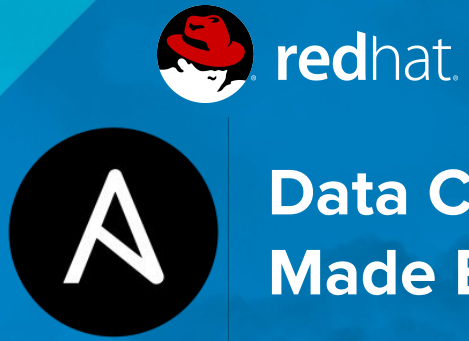


redhat.

Data Center Automation Made Easy With Ansible

Steffen Frömer, Technical Account Manager
Martin Reinke, Senior Solution Architect

Customer Community Meeting
Hamburg, 6 July 2017

Who we are?

Steffen Frömer *(“coding brain”)*

Technical Account Manager @ Red Hat
Global Support Services (part of Engineering)

- Living nearby Braunschweig
  Have fun with Garden, House and Barbecue
- Starting professional IT 2006 at German Aerospace Center (DLR)
- Joined Red Hat May 2016
- You can find me on Xing, LinkedIn, GitHub (Knumskull)

Martin Reinke *(“presenter”)*

Senior Solution Architect @ Red Hat
Manufacturing Vertical Germany

- Living nearby Hamburg (Glinde)
  Loves music, travelling, sailing and have fun /w family (2 kids)

- My IT Life Started 1993 at IBM
  - Infrastructure / IT Mgmt Consulting
  - Software Solution Sales / Pre-sales
- Joined Red Hat Oct 2014
- I’m on Xing, LinkedIn, Twitter…

Agenda

Data Center Automation

Ansible kurze Einführung

Virtualization Use Case
- Hypervisor Management
- Virtual Machine Management

Use Case Demo

Data Center + Automation

Nothing routine in IT should be done manually!

Agenda

Data Center Automation

Ansible kurze Einführung

Virtualization Use Case
- Hypervisor Management
- Virtual Machine Management

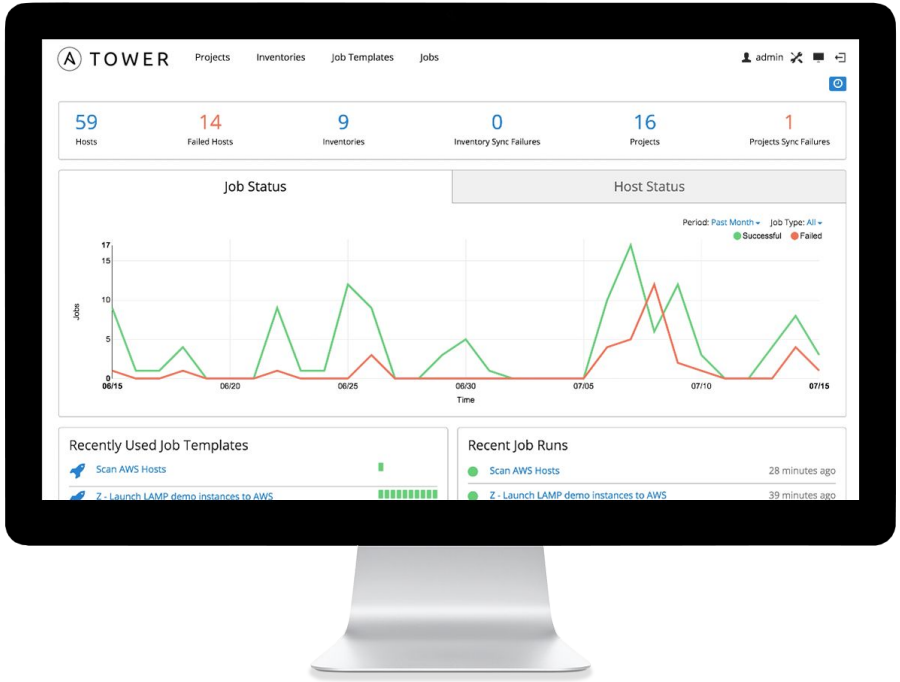
Use Case Demo

WHAT IS ANSIBLE?

It's a **simple automation language** that can perfectly describe an IT application infrastructure in Ansible Playbooks.

It's an **automation engine** that runs Ansible Playbooks.

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



Automate the deployment and management of your entire IT footprint

DO THIS...

Orchestration
Configuration management
Application deployment
Provisioning
Continuous delivery
Security and compliance

ON THESE...

Firewalls
Load Balancers
Applications
Containers
Clouds
Servers
Infrastructure
Network devices
(And more)

ANSIBLE MODULE SAMPLING

CLOUD

AWS
Azure
Centurylink
Cloudscale
Digital Ocean
Docker
Google
Linode
Openstack
Rackspace
+more

VIRT AND CONTAINER

Docker
VMware
RHEV
Openstack
OpenShift
Atomic
Cloudstack
+more

WINDOWS

ACLs
Files
Commands
Packages
IIS
Regedits
Shell
Shares
Services
Configs
Users
Domains
+more

NETWORK

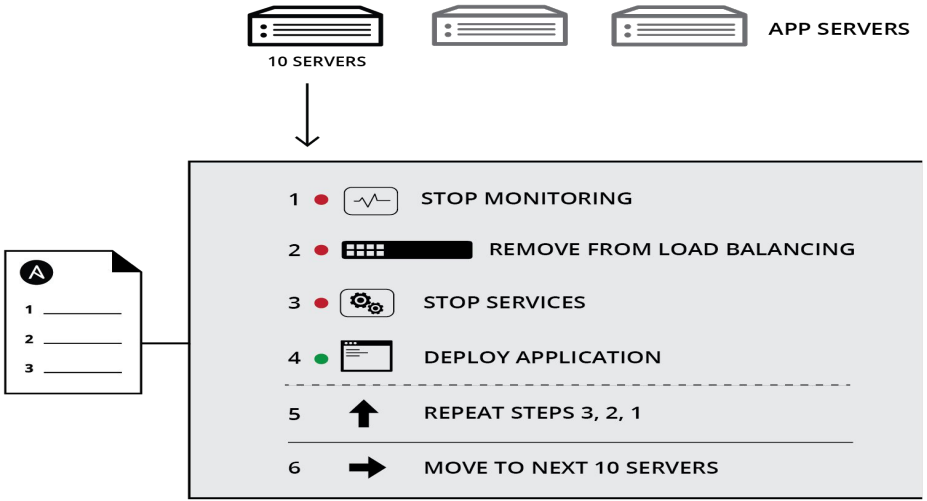
Arista
A10
Cumulus
Bigswitch
Cisco
Cumulus
Dell
F5
Juniper
Palo Alto
OpenSwitch
+more

NOTIFY

HipChat
IRC
Jabber
Email
Rocketchat
Sendgrid
Slack
Twilio
+more

Ansible ships with 900+ modules

ANSIBLE: ORCHESTRATION



Combination of configuration **and** order

Configuration Management alone is not sufficient for app deployment

Allows you to automate entire sets of process

THE ANSIBLE WAY

CROSS PLATFORM – Linux, Windows, UNIX, Networks, Clouds

Agentless support for all major OS variants, **physical**, **virtual**, **cloud** and **network devices**

HUMAN READABLE – YAML

Perfectly describe and document every aspect of your application environment

PERFECT DESCRIPTION OF APPLICATION

Every change can be made by playbooks, ensuring everyone is on the same page

VERSION CONTROLLED

Playbooks are plain-text. Treat them like code in your existing version control.

DYNAMIC INVENTORIES

Capture all the servers 100% of the time, regardless of infrastructure, location, etc.

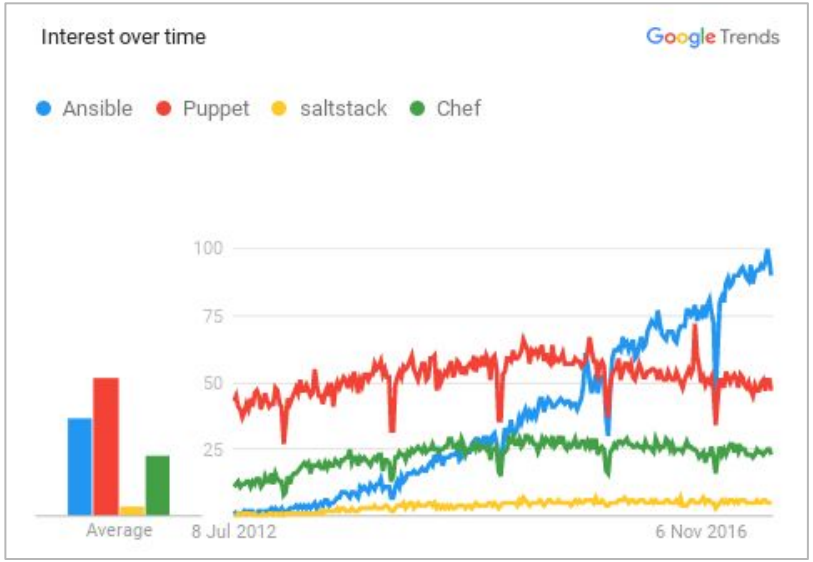
ORCHESTRATION THAT PLAYS WELL WITH OTHERS – HP SA, Puppet, Jenkins, RHNSS, etc.

Homogenize existing environments by leveraging current toolsets and update mechanisms.

