

Ansible Automatisierung mit Ansible Tower und Red Hat Satellite

DI (FH) René Koch
rkoch@rk-it.at
Freelancer

Secure Linux Administration Conference, 29.05.2019

Inhalt

- Ansible Engine
- Ansible Tower
- Red Hat Satellite
- Integration Ansible (Tower) und Red Hat Satellite
- Fragen und Antworten



Was ist Ansible?

Was ist Ansible?

- Automatisierung von Standard-Aufgaben wie Provisionierung von Systemen, Applikations-Deployment und Konfigurations-Management, Task-Automatisierung, Dokumentation
- Kein Agent wird am Zielsystem benötigt
- Verwendung von SSH, PowerShell, APIs
- Standardmäßig parallele Abarbeitung der Task auf mehreren Maschinen
- Einfach lesbare Automatisierungssprache (YAML-Format)



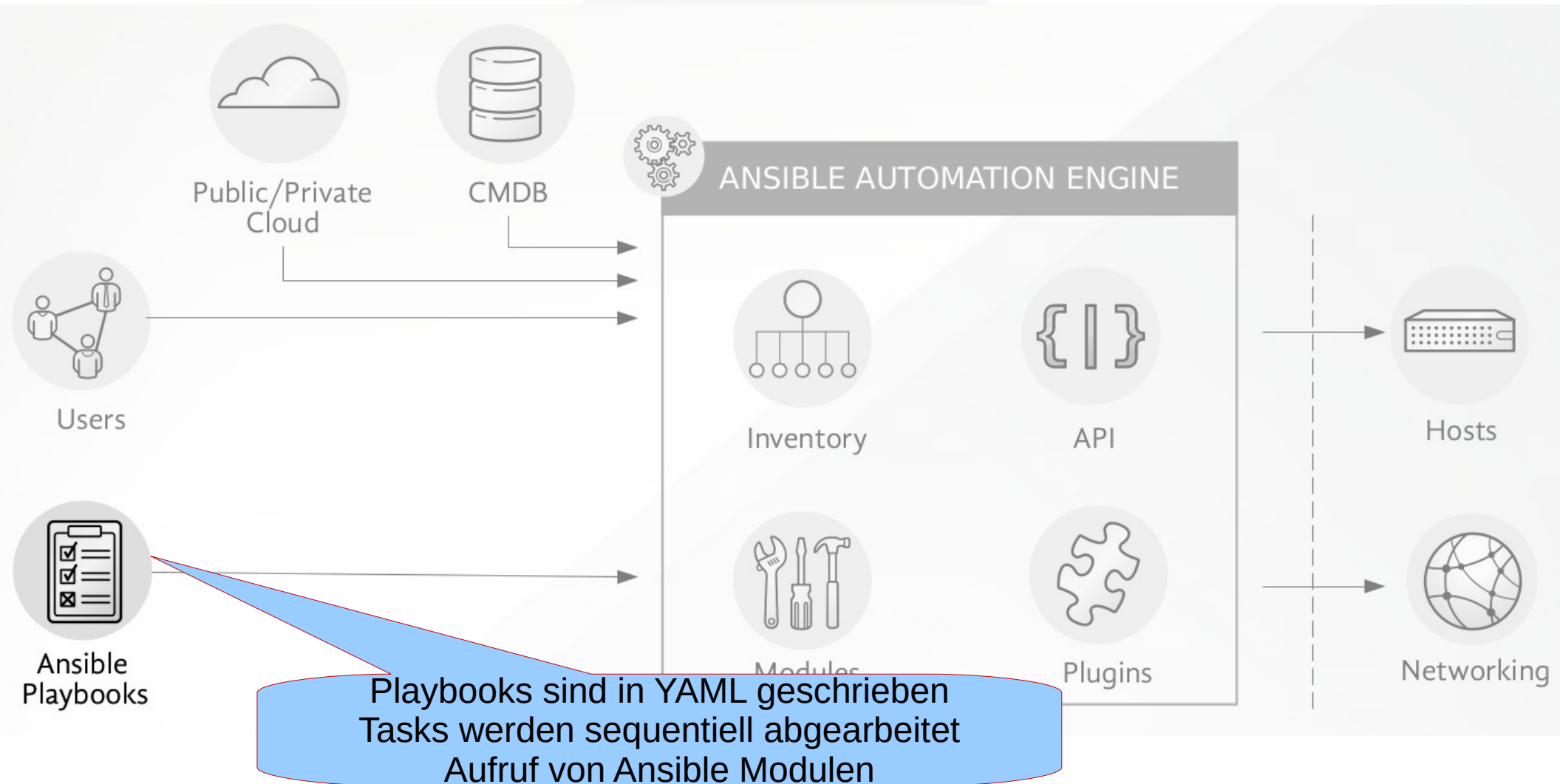
Was ist Ansible?

