(Almost) Instant monitoring

Ansible deploying Nagios+PMP

Daniel Guzman Burgos (Percona)

2015-04-14

Agenda

- Monitoring and Nagios quick review
- Percona Nagios Plugins
- Ansible Insights
- Vagrant in 120 seconds
- Monitoring deployment: Auto

Introductions

- Daniel Guzmán Burgos
  - daniel.guzman.burgos(at)percona.com
  - Consultant at Percona

Link to the Repo

- https://github.com/nethalo/instant-monitoring
  - Homepage for this session
  - git clone <above_link>
- plays/
  - All the playbooks that lead to happiness

Before we begin...

- If you get lost or cannot see something:
  - check out the repo!
    - *detailed instructions*
    - *complete files*

Nagios Review

Two major problems that are issued by monitoring:

- Alerting
- Trending

Nagios Review

- Alerting: Notify a responsible person about some major event, example: Replication broken
- Trending: Track the change of something over time, example: Replication lag

Nagios Review

What to monitor?

How to know which services are more important than others?

Nagios Review

It depends.

...but is very related to the SLA (and out of scope of this session)

Percona Nagios Plugins

- Group of scripts created by MySQL experts
- With good documentation,
- Support for the newest versions of MySQL and InnoDB
- Great integration with other Percona software, such as Percona Server and Percona Toolkit.

Percona Nagios Plugins

And the best: Easy to install and configure!

http://www.percona.com/doc/percona-monitoring-plugins/1.1/nagios/index

Ansible Insights - About

- Automation tool
- Written in python
- Agentless (plain SSH or python)
- Idempotent
- Easy to learn
- Relatively new (2012)
- Supports *NIX primarily (Windows: >1.7)

Ansible Insights - Terminology

- Management Workstation
  - *NIX machine
  - Some extra requirements
- Managed Node
  - Where the magic happens!

Ansible Insights - Terminology

- Inventory
  - Definition of host groups
  - Common settings for hosts
  - Can be extended and/or dynamically generated

Ansible Insights - Terminology

- Playbook
  - Top level “plan”
  - Tasks that run against a group of hosts
  - Sort of instructions

Ansible Insights - Terminology

- Tasks
  - The actual steps that execute
  - Execute sequentially
  - Idempotent
  - Can use “facts” to make smart decisions
  - Leverage modules to get the job done

Ansible Insights - Terminology

- Modules
  - Basic building blocks of Ansible
  - Execute actions
  - Programmable
  - Most likely to be written in Python

Ansible Insights - Requirements

- SSH
  - OpenSSH or Paramiko
  - Access, permissions
    - Deploy user vs. Operating user

Ansible Insights - Requirements

- Python
  - Already installed most of the time (LSB)
  - Management Workstation (>2.6)
  - Managed Hosts (>2.4)
  - python-simplejson (python 2.4)

Vagrant

- Software for creating and configuring virtual development environments.
- Can be seen as a wrapper around virtualization software such as VirtualBox

Vagrantfile

```
# -*- mode: ruby -*-
# vi: set ft=ruby :

Vagrant.configure(2) do |config|

  config.vm.box = "ubuntu/trusty64"
  config.vm.box_check_update = false

  config.vm.define "uno" do |uno|
    uno.vm.hostname = "nodo-uno"
    uno.vm.network "private_network", ip: "172.28.128.3"
  end

  config.vm.define "nagioscentral" do |nagioscentral|
     nagioscentral.vm.hostname = "nagios-central"
     nagioscentral.vm.network "forwarded_port", guest: 80, host: 1234
     nagioscentral.vm.network "private_network", ip: "172.28.128.7"
  end

  config.vm.provider "virtualbox" do |vb|
     vb.memory = "512"
  end
end
```

Start up VMs

- Start all VMs
  - vagrant up
- Check status of VMs
  - vagrant status

(Almost) Instant monitoring

- git clone http://bit.ly/1DVxiHo
- cd instant-monitoring/
- vagrant up
- ansible-playbook plays/set-keys.yml --ask-pass -e "username=XXXX"
- ansible-playbook site.yml

What is inside?

Templates

```
{% for host in groups['small-nodes'] %}
  define host{
        use                     generic-percona-host
        host_name               {{ hostvars[host]['ansible_hostname'] }}
        alias                   {{ hostvars[host]['ansible_eth1']['ipv4']['address'] }}
        address                 {{ hostvars[host]['ansible_eth1']['ipv4']['address'] }}
}
{% endfor %}

define hostgroup {
        hostgroup_name              mysql-servers
        alias                       mysql-servers
        members       {% set comma = joiner(",") %}{%- for host in groups['small-nodes'] -%}
        {{ comma() }}{{ hostvars[host]['ansible_hostname'] }}
        {%- endfor %}

}
```

What is inside?

Playbooks

```
---
- hosts: small-nodes
  sudo: yes
  tasks:
  - name: Add Percona Repo Key
    apt_key: keyserver=keys.gnupg.net id=1C4CBDCDCD2EFD2A state=present
  - name: Add Percona Repo Source
    apt_repository: repo='{{ item }}' state=present
    with_items:
      - "deb http://repo.percona.com/apt trusty main"
      - "deb-src http://repo.percona.com/apt trusty main"
```

Questions?

- ???
- *If there's no time, i can answer questions via email*