THE MOST POPULAR OPEN-SOURCE AUTOMATION COMMUNITY ON GITHUB

AS OF FEB 2017

- 21,000+ stars & 6,800+ forks on GitHub
- 2500+ Contributors (one of the largest open source projects communities)
- Ansible Galaxy: over 10,000 Roles
- 400,000+ downloads a month
- AnsibleFests in San Francisco, London
- Automates events globally



Source: https://twitter.com/reinke_m/status/795566055175442432

SIMPLE

Human readable automation

No special coding skills needed

Tasks executed in order

Get productive quickly

POWERFUL

App deployment

Configuration management

Workflow orchestration

Orchestrate the app lifecycle

AGENTLESS

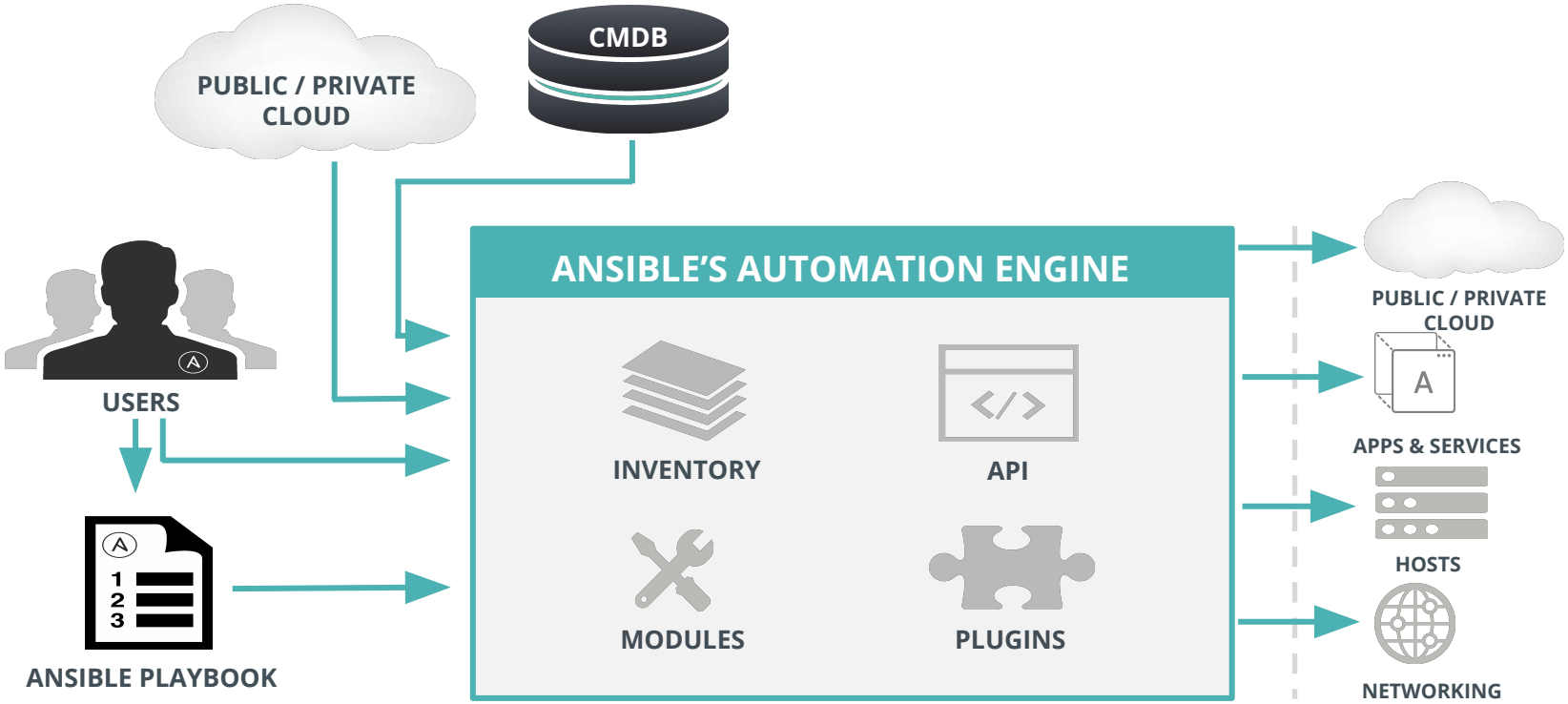
Agentless architecture

Uses OpenSSH & WinRM

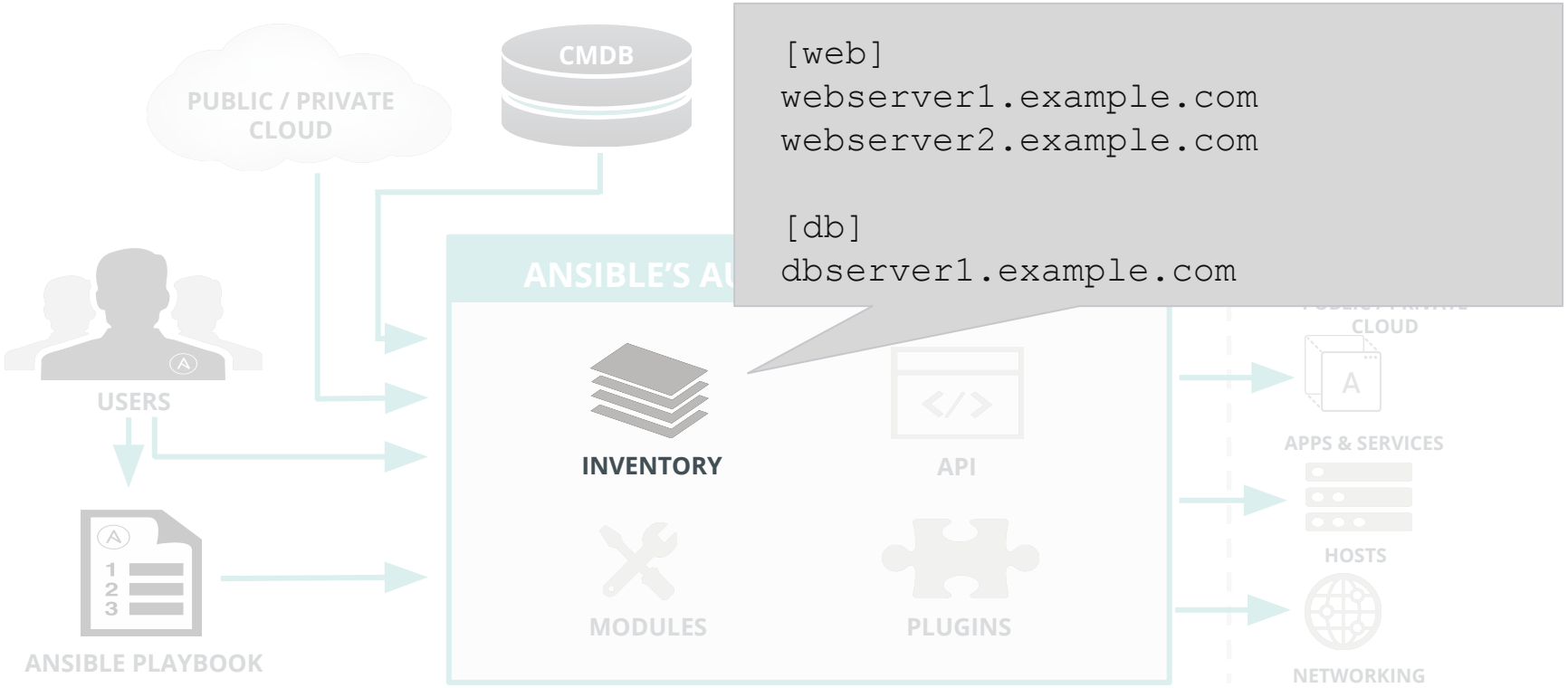
No agents to exploit or update

More efficient & more secure

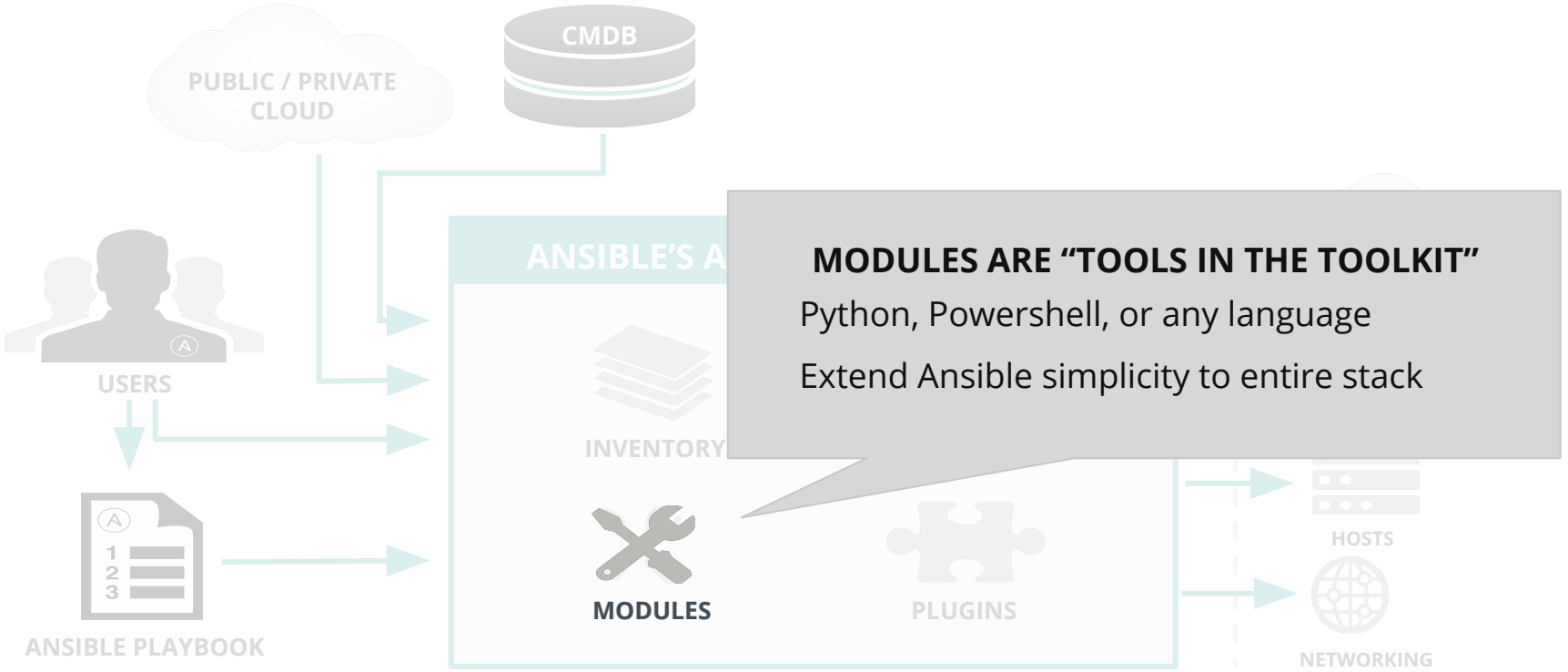
HOW ANSIBLE WORKS



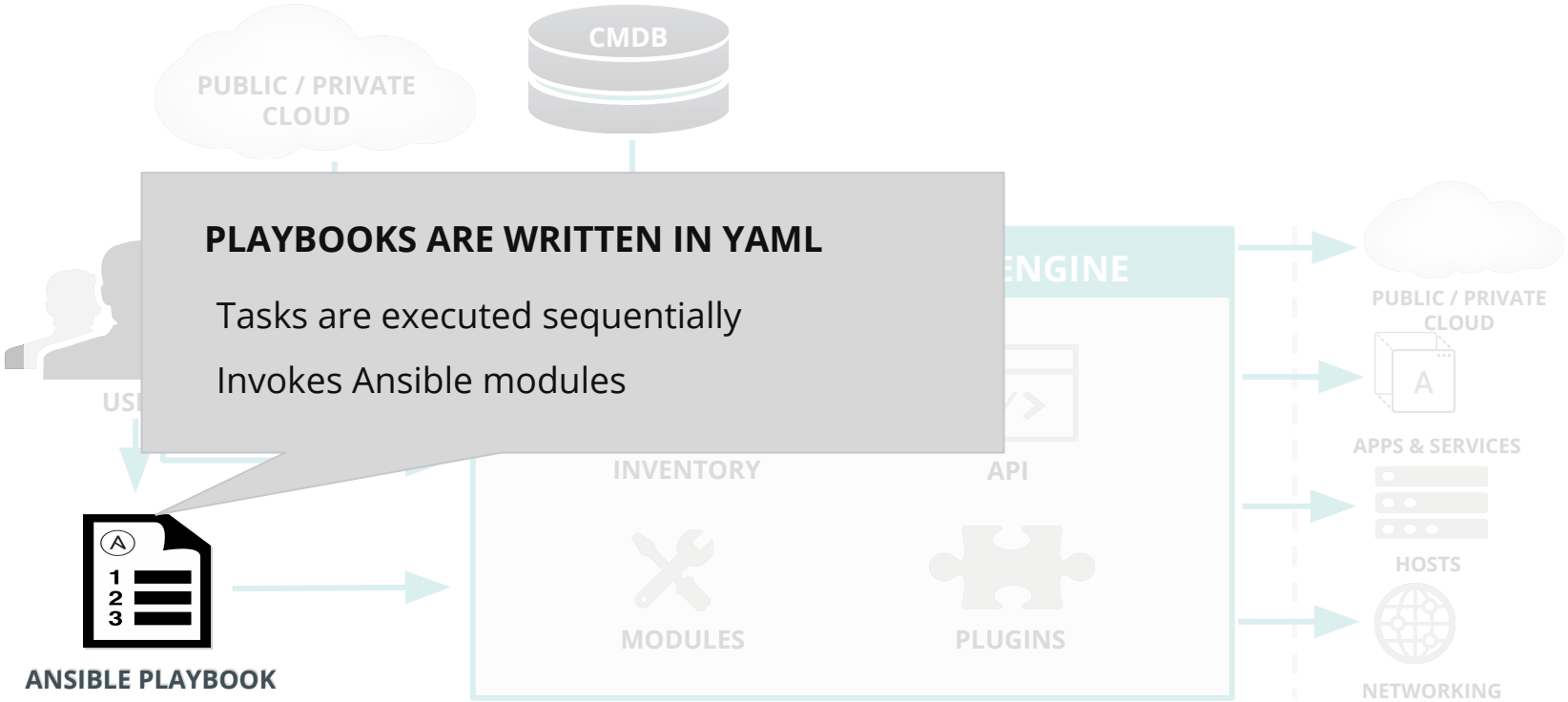
HOW ANSIBLE WORKS



HOW ANSIBLE WORKS



HOW ANSIBLE WORKS



PLAYBOOK EXAMPLE


```
---
- name: install and start apache
  hosts: all
  vars:
    http_port: 80
    max_clients: 200
  remote_user: root

  tasks:
  - name: install httpd
    package: name=httpd state=latest
  - name: write the apache config file
    template: src=/srv/httpd.j2 dest=/etc/httpd.conf
  - name: start httpd
    service: name=httpd state=running
```

PLAYBOOK EXAMPLE


```
---
- name: install and start apache
  hosts: all
  vars:
    http_port: 80
    max_clients: 200
  remote_user: root

  tasks:
  - name: install httpd
    package: name=httpd state=latest