- Alternative zu:

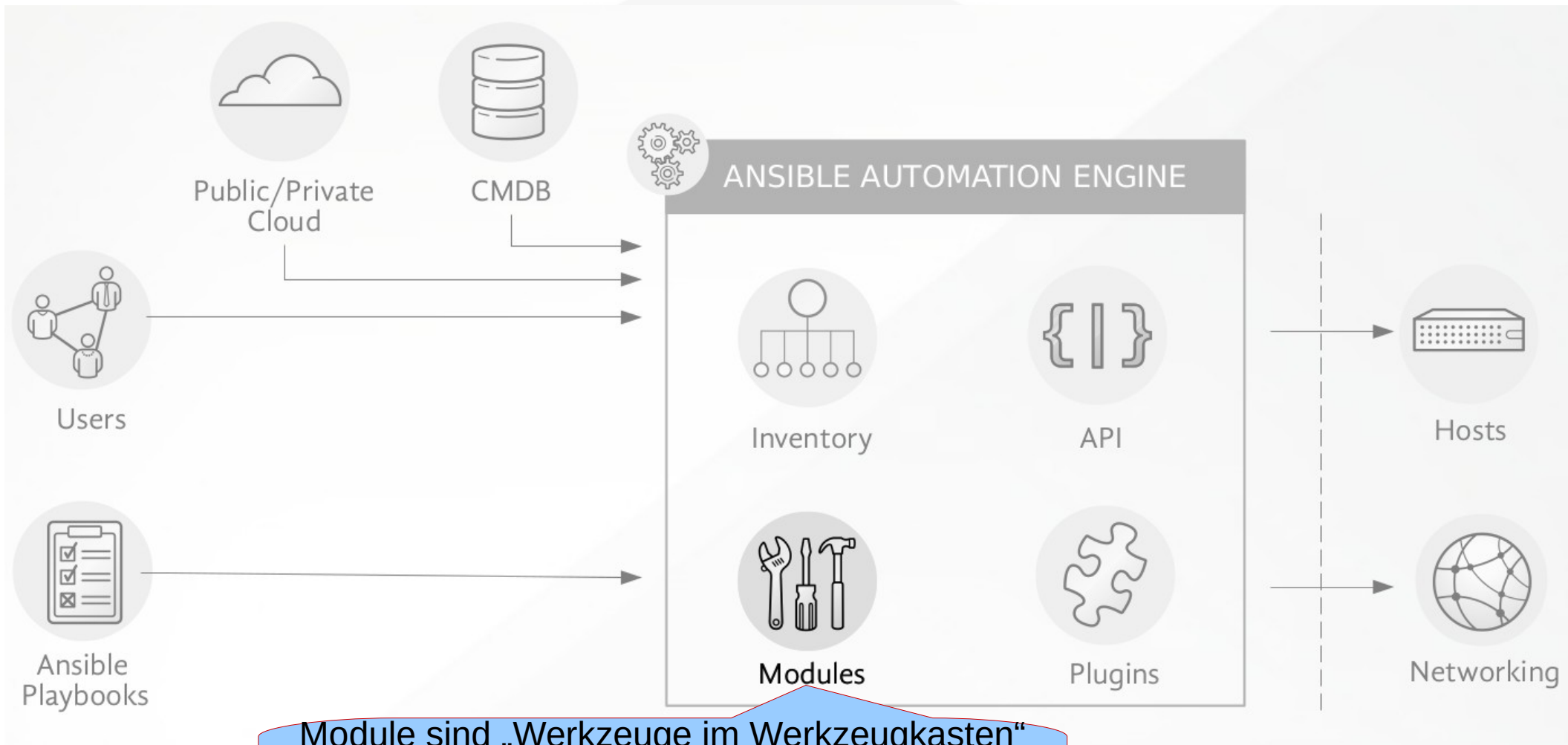
CFEngine



Wie Ansible funktioniert

