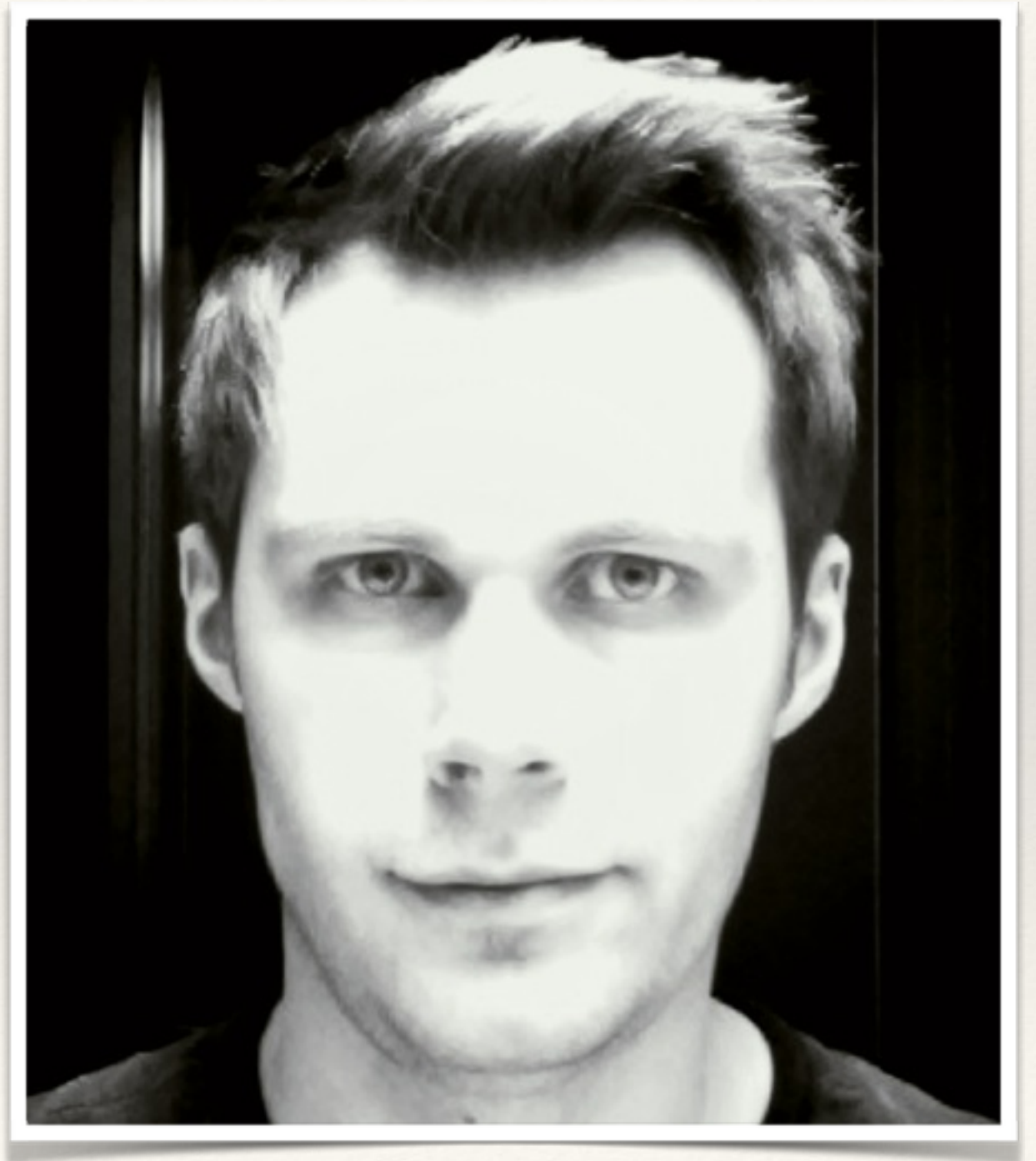




ANSIBLE

Who I am?

- ❖ Robert Reiz
- ❖ Software Developer
- ❖ I started VersionEye
- ❖ I'm NOT your IT guy!