  - name: write the apache config file
    template: src=/srv/httpd.j2 dest=/etc/httpd.conf
  - name: start httpd
    service: name=httpd state=running
```

PLAYBOOK EXAMPLE


```
---
- name: install and start apache
  hosts: all
  vars:
    http_port: 80
    max_clients: 200
  remote_user: root

  tasks:
  - name: install httpd
    package: name=httpd state=latest
  - name: write the apache config file
    template: src=/srv/httpd.j2 dest=/etc/httpd.conf
  - name: start httpd
    service: name=httpd state=running
```

PLAYBOOK EXAMPLE


```
---
- name: install and start apache
  hosts: all
  vars:
    http_port: 80
    max_clients: 200
  remote_user: root

  tasks:
  - name: install httpd
    package: name=httpd state=latest
  - name: write the apache config file
    template: src=/srv/httpd.j2 dest=/etc/httpd.conf
  - name: start httpd
    service: name=httpd state=running
```

What is missing?

SIMPLE

POWERFUL

AGENTLESS

CENTRAL

Central place for everyone

Overview of present and past

Create workflows, schedule jobs

Have one common view

INTEGRATION

Simple, powerful API

Uses REST for quick adoption

No special agents or lib needed

Integrate with everything

ACCESS

Teams and users enable RBAC

Deposit credentials securely

Assign access to unprivileged

Separate access and execution

WHY ANSIBLE TOWER

ANSIBLE TOWER by Red Hat®

Building, managing dynamic inventory

Organizing admin control with users and teams

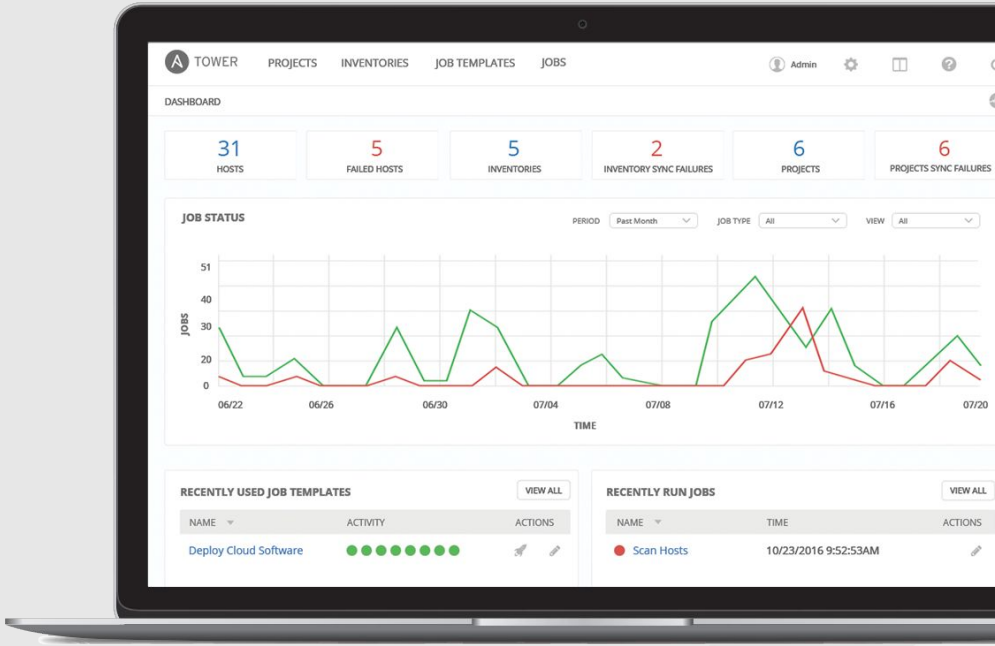
Leverage Ansible Workflows to break up tasks