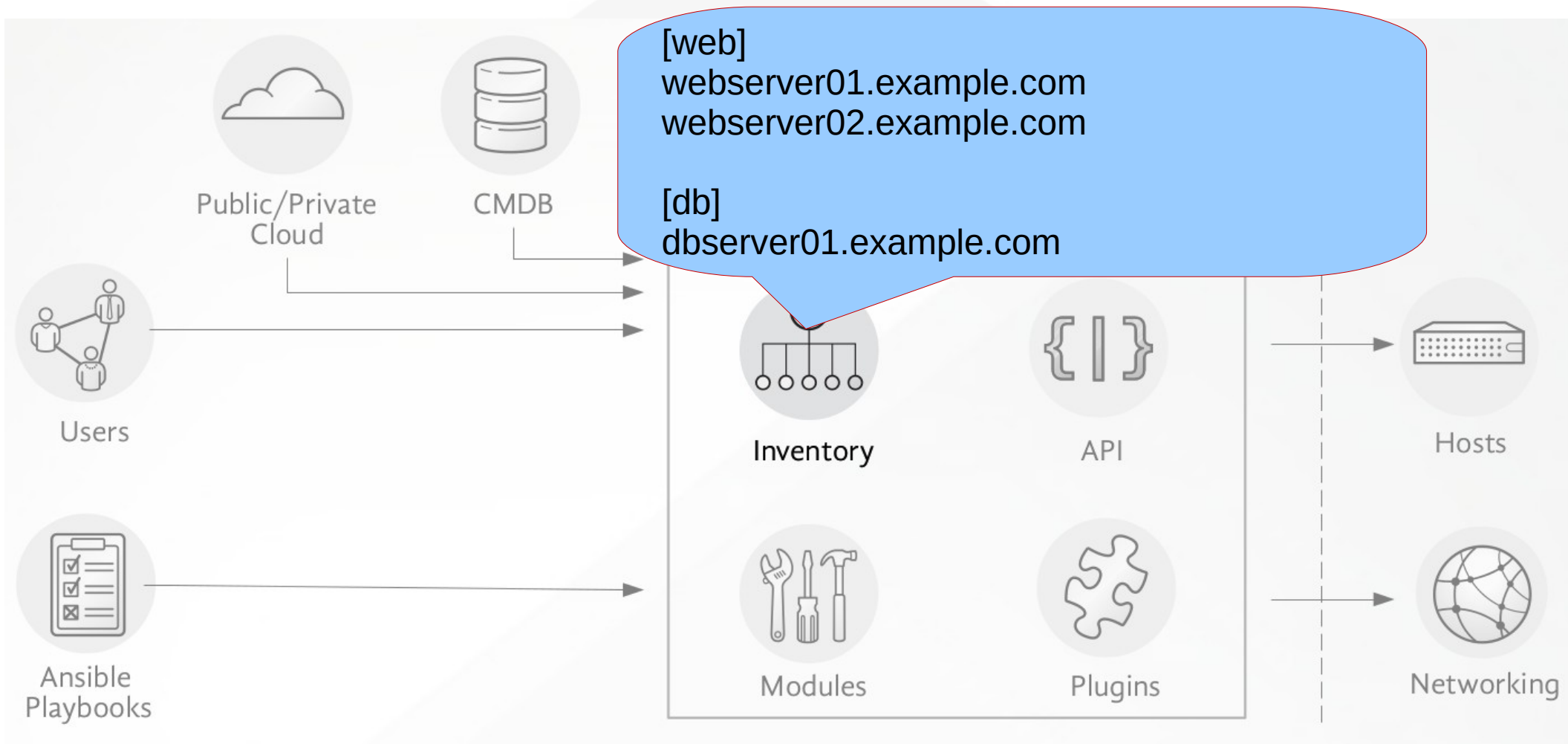


Wie Ansible funktioniert

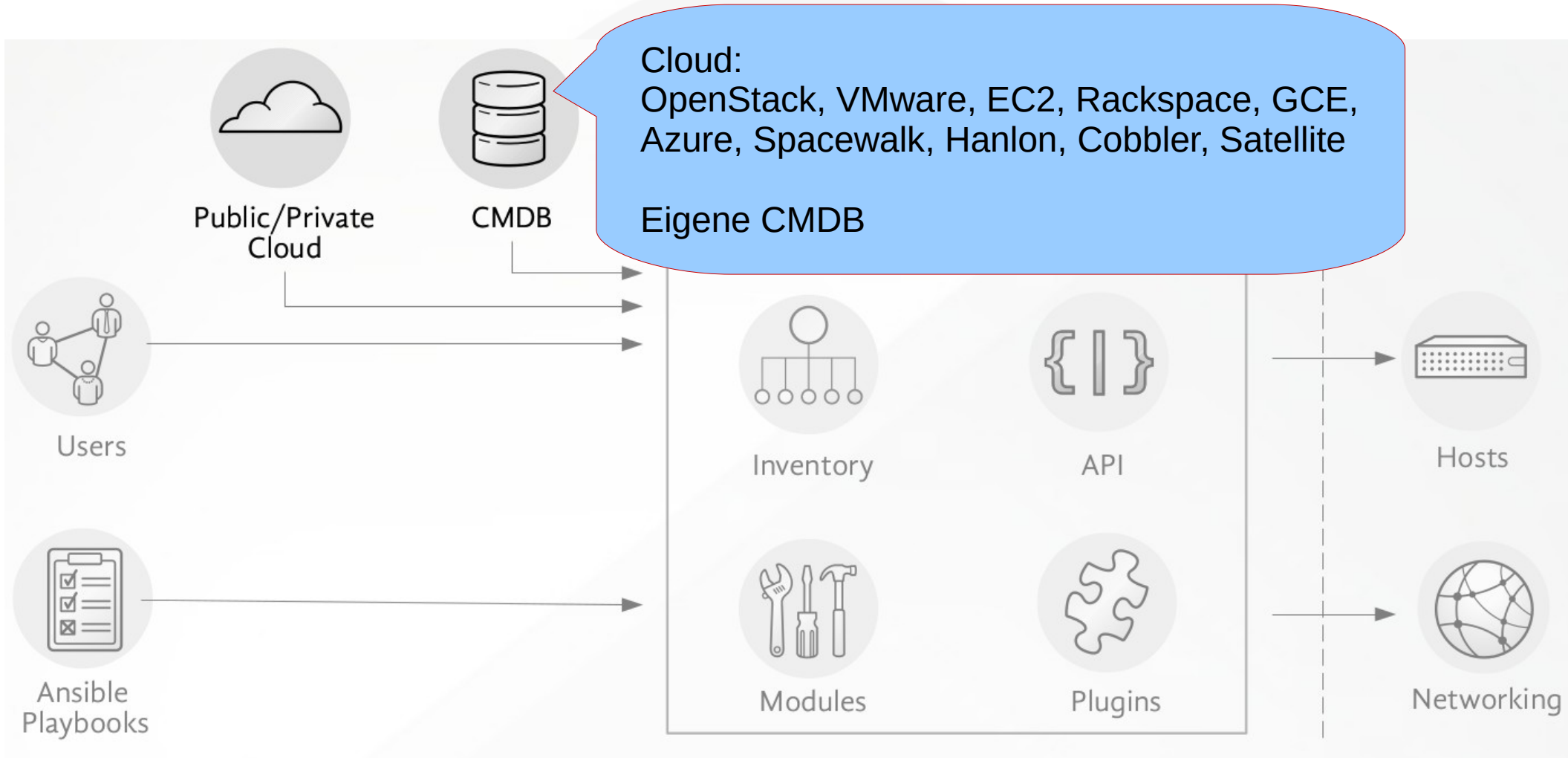


Module sind „Werkzeuge im Werkzeugkasten“
Python, Powershell oder jede andere Sprache
Erweiterung der Einfachheit von Ansible

