:       true
```

LET'S GET FANCY WITH THOSE PROFILES...

- inspec exec examples/inheritance

USE INSPEC WITH TEST-KITCHEN!!

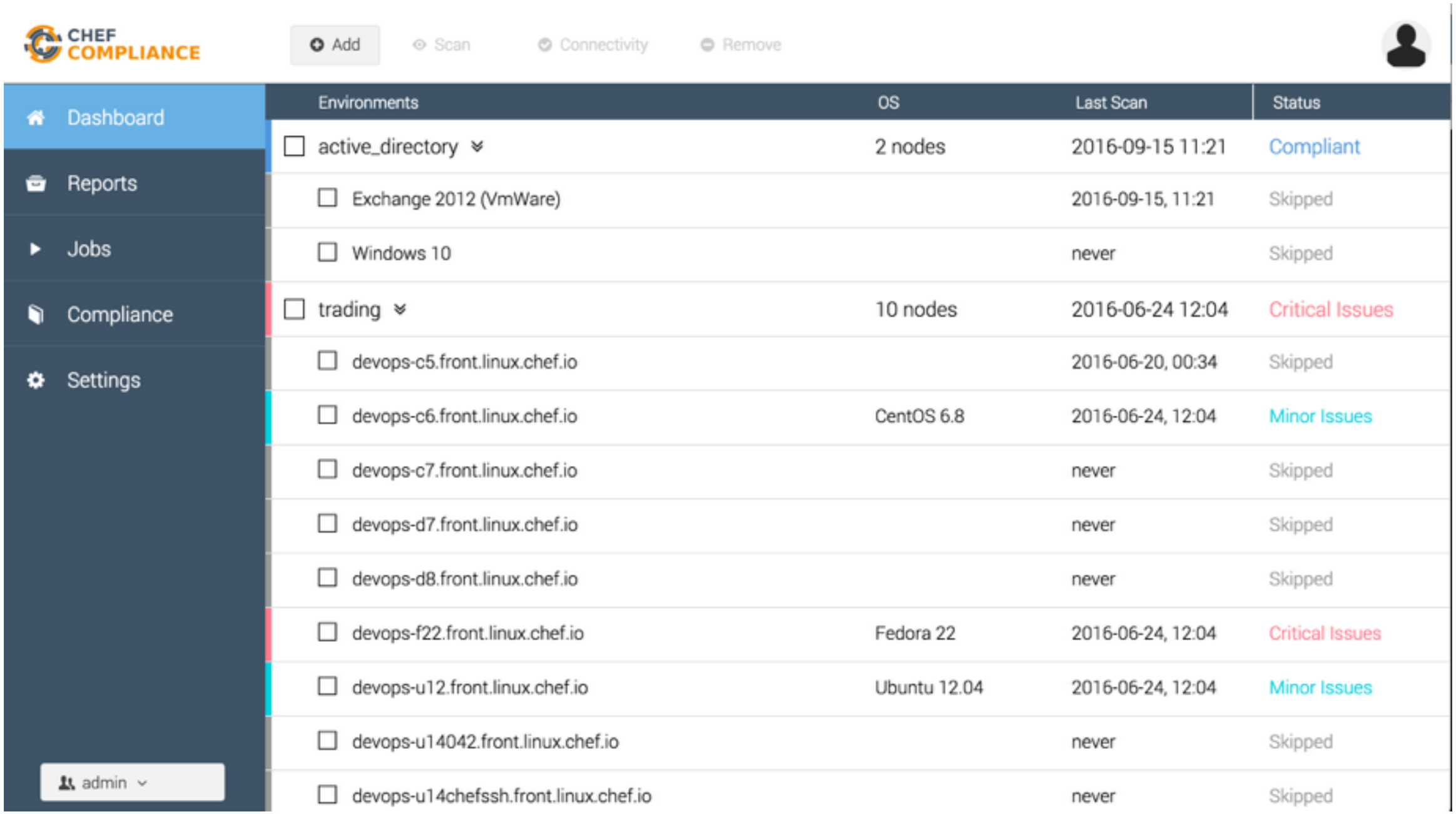
- add inspec to your .kitchen.yml as a verifier option!