- new in 3.1

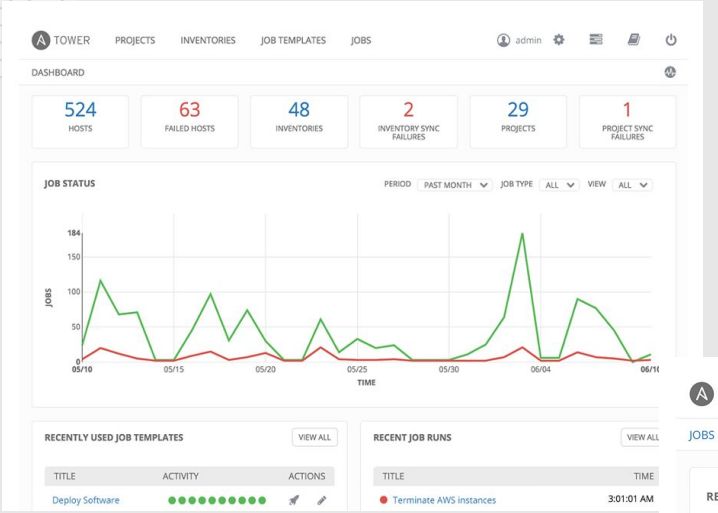
Ongoing compliance

- compare running configs to golden masters on schedules

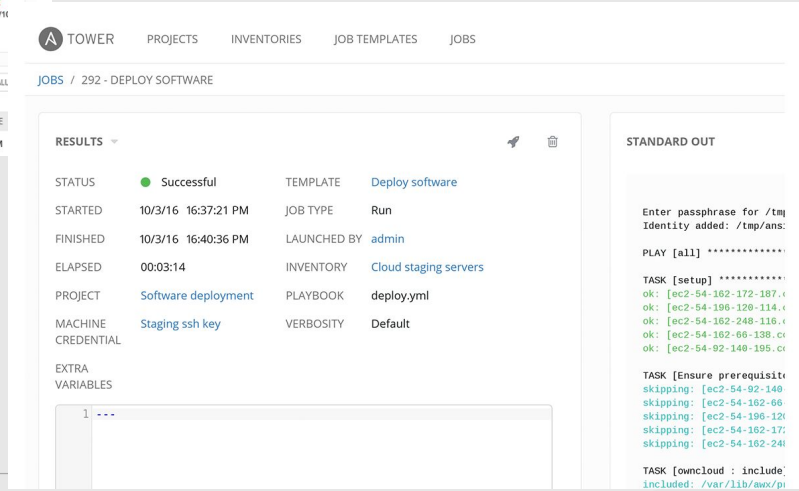
Utilize the RESTful API for anything



DASHBOARD & LIVE VIEW

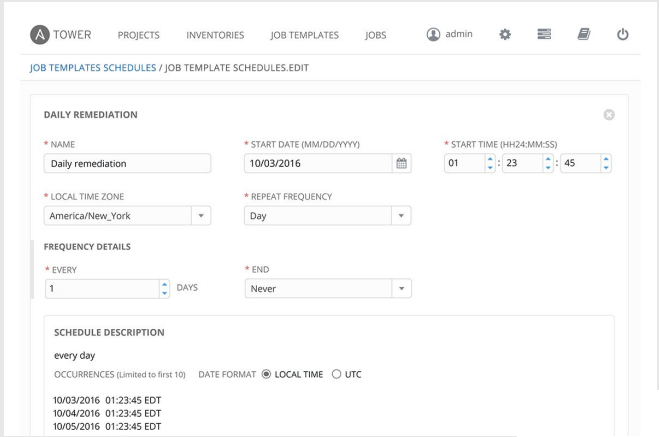


Heads-up NOC-style **automation dashboard** displays everything going on in your Ansible environment.



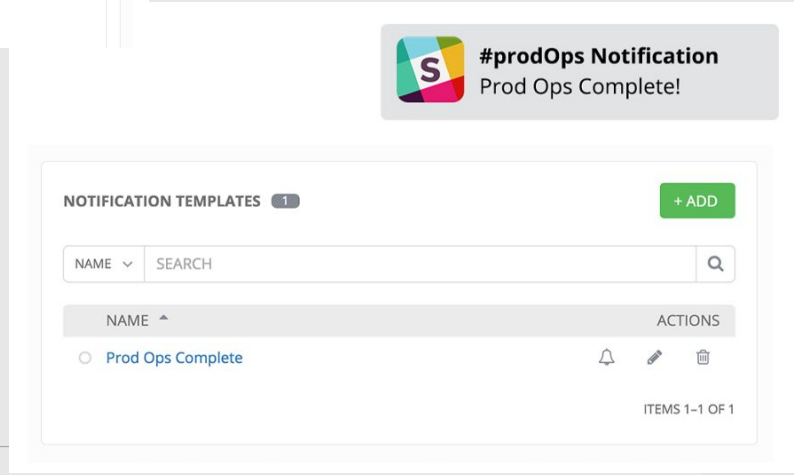
See **real-time output of your automation jobs**, and easily search through output to find exactly what you need.

SCHEDULING & NOTIFICATIONS

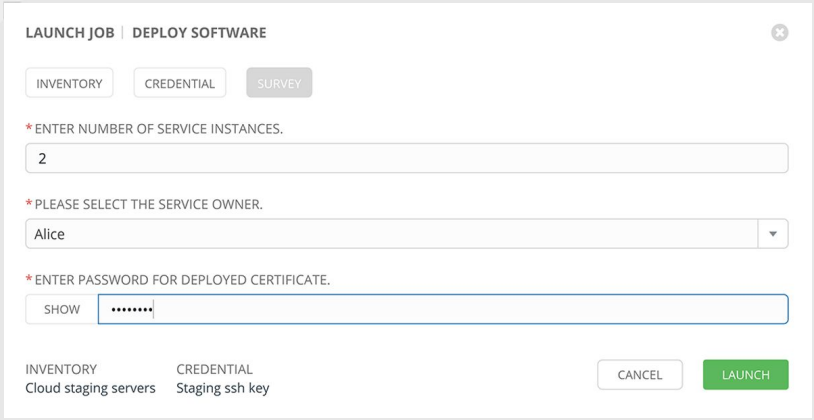


Scheduled Jobs inside Tower enable you to any Job now, later, or forever.

Stay informed of your automation status via **integrated notifications**. Connect Slack, Hipchat, SMS, email, and more.

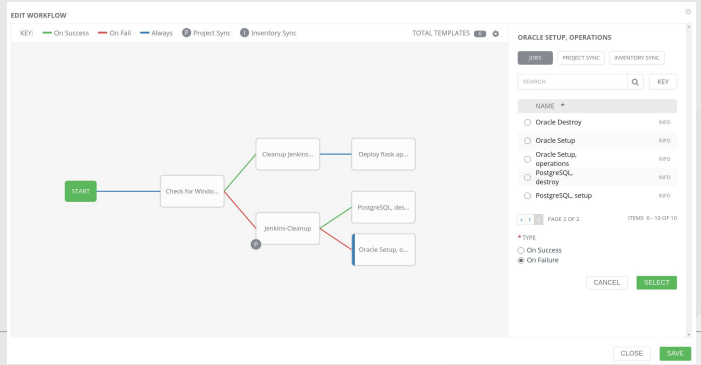


SELF SERVICE & WORKFLOWS



With **self-service,** Tower lets you launch Playbooks with just a single click. It can prompt you for variables, let you choose from available secure credentials and monitor the resulting deployments.

Tower's **multi-Playbook workflows** chains any number of Playbooks together to create a single workflow. Different Jobs can be run depending on success or failure of the prior Playbook.



Agenda

Data Center Automation

Ansible kurze Einführung

Virtualization Use Case & Demo
- Hypervisor Management
- Virtual Machine Management

Virtualization Use Case

Hypervisor Management for RHEV/RHV

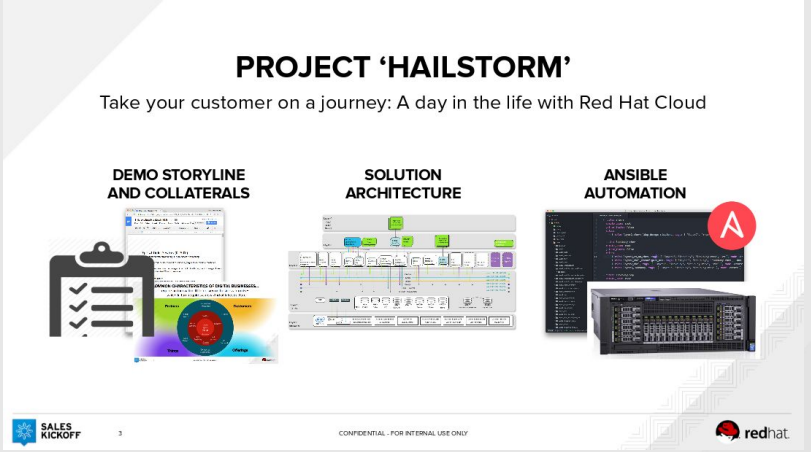
Planning and operations automation tasks

- Installation
- Configuration
- Upgrade of hypervisors (not RHEV-M)

Go further beyond hypervisors and virtual server (outlook)

Introduction to the demo environment

- internal environment to test, run and demonstrate Red Hat multi product integrated solutions
- Wide area of cloud usages scenario's
  - RHEL/RHV, OpenStack, OpenShift, JBOSS Middleware, CloudForms, Ansible Tower, Satellite ++
- chosen for our demo purposes today:
  - Red Hat Virtualization (part only) with some Ansible Automation



You can see/follow our entire project on GitHub
https://github.com/wrichter/hailstorm

What is Red Hat Virtualization (RHV)?

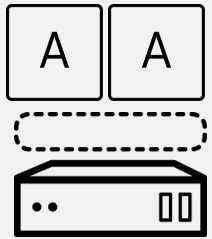
redhat.

What is RHV?