Wie Ansible funktioniert



Wie Ansible funktioniert



Ad-Hoc Commands

- Uptime

```
$ ansible testserver -m command -a uptime
testserver | SUCCESS | rc=0 >>
08:58:17 up 44 min, 1 user, load average: 0.00, 0.02, 0.0
```

- Logs betrachten

```
$ ansible testserver -a "tail -1 /var/log/messages" -b
testserver | SUCCESS | rc=0 >>
Jan 27 09:01:01 localhost systemd: Starting user-0.slice.
```

- Paket installieren

```
$ ansible testserver -m yum -a "name=httpd state=installed" -b
```

- System-Informationen auslesen

```
$ ansible testserver -m setup
```

Playbooks

- name: Configure nginx webserver
hosts: webserver
become: true
tasks:
 - name: Enable epel repository
yum:
 - name: epel-release
 - state: present
 - name: Install nginx
yum:
 - name: nginx
 - state: present
 - name: Copy static.html
copy:
 - src: files/static.html
 - dest: /usr/share/nginx/html/static.html

Playbooks

```
$ ansible-playbook web-notls.yml
```

```
PLAY [Configure nginx webserver] *****
```

```
TASK [setup] *****
```

```
ok: [instance-1]
```

```
TASK [Enable epel repository] *****
```

```
ok: [instance-1]
```

```
TASK [Install nginx] *****
```

```
changed: [instance-1]
```

```
TASK [Copy index.html] *****
```

```
changed: [instance-1]
```

```
TASK [Start and enable nginx] *****
```

```
changed: [instance-1]
```

```
PLAY RECAP *****
```

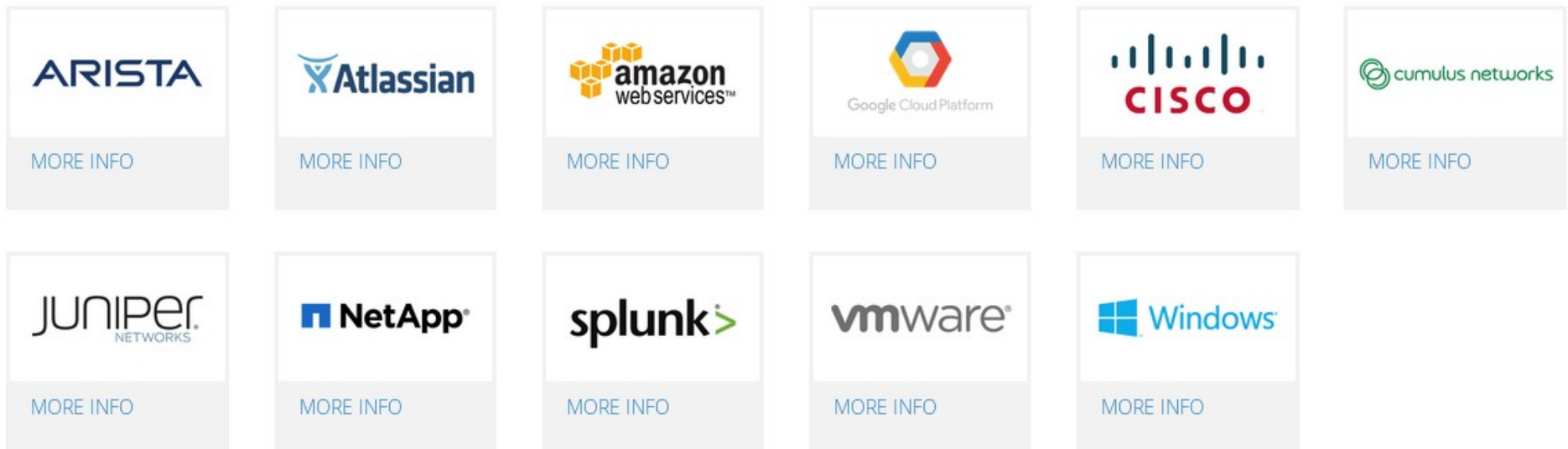
```
instance-1                : ok=9    changed=5    unreachable=0    failed=0
```

