



Ansible Hands-on Lab

Martijn Cielen

WIFI: Cronos Guest



Agenda

- 09:00 Welcome and setup
- 10:00 Ansible engine
- 11:00 Coffee break
- 11:20 Ansible tower
- 13:00 Lunch



Setting up your machine with Ansible Tower

Based on your user number you received (it's printed on your badge), log in to one of the Tower servers

Users with numbers from **1** to **15** log in to **t1.piros.be**

Users with numbers from **16** to **30** log in to **t2.piros.be**

Users with numbers from **31** log in to **t3.piros.be**



Setting up your machine with Ansible Tower

- Log in with username **wsadmin{number}**. Password is PirosUser!
- Go to the **templates** tab and select `create_azure_vm_{number}`
- Choose notifications and turn Slack notifications on
- Log out
- Log in with username **wsuser{number}**. Password is PirosUser!
- Go to the templates tab and execute `create_azure_vm_{number}`
- Fill in a hostname, your username (`wsuser{number}`) and a password to create a vm on Azure to work with (**and remember them, you'll need them later!**)
 - Use `uservm{number}` as vm name
 - Create a strong password (with caps, lowercase and numeric)



Ansible Engine (crash course)





SIMPLE

Human readable automation
No special coding skills needed
Tasks executed in order
Usable by every team
Supported
Get productive quickly



POWERFUL

App Deployment
Configuration management
Workflow orchestration
Network automation
Orchestrate the app lifecycle



AGENTLESS

Agentless architecture
No agents to exploit or update
Uses OpenSSH and WinRM
Get started immediately
More efficient and more secure

Ansible can do...

Orchestration

Configuration management

Application deployment

Provisioning

Continuous Delivery

Security and Compliance

WHAT
CAN IT
DO?

...on these:

Firewalls, Load balancers and other
network equipment

Applications and containers

Clouds

Servers

Infrastructure

Storage

And more...

WHAT
CAN IT
MANAGE?



Ansible Engine

Concepts

- Modules
- Inventory
- Ad-hoc commands
- Playbooks
- Transports
- Variables
- Facts
- Vaults



Ansible Engine Modules

Modules are pieces of code that enable a specific functionality

Fully documented on docs.ansible.com

Vendors are increasingly developing and shipping modules for their products

You can develop your own modules

Example: service

```
ansible all -m service -a "name=httpd state=started enabled=yes"
```



Ansible ships with over 1500 modules

Cloud: AWS, Azure, OpenStack, Rackspace, ...

Virtualisation: VMware, RHV, ...

Containers: Docker, OpenShift, ...

Windows: ACLs, Commands, Packages, Regedits, Users, ...

Network: Cisco, Dell, F5, Juniper, ...

Notify: HipChat, IRC, Slack, Email, ...

See https://docs.ansible.com/ansible/latest/modules/list_of_all_modules.html for full list



Ansible Engine Inventory

Inventory is the set of nodes to be managed by Ansible. They are stored in an ini formatted file. Examples:

```
[webservers]  
web1.example.com  
server1.example.com
```

```
[dbservers]  
db1.example.com  
server1.example.com
```

```
[wildcardservers]  
*.example.com  
db%.example.com
```

```
[rangeservers]  
w[01:50].example.com  
db-[a:f].example.com
```

```
[example:children]  
webservers  
dbservers
```

```
[specialservers]  
srv.example.com:5143
```

Ansible Engine

Ad-hoc commands

General format:

```
ansible <server pattern> -m <module> -a "<arguments>"
```

Guess what this does:

```
ansible all -m yum -a "name=* state=latest"
```



Ansible Engine

Playbooks

```
# This playbook will install and start/enable Apache
---
- hosts: webservers
  vars:
    http_port: 80
    max_clients: 200
    remote_user: root
  tasks:
    - name: ensure apache is at the latest version
      yum: name=httpd state=latest
    - name: ensure apache is running (and enable it at boot)
      service: name=httpd state=started enabled=yes
```



Ansible Engine

Playbooks

```
# This playbook will not take advantage of Ansible's capabilities
---
- hosts: web_servers
  vars:
    http_port: 80
    max_clients: 200
  remote_user: root
  tasks:
    - name: ensure apache is at the latest version
      yum: name=apache state=latest
    - name: ensure apache is running (and handle at 500)
      service: name=httpd state=started enabled=yes
```

DON'T DO THIS

DO YAML



Ansible Engine Transports

Ansible is agentless, so how can it execute stuff on the target servers?

