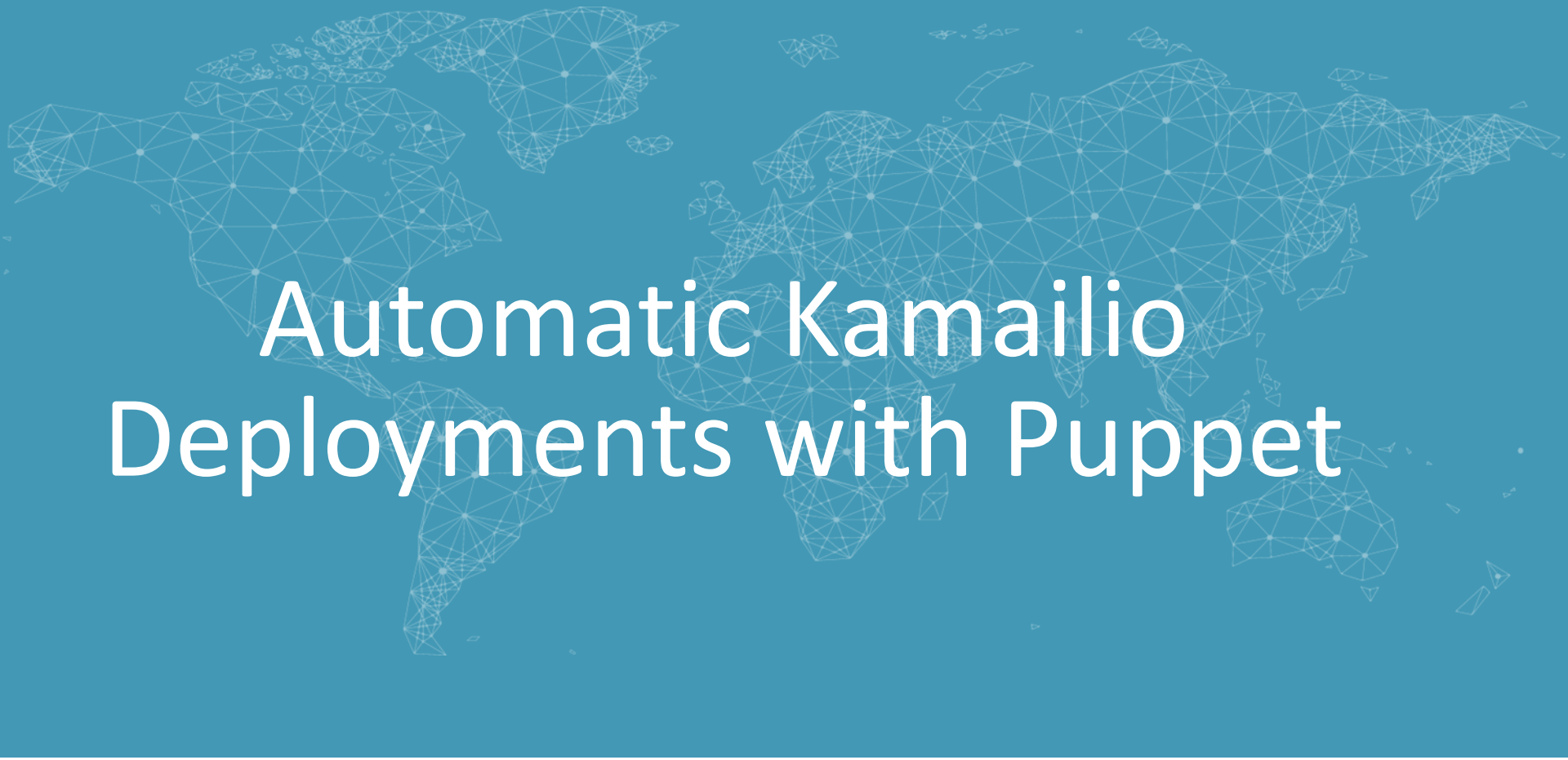


Automatic Kamailio Deployments with Puppet

Giacomo Vacca

Senior Network Applications Developer

truphone

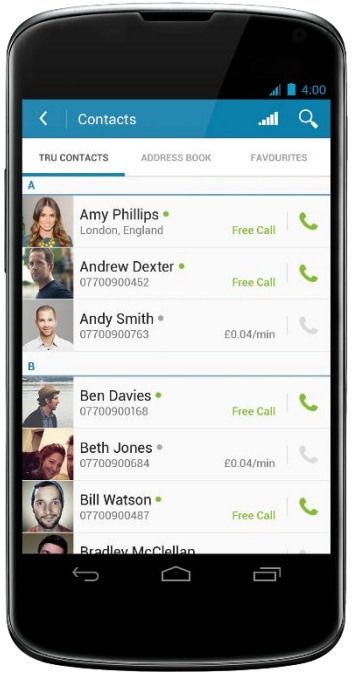
@giavac

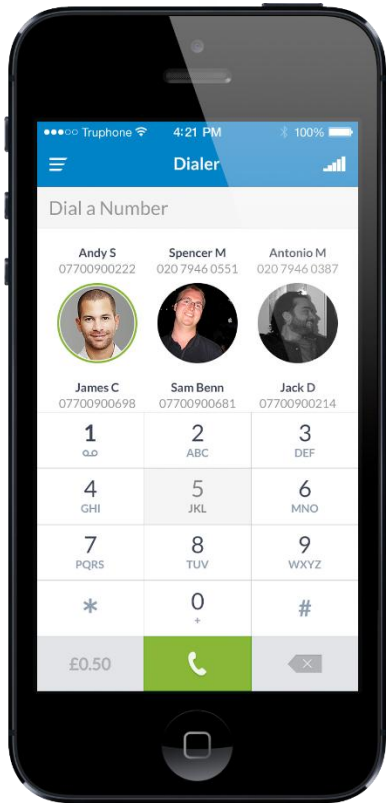
“He’s seen more SIP transactions than some SBCs in the market” (cit.)

- Leads Network Dev Team at Truphone Labs
- Hacking Kamailio, Asterisk, FreeSWITCH for 10+ ys
- Hacking ejabberd, more recently

Truphone

labs.truphone.com





The Problem

"You know you're in trouble when...

you realize you've become good at 3-way diffs"

The Problem – in the past

- Deploy and customize in-place
- Extract variables into included defines file
- A debian package with config per host

No (automatic) documentation!