Module

https://docs.ansible.com/ansible/list_of_all_modules.html

- yum/apt/zypper/...
- service
- template
- user
- file
- copy

2.834 Module in
Ansible 2.8 inkludiert



Quelle: <https://www.ansible.com/how-ansible-works>



Ansible Tower

Ansible Tower

TOWER EMPOWERS TEAMS TO AUTOMATE

CONTROL

Scheduled and
centralized jobs

KNOWLEDGE

Visibility and compliance

DELEGATION

Role-based access
and self-service

SIMPLE

Everyone speaks the
same language

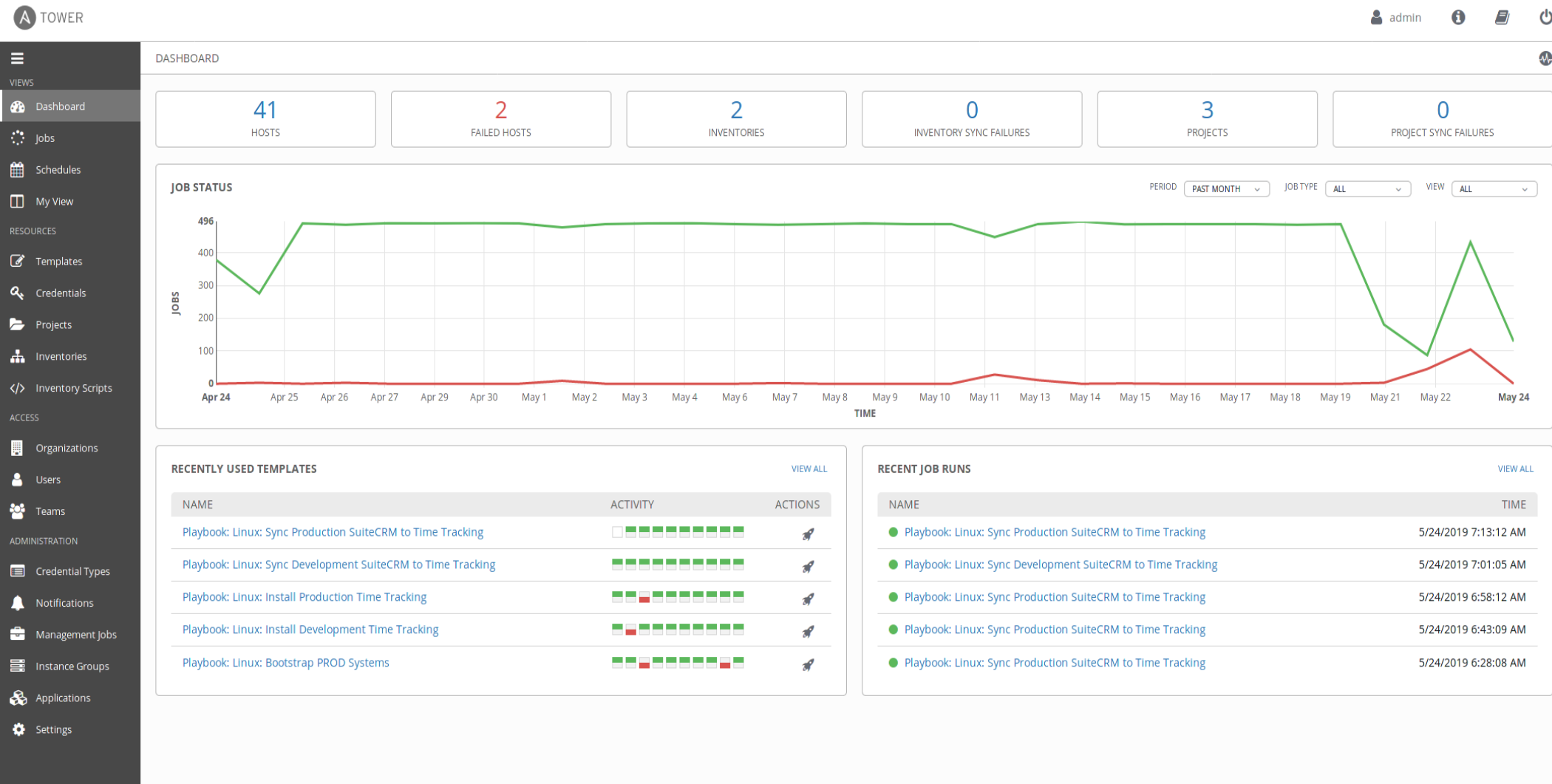
POWERFUL

Designed for