More than just KVM

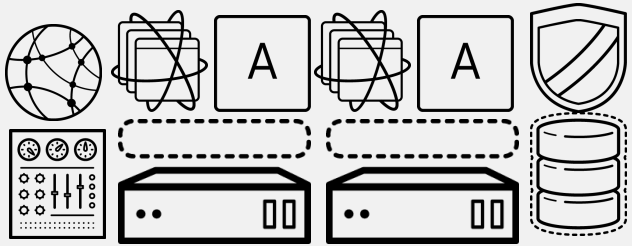
RHEL + KVM

- Basic support for KVM hypervisor
- No enterprise virtualization management features
- Limited number of VMs allowed



Red Hat Virtualization

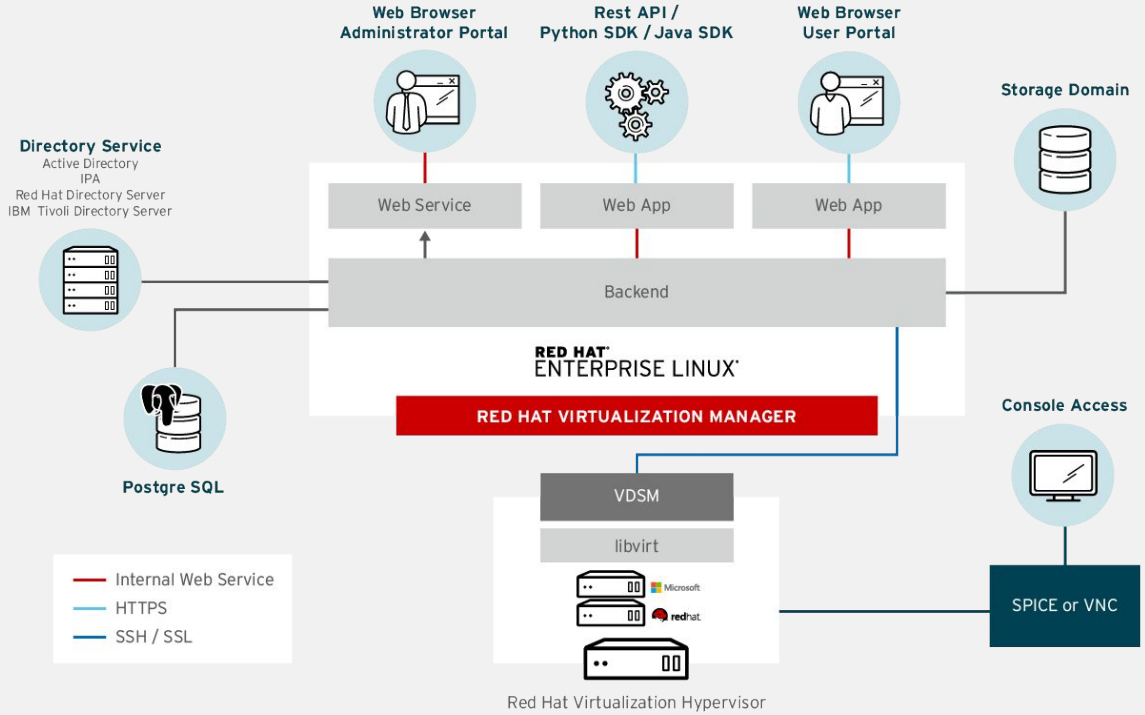
- Build on RHEL+KVM
- Centralized Management for the KVM hypervisor as well as compute, network, and storage resources
- Enterprise features to support mission critical applications



https://access.redhat.com/documentation/en-us/red_hat_virtualization/4.1/html-single/technical_reference/

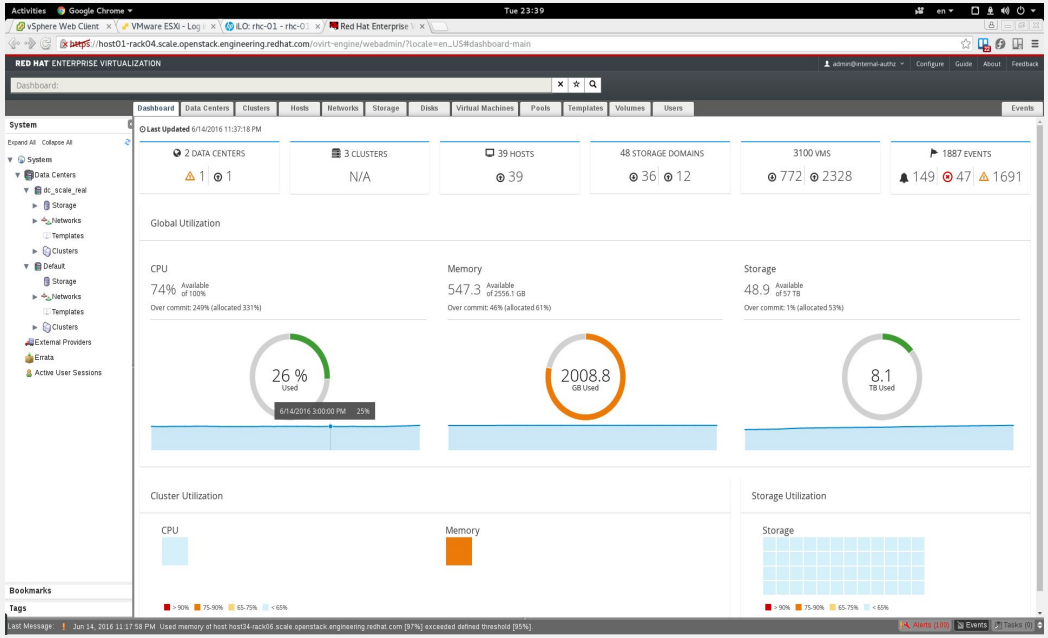
What is RHV?

Overview of RHV



RHV Features

RHV Manager (RHV-M)



- Centralized management of virtualized infrastructure (hosts / servers, VMs, templates, networks, data warehouses)
- Suitable for large installations (500+ hosts and 5,000+ VMs)
- Intuitive dashboard with detailed information

Some more RHV Features

Not comprehensive overview

Core	Live Migration	HA VMs	Affinity / Anti-Affinity	CPU Pinning
	NUMA Support	Resource Reservation	Large Page support	Memory page sharing
	VM templates	Hot Add Memory & CPU	Migrate/Import VMs	Overcommit
Network	VLAN tagging	PCI Passthrough	OVS (tech preview)	Network QoS
	NIC bonding	Jumbo frames	IPV6 (guest)	Network labels
Storage	Storage Live Migration	Thin/Thick Provisioning	Live Snapshots/Merge	SCSI, NFS, FC, GlusterFS
Security	RBAC & Tiered Access	Firewall/SELinux		
Mgmt & APIs	Browser Based Mgmt	Python & Java SDKs	REST APIs	

Overview of the demo

RHV environment setup steps

1. Provision base RHEL Server for hypervisors
   Provision RHEL Server for RHEV Manager
2. Install RHEV Manager
3. Provision Hypervisors, Cluster, Storage etc.
4. Provision Virtual Machines in Hypervisors
5. Do some updates on VMs

Already done (prep work)

Show Ansible playbooks at work

Show manual Ansible work using ad-hoc commands

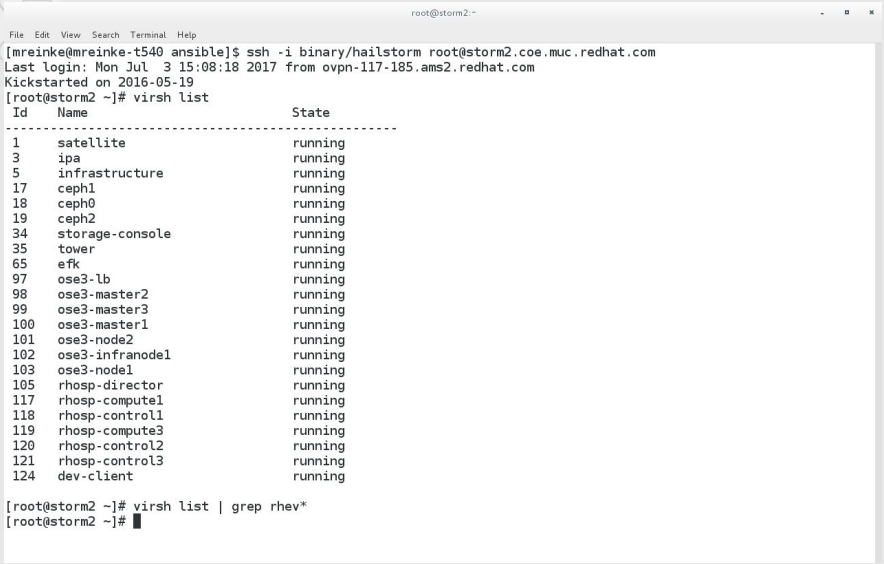
Demo introduction

Some screenshots from the prep work

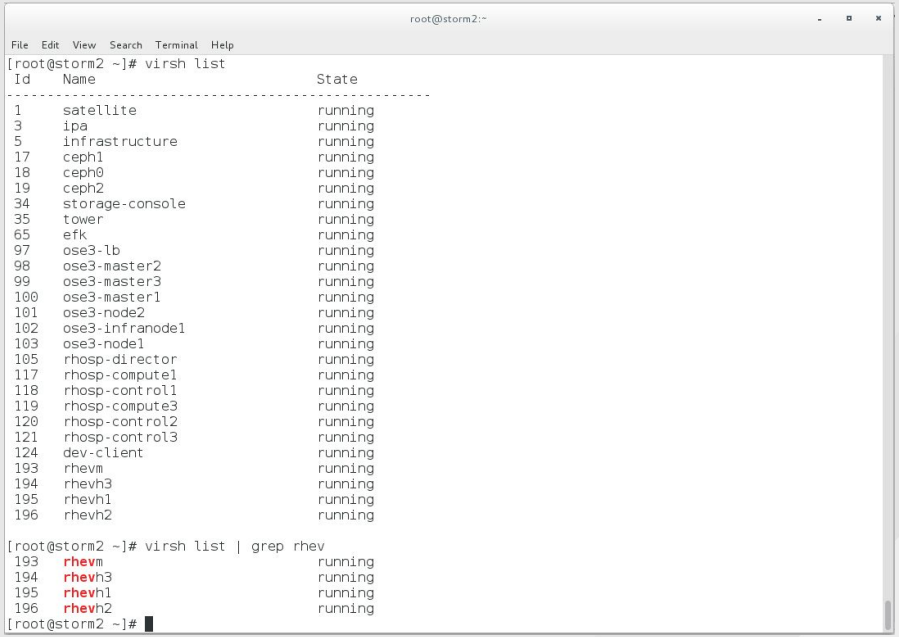
- Joined re-work of ansible automation of the RHEV environment from scratch
- Short / limited time for writing the playbooks
  ➜ done by Steffen in 4 h (+ reading / understanding >1 day)
- ½ day internal dry run
- and … Here we are !