```
verifier:
  name: inspec
  sudo: false
  format: json
```

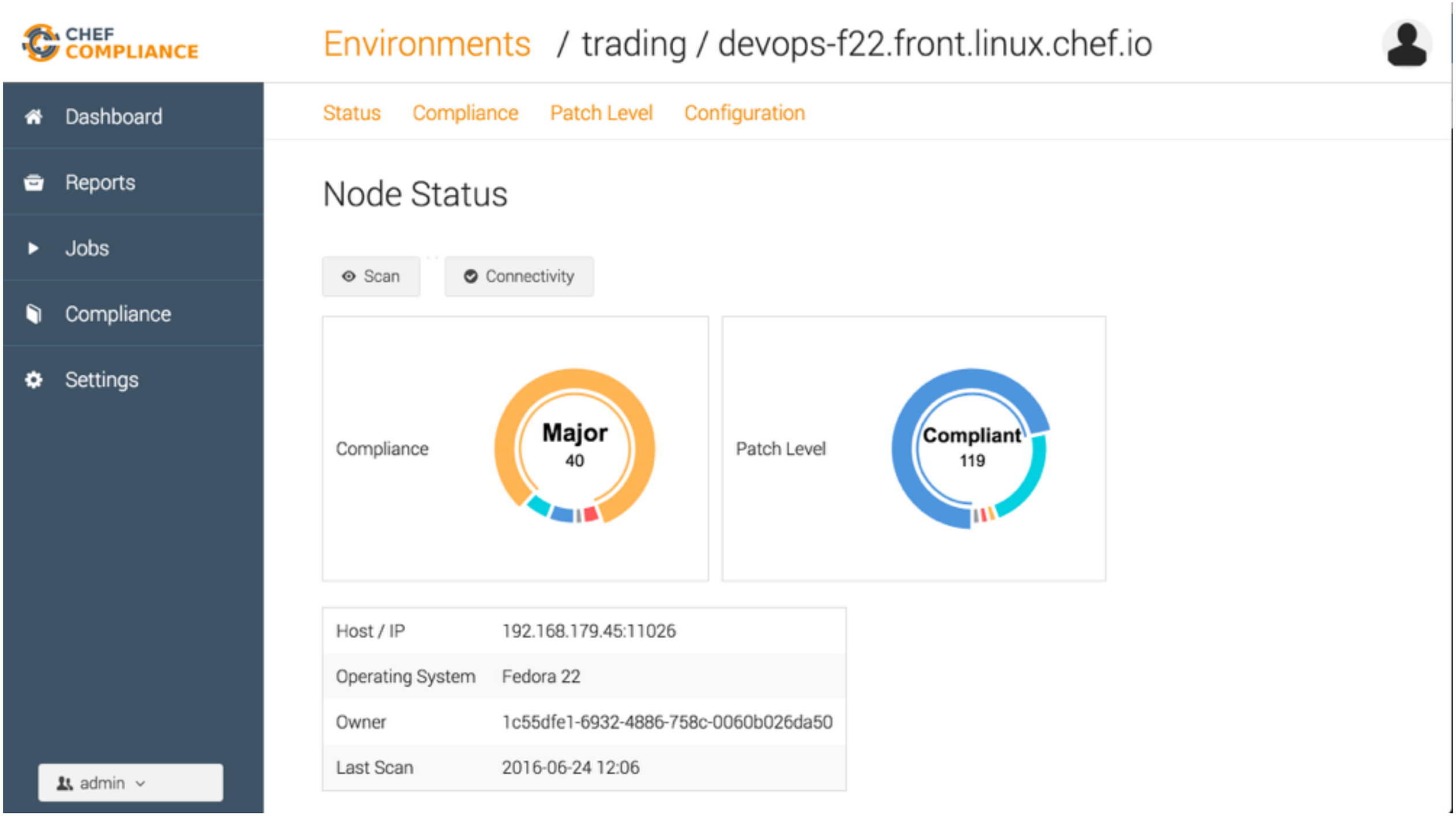
CHECK THIS THANG OUT >>>>>

- inspec exec examples/profile -- format json

COMPLIANCE SERVER ($$$)



COMPLIANCE SERVER ($$$)



WANNA EXPLORE SOME MORE??

- https://github.com/chef/inspec
- tutorial!!!
- inspec.io coming soon (monday, 9/26), includes interactive tutorial

THE END

VICTORIA D JEFFREY

TWITTER: @VICKKOALA

GITHUB: VJEFFREY

WHAT QUESTIONS CAN I ANSWER FOR YOU?

VICTORIA D JEFFREY

TWITTER: @VICKKOALA

GITHUB: VJEFFREY