Transport types:

OpenSSH (port 22) on Linux/Unix/...

Clients need to have python installed

WinRM (port 5986 for https or port 5985 for http) on Windows

Clients need to have Powershell installed

Many other transport types: local/paramiko ssh/network_cli/netconf/...

Sudo and runas constructs are fully supported

Transports are plugins, so are extendable

Ansible has a push architecture

Ansible Engine

Variables (or simply vars)

simple vars in playbook

```
---  
- hosts: webservers  
  vars:  
    http_port: 80  
    max_clients: 200  
  tasks:  
    - name: my task
```

array vars in separate file

```
---  
alpha: ['a', 'b', 'c']  
numbers: [1,2,3,4]
```

dict vars

```
---  
users:  
  alice:  
    name: Alice B  
    phone: +32 12 34 5  
  bob:  
    name: Bob S  
    phone: +32 67 89 0
```



Ansible Engine

Facts

Pieces of information regarding the remote system
eg IP addresses, OS, ethernet devices, ...

Implemented using the “setup” module

Implicitly run at the start of a playbook, so you can use them in your code



Ansible Engine

Facts

```
ansible localhost -m setup

localhost | SUCCESS => {
    "ansible_facts": {
        "ansible_XHC20": {
            "device": "XHC20",
            "flags": [],
            "ipv4": [],
            "ipv6": [],
            "macaddress": "unknown",
            "mtu": "0",
            "type": "unknown"
        },
        ...
    }
}
```



Ansible Engine Vaults

Vaults are encrypted files protected with a password (file)

Any file can be a vault: playbooks, inventory, var files, even binaries

Unlock during playbook run giving password

Password can be asked or delivered using a file or script

Vaulted data is **never** shown in logs

```
ansible-playbook site.yml -vault-password-file ~/.vault_pass.py
```



Ansible Engine Exercises

Let's get some hands-on experience!

Open your browser and go to **<http://piros.be/ansiblelab/Guide.pdf>**

Do not use browser, but download this pdf file.



Ansible Tower



What is Ansible Tower?

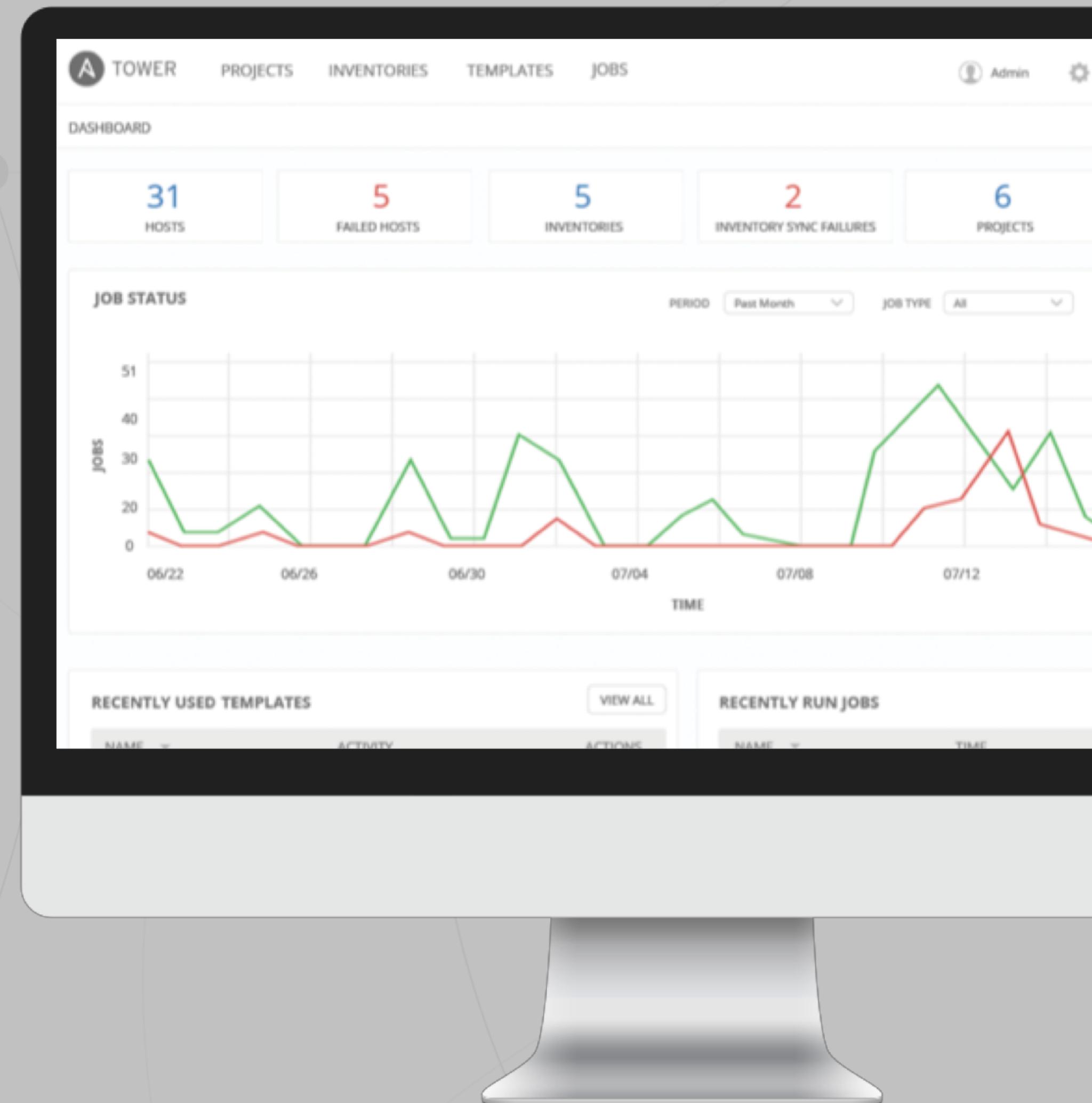
Ansible Tower is an **enterprise framework** for controlling, securing and managing your Ansible automation with a **UI** and **RESTful API**.