Before - no RHV environment exists

Before Install

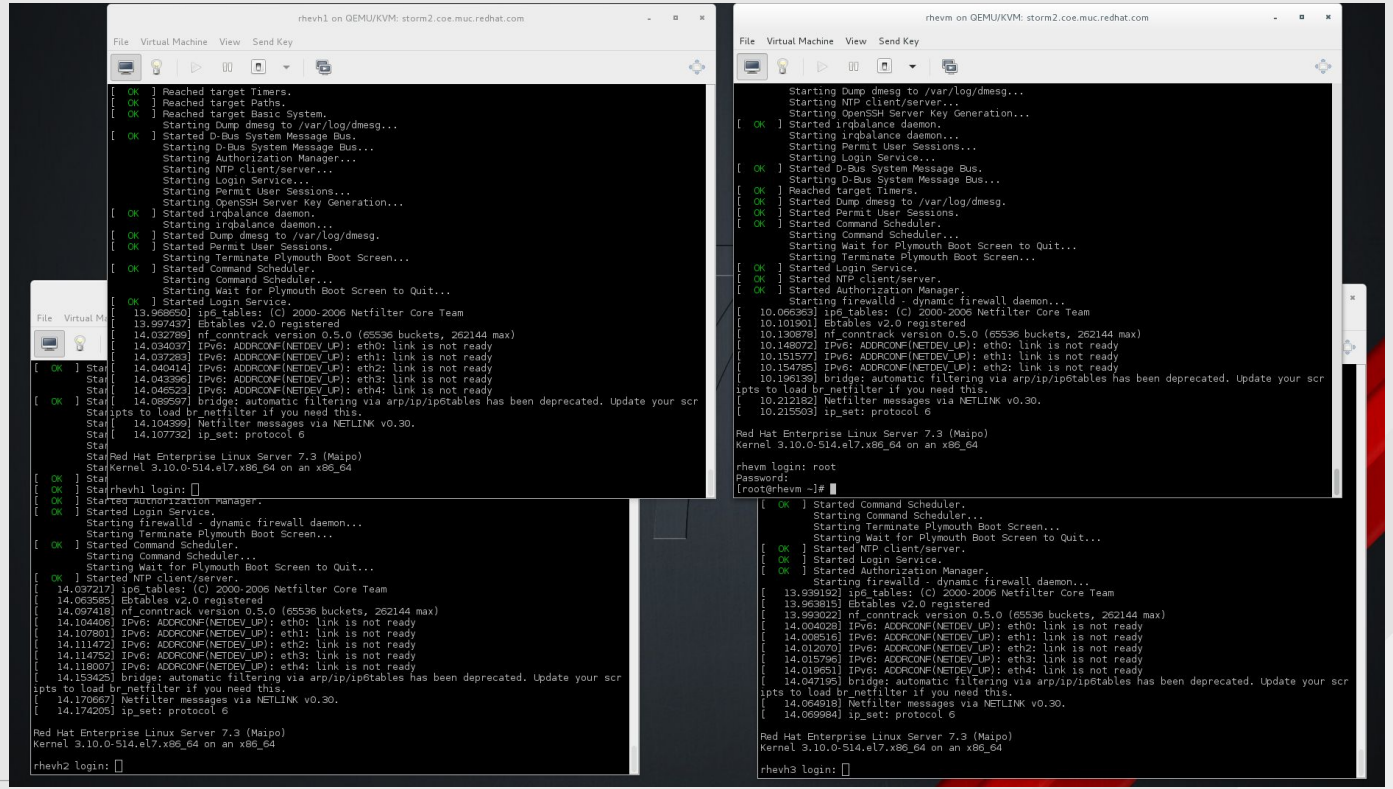


After Install



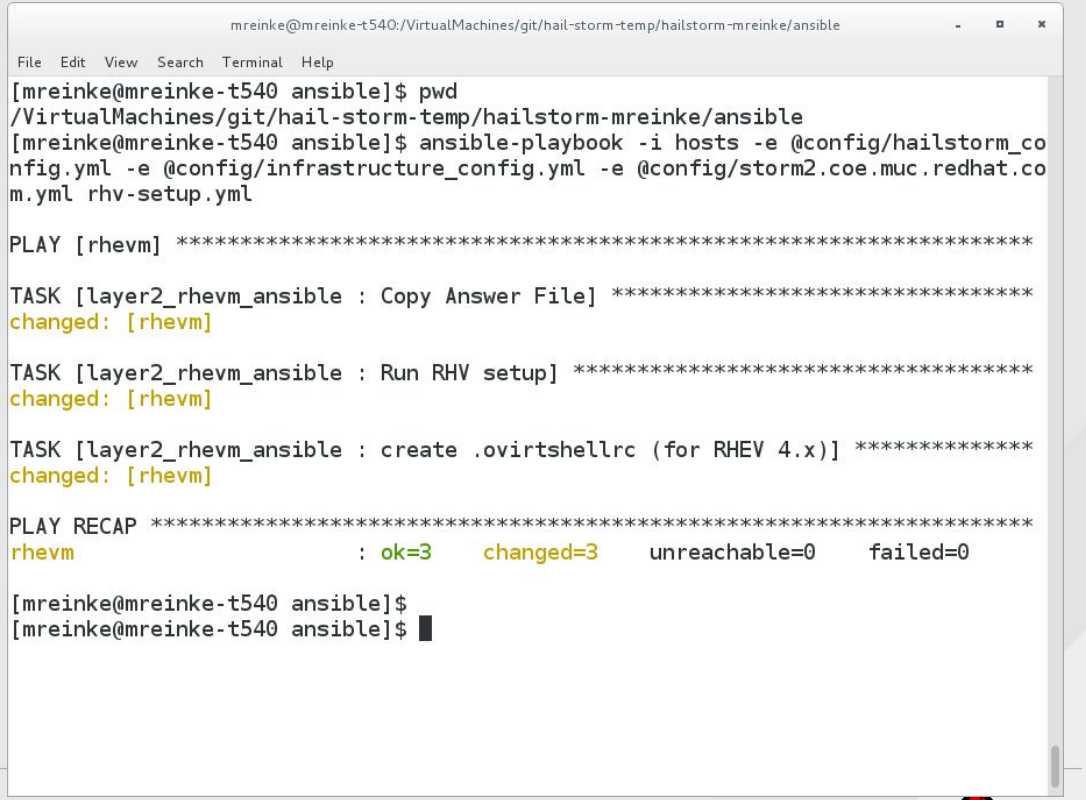
Setup plain RHEL systems for 3 HVs and 1 RHEV-M

- Pre-configured RHELs
- Registered at Satellite
- Repo config done



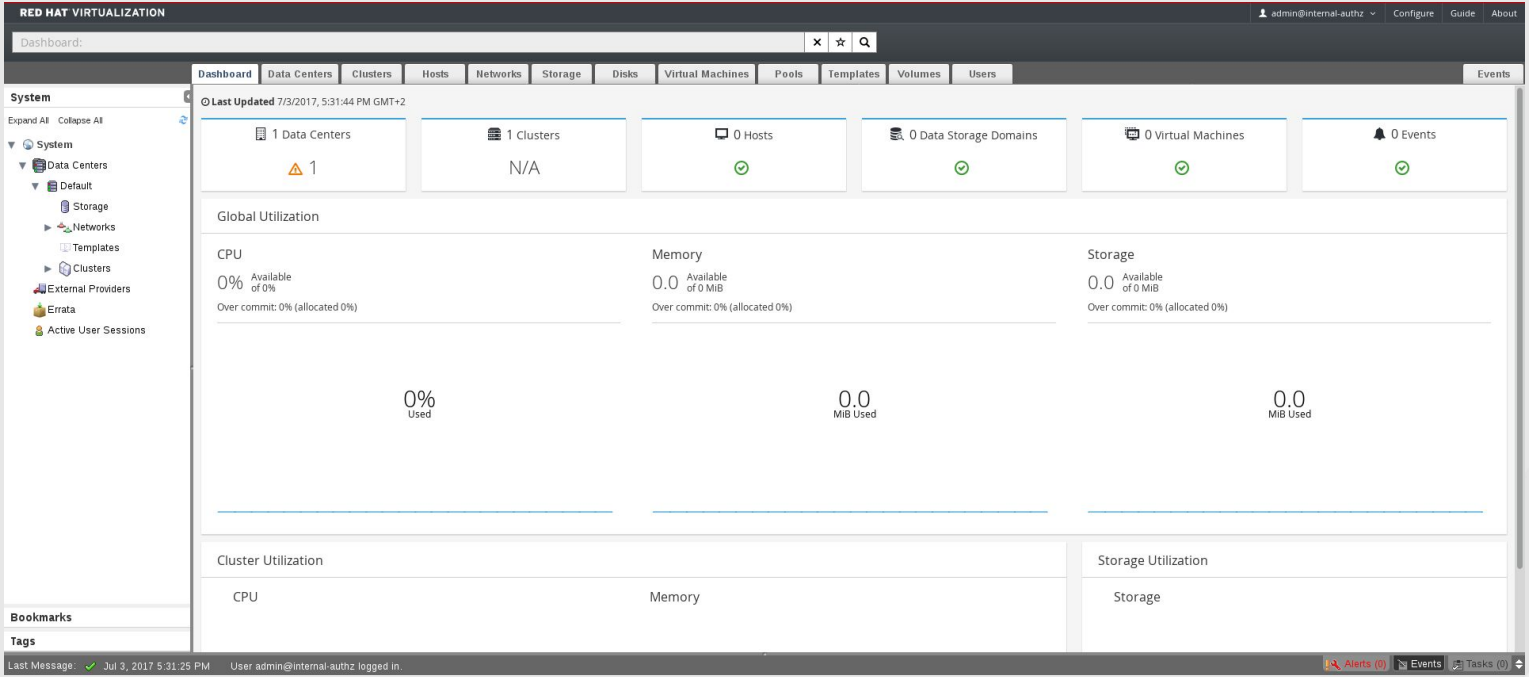
Install RHEV-M

- Create Installation answer file
- Run setup
- after RHEV-M Installation Management UI and API's are working



Blank RHEV Web User Interface

but no hypervisors are managed by it yet



Sounds like a piece of cake

YES!