2012 VersionEye is running on Heroku.

2013 VersionEye moves to Yunion because of performance reasons.

2 XXL Servers!
Lovely hand crafted.

2013 VersionEye moves to AWS because
of the Amazon Activate Program!

VersionEye 2014



Handcrafted Servers are

- ❖ **hard to maintain**
- ❖ **very time/cost intensive**
- ❖ **setup is not easily reproducible**
- ❖ **many times very buggy**

Options to automate IT

Chef

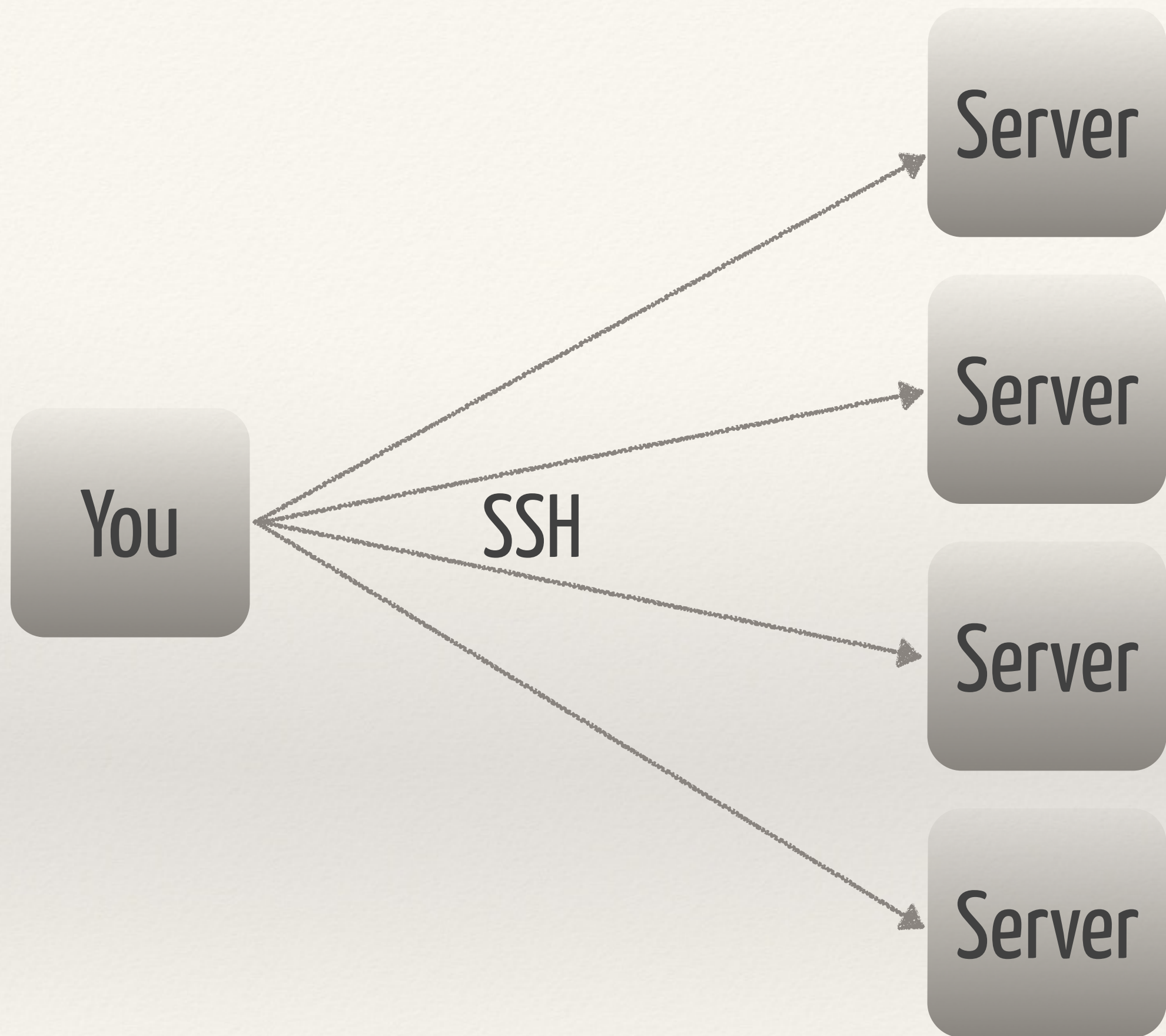
Puppet

Ansible

Salt

Reasons for Ansible

- ❖ No Master
- ❖ No Agents
- ❖ Configuration in Yaml
- ❖ Very easy to learn



Ansible works via SSH. No Master Server! No Agent on the Server is required.

Installation


```
sudo pip install ansible
```

brew update
brew install ansible

Ansible Concepts

- ❖ Inventory
- ❖ Playbooks
 - ❖ Roles
 - ❖ Tasks / Handlers / Vars
 - ❖ Modules

Inventory

Inventory

[mongo_master]
168.197.1.14

[mongo_slaves]
168.197.1.15
168.197.1.16
168.197.1.17

[www]
168.197.1.2

Inventory files are simple text files which describe your servers. IP Addresses or DNS Names grouped by names.