multi-tier deployments

AGENTLESS

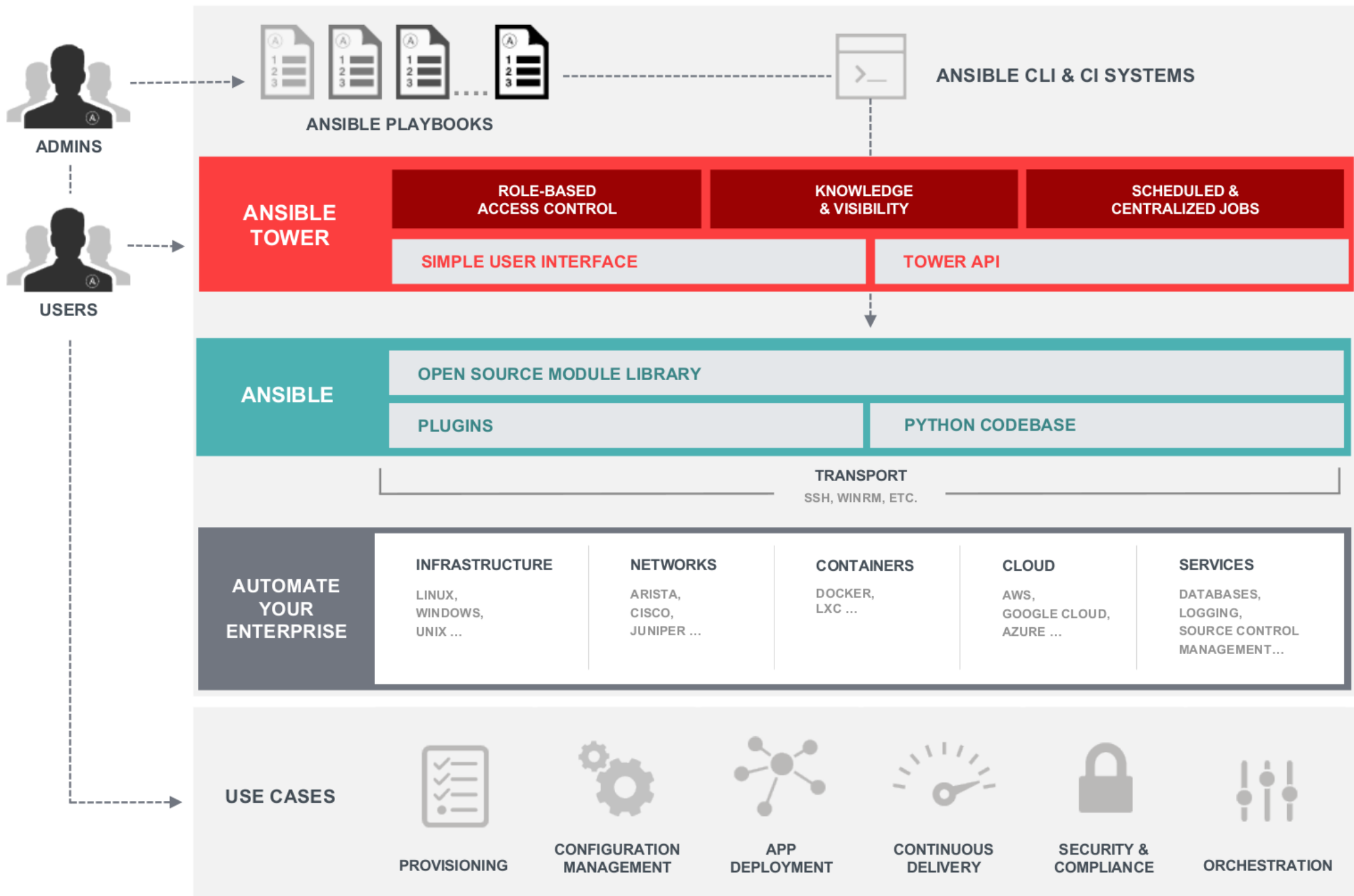
Predictable, reliable,
and secure

Ansible Tower



Ansible Tower

- Enterprise Framework zur Verwaltung, Kontrolle und Absicherung der Ansible Automatisierung via Web-UI und REST-API
 - Role based access control ermöglicht sichere Umgebungen und effiziente Teams
 - Unprivilegierte Benutzer können Applikationen auf Knopfdruck sicher ausrollen.
 - Alle Ansible Automatisierungs-Tasks sind zentral geloggt, komplett auditierbar und compliant.