...but as always, there were some obstacles overcome...

- Ansible version updates
  (always check for, if you want to use latest & greatest)
- Environment related problems
  (libvirtd restart required due to nested virtualization setup)
- Subscriptions expiration
  (needed to renew most of the subscriptions on monday)

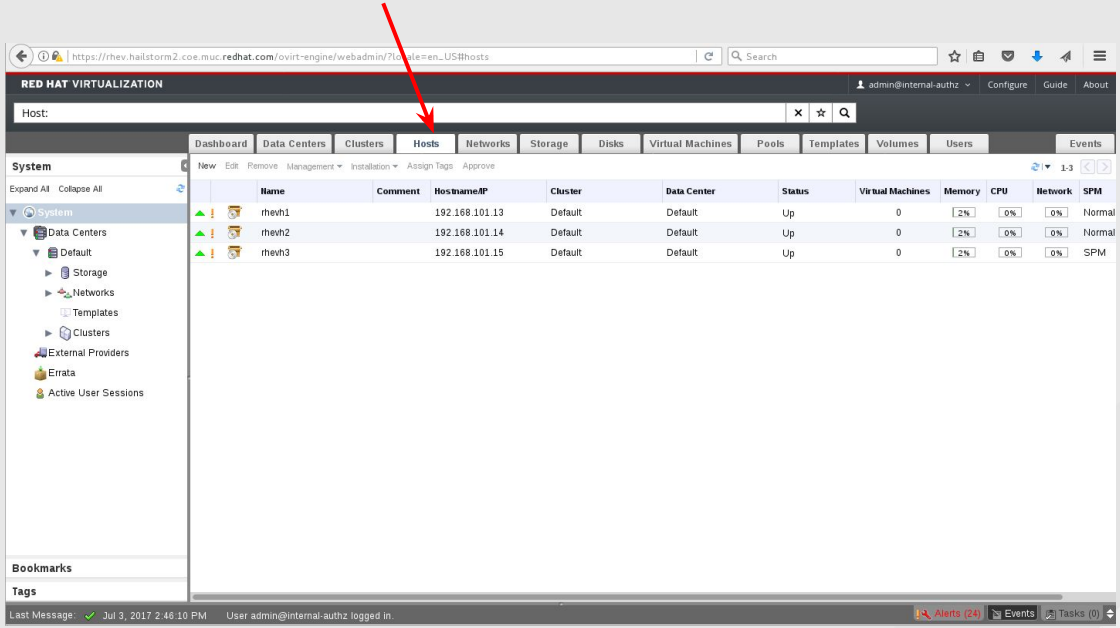

LET’S START TO DO DEMO ...
SHOWTIME

Successfully create RHV environment with 3 HVs

Make 3 RHEL servers connected into RHEV-M as new hypervisors,

Show:

1. RHEV documentation [1]
2. Ansible Playbooks



Links:
[1] - https://access.redhat.com/documentation/en-us/red_hat_virtualization/4.1/html-single/installation_guide/

Successfully create RHV environment with 3 HVs

Ansible Playbook and sample output

```
mreinke@mreinke-t540:/VirtualMachines/git/hail-storm-temp/hailstorm-mreinke/ansible
File  Edit  View  Search  Terminal  Help
[mreinke@mreinke-t540 ansible]$ cat 03-provision-rhv-hv
#!/bin/bash

# Setup Datacenter and Clusterconfiguration
ansible-playbook -i hosts -e @config/hailstorm_config.yml -e @config/infrastructure_config.yml -e @config/storm2.coe.m
uc.redhat.com.yml rhv-deploy-cluster.yml

# Add Hypervisors to RHV cluster
ansible-playbook -i hosts -e @config/hailstorm_config.yml -e @config/infrastructure_config.yml -e @config/storm2.coe.m
uc.redhat.com.yml rhv-deploy-hv.yml

# Setup Datacenter and Clusterconfiguration
ansible-playbook -i hosts -e @config/hailstorm_config.yml -e @config/infrastructure_config.yml -e @config/storm2.coe.m
uc.redhat.com.yml rhv-deploy-storage.yml

# Setup Datacenter and Clusterconfiguration
ansible-playbook -i hosts -e @config/hailstorm_config.yml -e @config/infrastructure_config.yml -e @config/storm2.coe.m
uc.redhat.com.yml rhv-deploy-templates.yml
[mreinke@mreinke-t540 ansible]$ cat rhv-deploy-cluster.yml
---
- hosts: rhevm
  gather_facts: false

  vars:
    engine_url: "https://{{ rhv_dns_domain }}/ovirt-engine/api"
    username: admin@internal

    engine_cafile: /etc/pki/ovirt-engine/ca.pem
    datacenter: Default
    cluster: Default

  tasks:
    - block:
        - name: obtain SSO token
          no_log: true
          ovirt_auth:
            url: "{{ engine_url }}"
            username: "{{ username }}"
            password: "{{ root_password }}"
            ca_file: "{{ engine_cafile }}"

        - name: Ensure Datacenter "{{ datacenter }}" exists
          ovirt_datacenters:
            auth: "{{ ovirt_auth }}"
            name: "{{ datacenter }}"
            comment: "The primary datacenter"
            compatibility_version: 4.1
            quota_mode: enabled
            local: False

        - name: Ensure cluster "{{ cluster }}" exists
          ovirt_clusters:
            auth: "{{ ovirt_auth }}"
```

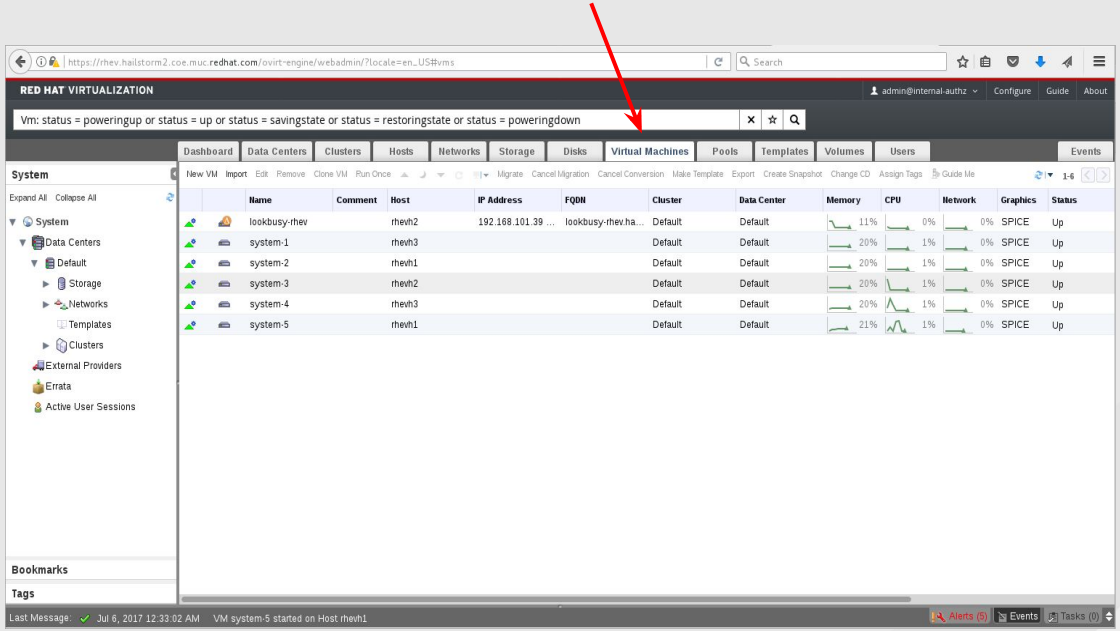
```
mreinke@mreinke-t540:/VirtualMachines/git/hail-storm-temp/hailstorm-mreinke/ansible
File  Edit  View  Search  Terminal  Help
[mreinke@mreinke-t540 ansible]$ ./03-provision-rhv-hv
 [WARNING]: Found both group and host with same name: storage-console

 [WARNING]: Found both group and host with same name: lookbusy-rhev

 [WARNING]: Found both group and host with same name: cloudforms

 [WARNING]: Found both group and host with same name: ipa

 [WARNING]: Found both group and host with same name: test-rhel6

 [WARNING]: Found both group and host with same name: dev-client

 [WARNING]: Found both group and host with same name: infrastructure

 [WARNING]: Found both group and host with same name: rhevm

 [WARNING]: Found both group and host with same name: satellite

 [WARNING]: Found both group and host with same name: ose3-lb

 [WARNING]: Found both group and host with same name: lookbusy-osp

 [WARNING]: Found both group and host with same name: layer1

 [WARNING]: Found both group and host with same name: efk

 [WARNING]: Found both group and host with same name: tower

 [WARNING]: Found both group and host with same name: rhosp-director

 [WARNING]: Found both group and host with same name: test-rhel7


PLAY [rhevm] ***********************************************************************************************************

TASK [obtain SSO token] ************************************************************************************************
ok: [rhevm]

TASK [Ensure Datacenter "Default" exists] ******************************************************************************
changed: [rhevm]

TASK [Ensure cluster "Default" exists] *********************************************************************************
changed: [rhevm]

TASK [Revoke the SSO token] ********************************************************************************************
ok: [rhevm]

PLAY RECAP *************************************************************************************************************
rhevm                      : ok=4    changed=2    unreachable=0    failed=0

 [WARNING]: Found both group and host with same name: storage-console
```

Create Virtual Machines within the RHV environment

Create a number of new RHEL virtual machines on the RHEV hypervisors,

Show:

1. RHEV VM mgmt guide [1]
2. RHEV Admin Guide [2]
3. Ansible Playbooks



Links:
[1] - https://access.redhat.com/documentation/en-us/red_hat_virtualization/4.1/html-single/virtual_machine_management_guide/
[2]- https://access.redhat.com/documentation/en-us/red_hat_virtualization/4.1/html-single/administration_guide/