Inventory

[mongo_master]
168.197.1.14

[mongo_slaves]
mongo1.server
mongo2.server
mongo3.server

[www]
168.197.1.2

List of target hosts.
Usually located in
/etc/ansible/hosts

Inventory

[mongo_master]
mongo-[a:c]-server

[mongo_slaves]
mongo[1:3].server

[www]
{{my_little_webserver}}

Inventory files can take advantage of
variables and enumerations

Playbooks

Simple Playbook

```
- hosts: dev_servers
  user: ubuntu
  sudo: true
  roles:
    - java
    - memcached
```

```
- hosts: www_servers
```

group name from the inventory file

```
  user: ubuntu
  sudo: true
```

server auth

```
  roles:
```

```
    - nginx
```

Role which should be installed on the server

Roles / Modules

apt module

- name: update debian packages

apt: update_cache=true

- name: install Java JDK

apt: name=openjdk-7-jdk state=present

apt module

- name: update debian packages
apt: update_cache=true

- **name: install Java JDK** Documentation of this step!

apt: name=openjdk-7-jdk state=present Parameters of the Module

Module

- name: update debian packages
apt: update_cache=true
- name: upgrade packages
apt: upgrade=full
- name: ensure that basic packages are installed
apt: name={{ item }} state=present
with_items:
 - tree
 - wget
 - links2
 - gcc
 - g++
 - make
 - autoconf
 - automake
 - libssl-dev
 - libcurl4-openssl-dev

- [docker](#) - manage docker containers
- [docker_image](#) - manage docker images
- [easy_install](#) - Installs Python libraries
- [ec2](#) - create, terminate, start or stop an instance in ec2, return instanceid
- [ec2_ami](#) - create or destroy an image in ec2, return imageid
- [ec2_ami_search](#) - Retrieve AWS AMI for a given operating system.
- [ec2_asg](#) - Create or delete AWS Autoscaling Groups
- [ec2_eip](#) - associate an EC2 elastic IP with an instance.
- [ec2_elb](#) - De-registers or registers instances from EC2 ELBs
- [ec2_elb_lb](#) - Creates or destroys Amazon ELB.
- [ec2_facts](#) - Gathers facts about remote hosts within ec2 (aws)
- [ec2_group](#) - maintain an ec2 VPC security group.
- [ec2_key](#) - maintain an ec2 key pair.
- [ec2_lc](#) - Create or delete AWS Autoscaling Launch Configurations
- [ec2_metric_alarm](#) - Create/update or delete AWS Cloudwatch 'metric alarms'
- [ec2_scaling_policy](#) - Create or delete AWS scaling policies for Autoscaling groups
- [ec2_snapshot](#) - creates a snapshot from an existing volume
- [ec2_tag](#) - create and remove tag(s) to ec2 resources.
- [ec2_vol](#) - create and attach a volume, return volume id and device map
- [ec2_vpc](#) - configure AWS virtual private clouds
- [ejabberd_user](#) - Manages users for ejabberd servers
- [elasticsearch](#) - Manage cache clusters in Amazon ElastiCache.
- [facter](#) - Runs the discovery program *facter* on the remote system
- [fail](#) - Fail with custom message
- [fetch](#) - Fetches a file from remote nodes
- [file](#) - Sets attributes of files
- [filesystem](#) - Makes file system on block device
- [fireball](#) - Enable fireball mode on remote node
- [firewalld](#) - Manage arbitrary ports/services with firewalld
- [flowdock](#) - Send a message to a flowdock
- [gc_storage](#) - This module manages objects/buckets in Google Cloud Storage.
- [gce](#) - create or terminate GCE instances
- [gce_lb](#) - create/destroy GCE load-balancer resources

Thousands
of
Modules

shell module

- name: do what you want

shell: /opt/I_can_do_what_I_want.sh

A role directory

```
└─ rails_app
   └─ files
   └─ handlers
   └─ tasks
      main.yml
   └─ vars
```

Variables

rails_app/vars/main.yml

```
---  
app_dir: /var/www/versioneye
```

rails_app/tasks/main.yml

```
---  
- name: create versioneye directory  
  command: mkdir -p {{ app_dir }}
```

Handlers

mongo/handlers/main.yml

```
---  
- name: restart mongodb  
  service: name=mongod state=restarted
```

mongo/tasks/main.yml

```
- name: copy MongoDB configuration to the server  
  copy: src=mongodb.conf dest=/etc/mongodb.conf  
  notify: restart mongodb
```