Ansible Tower offers:

Role-based access control

Deploy entire applications with **push-button deployment** access

All automations are **centrally logged**





**RED HAT®
ANSIBLE®**
Automation

RED HAT ANSIBLE TOWER

Scale + operationalize your automation

CONTROL

KNOWLEDGE

DELEGATION

RED HAT ANSIBLE ENGINE

Support for your Ansible automation

SIMPLE

POWERFUL

AGENTLESS

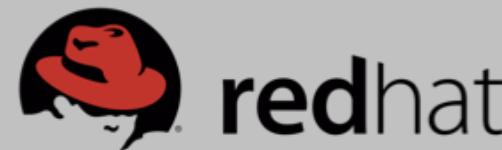
FUELED BY AN INNOVATIVE **OPEN SOURCE** COMMUNITY



Ansible Tower Exercises

Let's get some hands-on experience!

Go to chapter 2 in your manual.



Ansible Tower

Some encores!

Here's some more stuff to look at

Schedule the CIS job to run every hour

Take a look at the CIS template, try and find out what that callback url thing is

Take a look at the various inventory scripts we ship, as well as the types of credentials we have in Tower. What are they for?



Getting started with Ansible and Ansible Tower

This assumes you have access to some form of virtualization on your laptop or workstation, like VirtualBox, KVM, Hyper-V or Xhyve.

Get your free RHEL developer sub at <https://developers.redhat.com/faq/>

Get your Ansible Tower eval sub at <https://www.ansible.com/product/tower/trial>

Create a new VM with RHEL

Install Tower on the RHEL VM, like described at <https://docs.ansible.com/ansible-tower/latest/html/quickinstall/prepare.html>

Create a second VM to manage

...

Feel like an Ansible super hero!



Stuff to read and view

Some texts, videos and courses to look at tomorrow, when you wake up
(or tonight, instead of sleeping).

Name	Location	What?
Ansible Essentials	here	Free course
DO407: Automation with Ansible	here	Course
EX407: Automation with Ansible	here	Exam
Building an Ansible Tower cluster	here	Video
Integrating Tower and Satellite 6	here	Video
Getting started docs	here	Documentation
More Ansible video resources	here	Videos
Ansible and Windows	here	Documentation



From Red Hat Summit 2018

Name	Location	What?
Manage Windows Like Linux with Ansible	here	Summit Breakout Session
Manage 15,000 network devices with Ansible	here	Summit Breakout Session
Push Button Deployments, 3rd Edition	here	Summit Breakout Session
Hybrid Cloud Network Interconnect	here	Summit Breakout Session
Ansible Benelux Meetup	here	Local Meetup around Ansible





Martijn Cielen
Technical business developer
mc@piros.be
+32 475 757 998

Thank You!