The Solution - Puppet

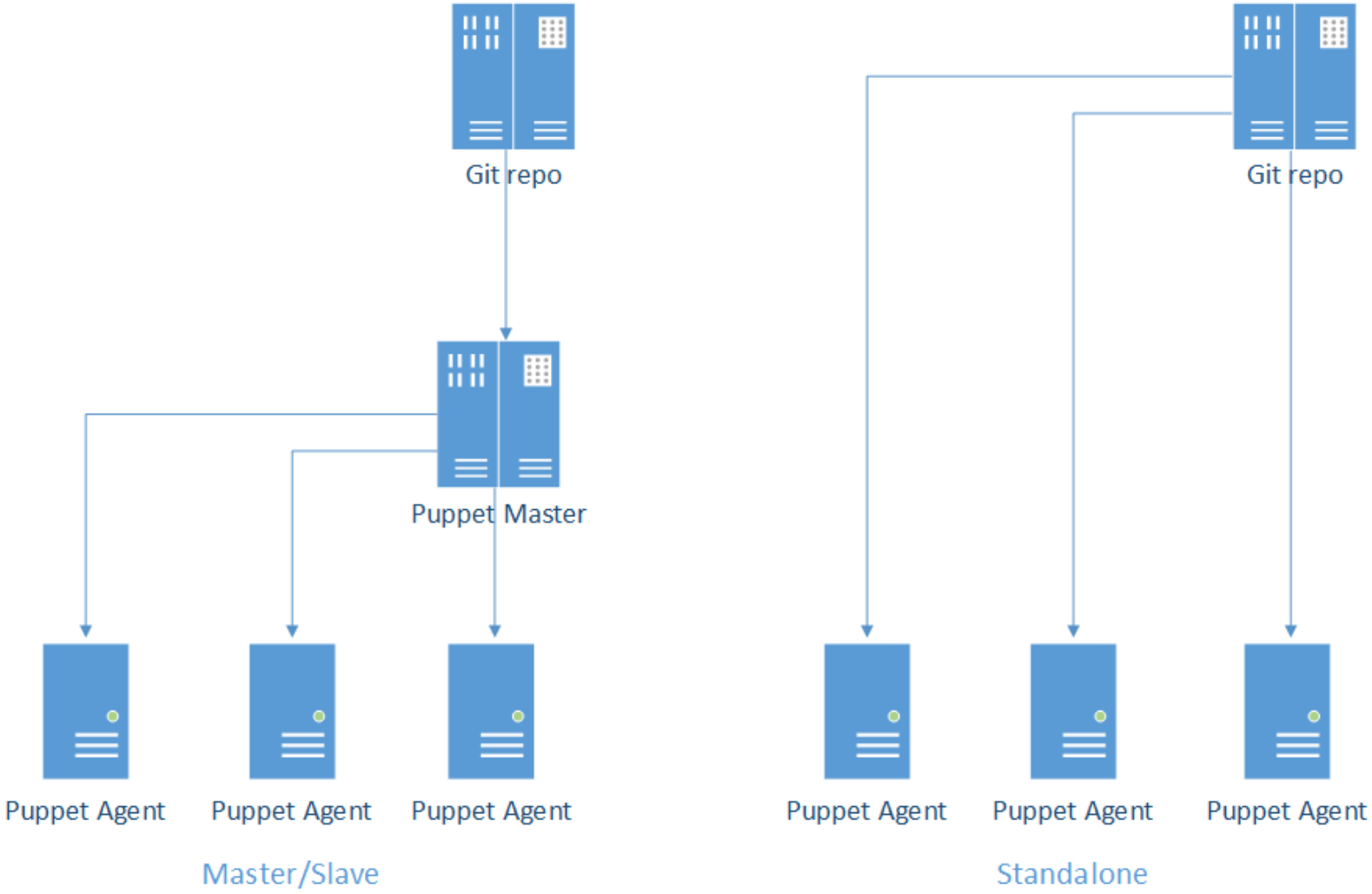
- Open Source configuration management tool
- Defines the final status (‘what’, **not** ‘how’)
- Idempotent (i.e. safely run multiple times)

puppetlabs.com

The Solution - Puppet

- Puppet code is contained in MANIFESTS
- Puppet functionalities are organized in MODULES
- “Compiled” manifests are CATALOGUES

Master/Slave vs Standalone



Puppet
Environments, Sites, Nodes

```
# nodes.pp
node ‘k01.domain.com’ {
  include ‘kamailio’
}
```

- As many environments as you want
  - Each environment defines a Site
    - A Site defines a group of Nodes
      - Every host is a Node

Puppet
Separate logic from data

- Simple manifests
- Independent from data
- Extract data into nodes definition

Kamailio with Puppet

https://github.com/trulabs/puppet-kamailio

1. Deploy pre-requisites
2. Install packages (deb, rpm)
3. Set configuration files (kamailio.cfg, tls.cfg)
4. Define ‘service’ (“should be running”)

Kamailio with Puppet
The system around Kamailio

Other areas (see 3rd party modules at PuppetForge):

- apt/yum
- TCP keepalives
- Firewall (iptables)
- monit
- nagios
- fail2ban
- logrotate
- sec
- SSL certificates
- Swappiness

» What else? →Tell me at the Q&A session.

