Essential application management with Tiny Puppet

Presented by Alessandro Franceschi @alvagante
Founder and CTO example42 GmbH
at #cfgmgmtcamp 2016

not only files...

- CfgMgmt is not only about Files
- We (try to) manage the entire infrastructure
- There are modules for everything

but **also** files

- Still sometimes we just have to manage packages, services and files

- And we do quick local modules, reinventing simple wheels

- Or we search for public ones, sometimes overkill for our needs, or not compatible, or need fixes

- Different modules, different usage patterns, conventions, dependencies

meet Tiny Puppet

- A Puppet module
  `puppet module install example42/tp`

- It let us manage packages, services, and files
  in the **way we want**

- It supports potentially **any application**
  on **any operating systems**

- It let us concentrate on
  how to shape our configurations

tp usage: install an application

- Install an application and start its service

```
tp::install { 'nginx': }
```

- Tiny Puppet takes care of dependencies and resources' names on different Operating Systems

tp usage: manage files

- Manage the **main** configuration file with a erb template based on a custom options hash

```
tp::conf { 'nginx':
  template     => 'site/nginx/nginx.conf.erb',
  options_hash => hiera_hash('nginx_options',
}
```

tp usage: manage files, more

- Manage a virtual host , this time using an epp template and a different base dir

```
tp::conf { 'nginx::example.com.conf':
  base_dir     => 'confd',
  epp_template => 'site/nginx/vhost.epp',
  options_hash => $vhost_params,
}
```

tp usage: manage dirs

- Manage entire directories, from fileserver, vcs repos...

```
tp::dir { 'nginx::www.example42.com:
  base_dir => 'data',
  source   => 'git@example42.com/site/',
  vcsrepo  => 'git',
}
```

Essential testing

- To be able to test an app (alternatives)

```
tp::test { 'redis': }

tp::install { 'redis':
  test_enable   => true,
  test_template => 'site/test/nginx.erb',
}
```

- This creates the script `/etc/tp/test/redis` which can be used for integration tests, monitoring...

Essential acceptance tests

- TP be able to test an app (alternatives)

```
git clone https://github.com/example42/tp-acceptance
cd tp-acceptance
r10k puppetfile install
vagrant status

# Test munin on all supported OS
bin/test_app.sh munin all acceptance

# Test all apps on a specific OS
bin/test_app.sh all Centos7 acceptance

# Test all appas on all supporte OS
bin/test_app.sh all all acceptance
```

Compatibility Matrix

It's all about Tiny Data

- Data to manage apps on different OS is on separated tinydata module

- Here we have a directory for each supported application where the relevant data, for different OS, is stored in YAML files with an Hiera-like approach:

```
---
  redis::settings:
    package_name: 'redis'
    service_name: 'redis'
    config_file_path: '/etc/redis/redis.conf'
    config_dir_path: '/etc/redis'
    tcp_port: '6379'
    pid_file_path: '/var/run/redis.pid'
    log_file_path: '/var/log/redis.log'
    log_dir_path: ''
    process_user: 'redis'
    process_group: 'redis'
```

Usage cases

- When we have to configure applications that **we know how to configure**

- In glue code, profiles, wrappers...

- Tiny Puppet can **coexist** with component modules and replace or complement them (stdlib is the only dependency)

When is better a dedicated module

- When application setup is elaborated or complex

- When there is more to manage than just packages, services and files

- When it hides the complexity of the application configuration

Future steps

- New defines based on the same Tiny Data

```
tp::instance { 'memcached': }

tp::monitor { 'redis': }

tp::firewall { 'redis': }

tp::netinstall { 'rails': }
```

- Use Tiny data in other tools (tiny-ansible, tiny-chef...?)

Where

- Tiny Puppet website: http://www.tiny-puppet.com
- Tp module: https://github.com/example42/puppet-tp/
- Tiny Data module: https://github.com/example42/tinydata/
- TP Playground: https://github.com/example42/tp-playground
- TP Acceptance tests: https://github.com/example42/tp-acceptance
- Puppet Forge example42/tp