Create Virtual Machines within the RHV environment

Ansible Playbook and sample output

```
mreinke@mreinke-t540:/VirtualMachines/git/hail-storm-temp/hailstorm-mreinke/ansible
File  Edit  View  Search  Terminal  Help

[mreinke@mreinke-t540 ansible]$ cat 04-provision-rhev-vms
#!/bin/bash

# Setup Datacenter and Clusterconfiguration
ansible-playbook -i hosts -e @config/hailstorm_config.yml -e @config/infrastructure_config.yml -e @config/storm2.coe.muc.redhat.com.yml create-04-democontent.yml --tags layer3,lookbusy,lookbusy-rhev

# Setup Datacenter and Clusterconfiguration
ansible-playbook -i hosts -e @config/hailstorm_config.yml -e @config/infrastructure_config.yml -e @config/storm2.coe.muc.redhat.com.yml rhv-vm-deploy.yml

[mreinke@mreinke-t540 ansible]$ cat create-04-democontent.yml
- hosts: rhevm
  remote_user: root
  gather_facts: false
  roles:
    - { role: layer2_rhevm_ldap_demopermissions, tags: [ 'layer2', 'rhev', 'rhevm', 'rhevm-ldap' ], mode: create }

- hosts: lookbusy-rhev
  remote_user: root
  gather_facts: false
  roles:
    - { role: layer3_vm_on_rhev, tags: [ 'layer3', 'lookbusy', 'lookbusy-rhev', 'vm'], mode: create }
    - { role: layer2_rhel_reconfigure_dns, tags: [ 'layer3', 'lookbusy', 'lookbusy-rhev', 'dns' ], mode: create, nameserver: "{{ hostvars['ipa'].vm_nics[0].ip }}" }
    - { role: layer2_rhel, tags: [ 'layer3', 'lookbusy', 'lookbusy-rhev', 'rhel'], mode: create }
    - { role: layer3_lookbusy, tags: [ 'layer3', 'lookbusy', 'lookbusy-rhev' ], mode: create }

- hosts: lookbusy-osp
  remote_user: root
  gather_facts: false
  roles:
    - { role: layer3_vm_on_openstack, tags: [ 'layer3', 'lookbusy', 'lookbusy-osp', 'vm'], mode: create }
    - { role: layer2_rhel, tags: [ 'layer3', 'lookbusy', 'lookbusy-osp', 'rhel'], mode: create }
    - { role: layer3_lookbusy, tags: [ 'layer3', 'lookbusy', 'lookbusy-osp' ], mode: create }

- hosts: rhosp-director
  remote_user: root
  gather_facts: false
  roles:
    - { role: layer2_rhosp_overcloud_demo_content, tags: [ 'layer2', 'rhosp', 'overcloud', 'democontent' ], mode: create }

- hosts: cloudforms
  remote_user: root
  gather_facts: false
  roles:
    - { role: layer3_cloudforms_content, tags: [ 'layer3', 'cf', 'cfme-content' ], mode: create, content_location: "../CloudForms-Internals" }

- hosts: ose3-installer
  remote_user: root
  gather_facts: false
  roles:
    - { role: layerX_openshift_devops_tools, tags: [layer2, 'ose3', 'ose3-devops'], mode: create }
    - { role: layerX_openshift_demo_monster, tags: [layer2, 'ose3', 'ose3-demo', 'ose3-demo-ticketmonster'], mode: create }
    - { role: layerX_openshift_demo_wordpress, tags: [layer2, 'ose3', 'ose3-demo', 'ose3-demo-wordpress'], mode: create }
    - { role: layerX_openshift_demo_amq, tags: [layer2, 'ose3', 'ose3-demo-amq'], mode: create }
    - { role: layerX_openshift_cfme_mwmgr, tags: [layer2, 'ose3', 'ose3-cfme-mwmgr'], mode: create }
```

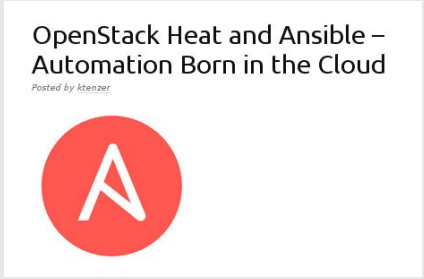
```
mreinke@mreinke-t540:/VirtualMachines/git/hail-storm-temp/hailstorm-mreinke/ansible
File  Edit  View  Search  Terminal  Help
[mreinke@mreinke-t540 ansible]$ ./04-provision-rhev-vms
 [WARNING]: Found both group and host with same name: storage-console

 [WARNING]: Found both group and host with same name: lookbusy-rhev

 [WARNING]: Found both group and host with same name: cloudforms

 [WARNING]: Found both group and host with same name: ipa

 [WARNING]: Found both group and host with same name: test-rhel6

 [WARNING]: Found both group and host with same name: dev-client

 [WARNING]: Found both group and host with same name: infrastructure

 [WARNING]: Found both group and host with same name: rhevm

 [WARNING]: Found both group and host with same name: satellite

 [WARNING]: Found both group and host with same name: ose3-lb

 [WARNING]: Found both group and host with same name: lookbusy-osp

 [WARNING]: Found both group and host with same name: layer1

 [WARNING]: Found both group and host with same name: efk

 [WARNING]: Found both group and host with same name: tower

 [WARNING]: Found both group and host with same name: rhosp-director

 [WARNING]: Found both group and host with same name: test-rhel7


PLAY [rhevm] ***********************************************************************************************************

PLAY [lookbusy-rhev] ***************************************************************************************************

TASK [layer3_vm_on_rhev : include] **************************************************************************************
included: /home/mreinke/VirtualMachines/git/hail-storm-temp/hailstorm-mreinke/ansible/roles/layer3_vm_on_rhev/tasks/instantiate_vm_rhv4.yml for lookbusy-rhev

TASK [layer3_vm_on_rhev : find existing VM] *****************************************************************************
ok: [lookbusy-rhev -> 192.168.103.12]

TASK [layer3_vm_on_rhev : debug] ****************************************************************************************
ok: [lookbusy-rhev] => {
    "rhev_image": "RHEL7"
}

TASK [layer3_vm_on_rhev : create VM from template] **********************************************************************
skipping: [lookbusy-rhev]

TASK [layer3_vm_on_rhev : wait for VM to unlock] ************************************************************************
skipping: [lookbusy-rhev]

TASK [layer3_vm_on_rhev : get nic info] *********************************************************************************
skipping: [lookbusy-rhev]

TASK [layer3_vm_on_rhev : debug] ****************************************************************************************
skipping: [lookbusy-rhev]

TASK [layer3_vm_on_rhev : reconfigure improperly configured nics] *******************************************************
```

Simple adhoc management tasks

```
$: cat ovirt-credentials.yml
ovirt_auth:
  url: "https://{{ rhv_dns_domain }}/ovirt-engine/api"
  username: "admin@internal"
  password: "{{ root_password }}"
  ca_file: "/etc/pki/ovirt-engine/ca.pem"

$: alias ansible-h2
alias ansible-h2='ansible -e @config/hailstorm_config.yml -e @config/infrastructure_config.yml -e @config/storm2.coe.muc.redhat.com.yml'

# shutdown VM system-1
$: ansible-h2 rhevm -e @ovirt-credentials.yml -m ovirt_vms --args='auth={{ ovirt_auth }} name=system-1 state=stopped'

# remove VM system-1
$: ansible-h2 rhevm -e @ovirt-credentials.yml -m ovirt_vms --args='auth={{ ovirt_auth }} name=system-1 state=absent'

# migrate VM system-2 to Hypervisor rhevh1
$: ansible-h2 rhevm -e @ovirt-credentials.yml -m ovirt_vms --args='auth={{ ovirt_auth }} name=system-2 host=rhevh1'

# put HV rhevh3 into maintenance
$: ansible-h2 rhevm -e @ovirt-credentials.yml -m ovirt_hosts --args='auth={{ ovirt_auth }} name=rhevh3 state=maintenance'

# create VM with cloud-init
$: cat cloudinit.yml
cloudinit:
  user_name: root
  root_password: "{{ root_password }}"

$: ansible-h2 rhevm -e @cloudinit.yml -e @ovirt-credentials.yml -m ovirt_vms --args='auth={{ ovirt_auth }} name=system-6 cluster=Default
template=RHEL7 cloud_init={{ cloudinit }}'
```

Further story

- Network automation solution
  - see inside and at [1]
- Openstack & wordpress Application sample
  - *see link from Keith Tenzer blog [2]*
- Red Hat Cloud Solution
  - Container / Openshift Installer
  - Hyper Converged Infrastructure (HCI) Installer
- Training
  - Free starter training (see next slides)



Links:
[1] - https://www.ansible.com/networking
[2] - https://keithtenzer.com/2016/05/09/openstack-heat-and-ansible-automation-born-in-the-cloud/

WHERE DO I BEGIN?

Learn Ansible

- Join existing Ansible network automation communities
- Take Ansible training courses from Red Hat or elsewhere

Start small!

- Create Playbooks that read or check only
- Create simple jobs that eliminate the most annoying tasks
- Leverage existing knowledge internally

Develop success criteria

- Create specific goals that require planning, tailored to your organization
- Create phases to ensure people and processes aren't alienated

Ressource

Startup Training - Ansible Essentials (Check for free)

goo.gl/qBtFxQ (shortened URL)
redhat.com/rhtapps/promo-do007 (direct register link to start)

Download Ansible v2.3:

releases.ansible.com/ansible/

Evaluate Ansible Tower:

ansible.com/tower-trial/

Join the Community

Users list: ansible-project
Development list: ansible-devel
Announcement list: ansible-announce *(read only)*
irc.freenode.net: #ansible

redhat.

THANK YOU

plus.google.com/+RedHat

facebook.com/redhatinc

linkedin.com/company/red-hat

twitter.com/RedHatNews

youtube.com/user/RedHatVideos