Useful tools:

Check your modules against Best Practices with 'puppet-lint'.

http://puppet-lint.com/

Build and deploy a new instance

Video: http://vimeo.com/giavac/akdp

1. Build VM
2. SSH into VM
3. git clone Puppet repo
4. Run Puppet NOOP mode
5. Run Puppet

Summary

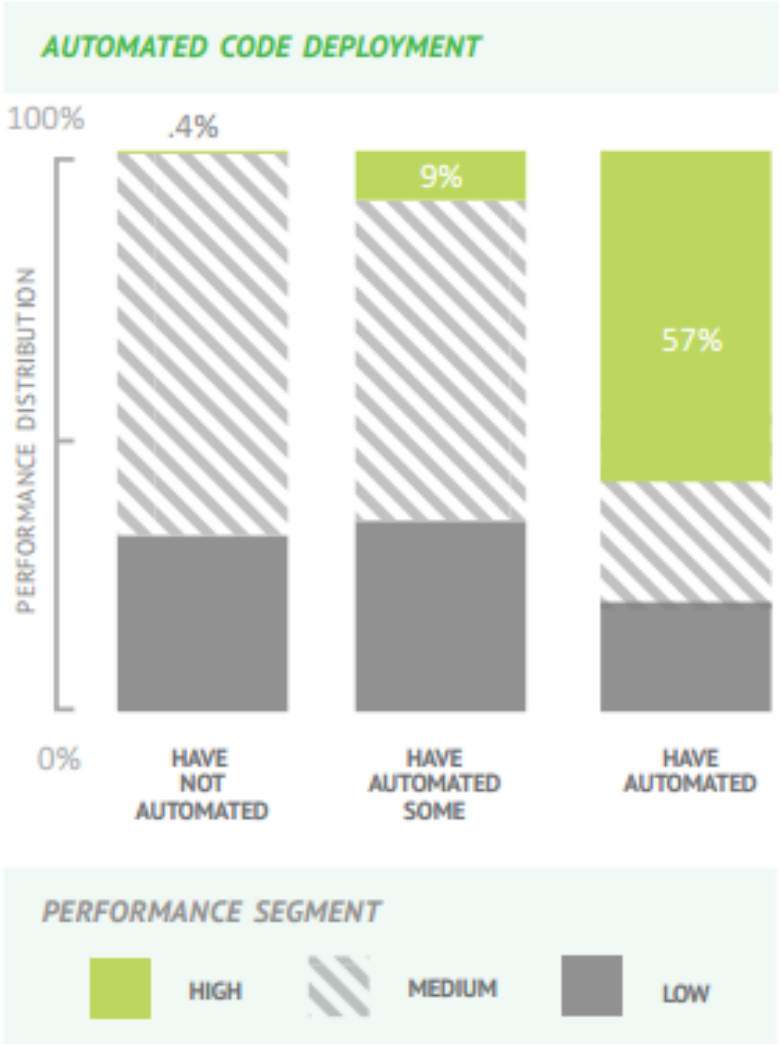
Automation is easy

Deployments MUST be automated

Deployments SHOULD be boring

Deployments MAY be fun

Puppet does the job



(Diagram source: http://info.puppetlabs.com/2013-state-of-devops-report.html)

Q&A

@giavac

labs@truphone.com *

https://labs.truphone.com/about/

ADDITIONAL SLIDES

Kamailio with Puppet
Kamailio module – repo/apt.pp

```
class kamailio::repo::apt inherits kamailio::repo {
  include '::apt'

  # Note the 40!
  apt::source { 'kamailio40_wheezy':
      location                => 'http://deb.kamailio.org/kamailio40',
      release                 => 'wheezy',
      repos                   => 'main',
      required_packages => 'debian-keyring debian-archive-keyring',
      key                     => '07D5C01D',
      key_server              => 'http://deb.kamailio.org/kamailiodebkey.gpg',
      include_src             => true,
  }

  Apt::Source['kamailio40_wheezy'] -> Package<|tag == 'kamailio'|>
}
```

Kamailio with Puppet
Kamailio module – config.pp

- Manages configuration files
- Can install from files or templates

Example: **templatize** a kamailio.cfg define

```
<% if with_tls %>#!define WITH_TLS<% end %>
```