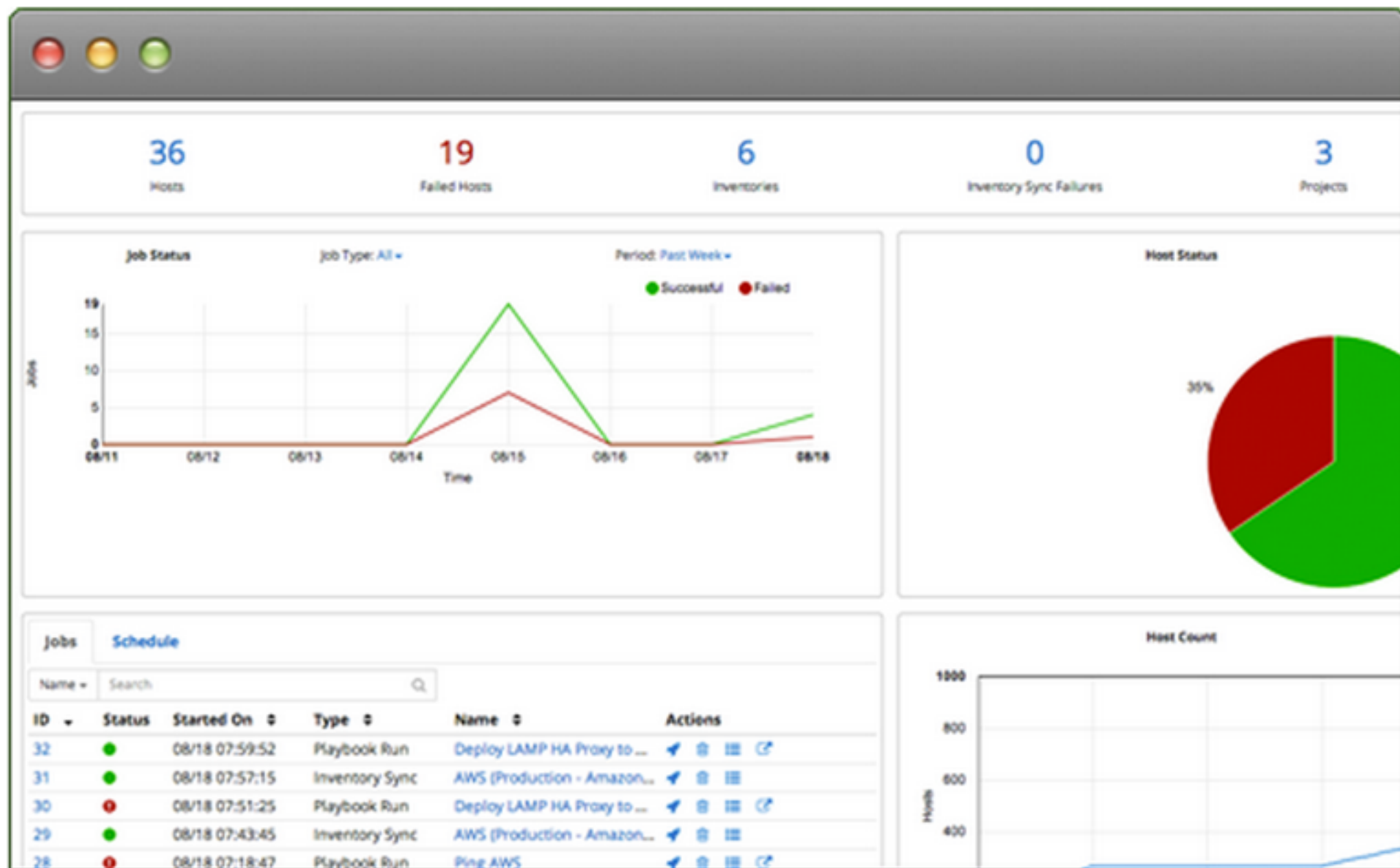
Files

mongo/files/mongodb.list

mongo/tasks/main.yml

- name: add MongoDB debian server to the list of servers
copy: src=mongodb.list dest=/etc/apt/sources.list.d/mongodb.list

Demo



<http://www.ansible.com/tower>

ANSIBLE GALAXY

Ansible Galaxy is your hub for finding, reusing, and sharing the best Ansible content.

<https://galaxy.ansible.com/>

Further Links

Intro do Ansible

<http://blog.versioneye.com/2014/07/03/intro-to-ansible/>

Rebuilding Capistrano like deployment with Ansible

<http://blog.versioneye.com/2014/09/24/rebuilding-capistrano-like-deployment-with-ansible/>

Questions?



@RobertReiz