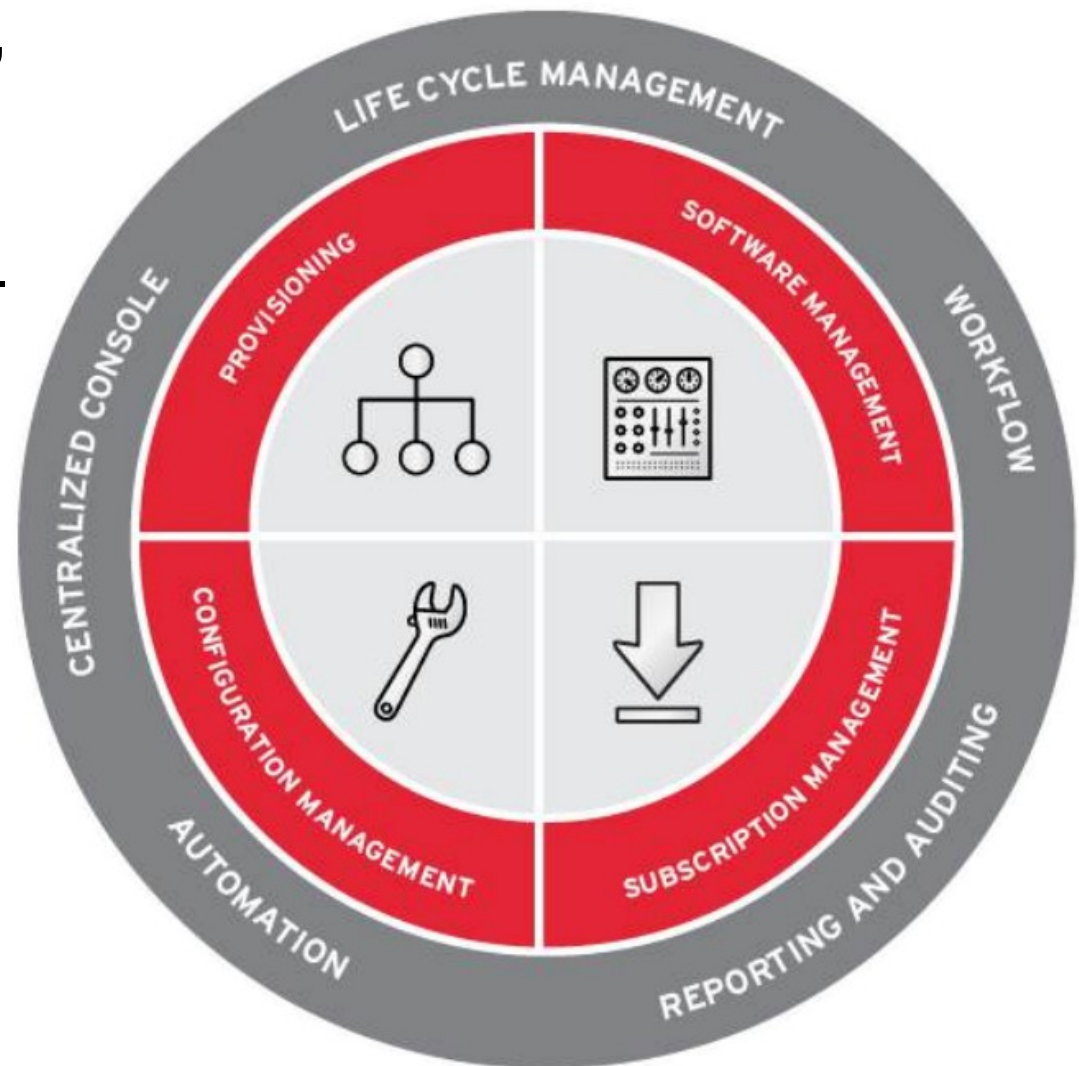


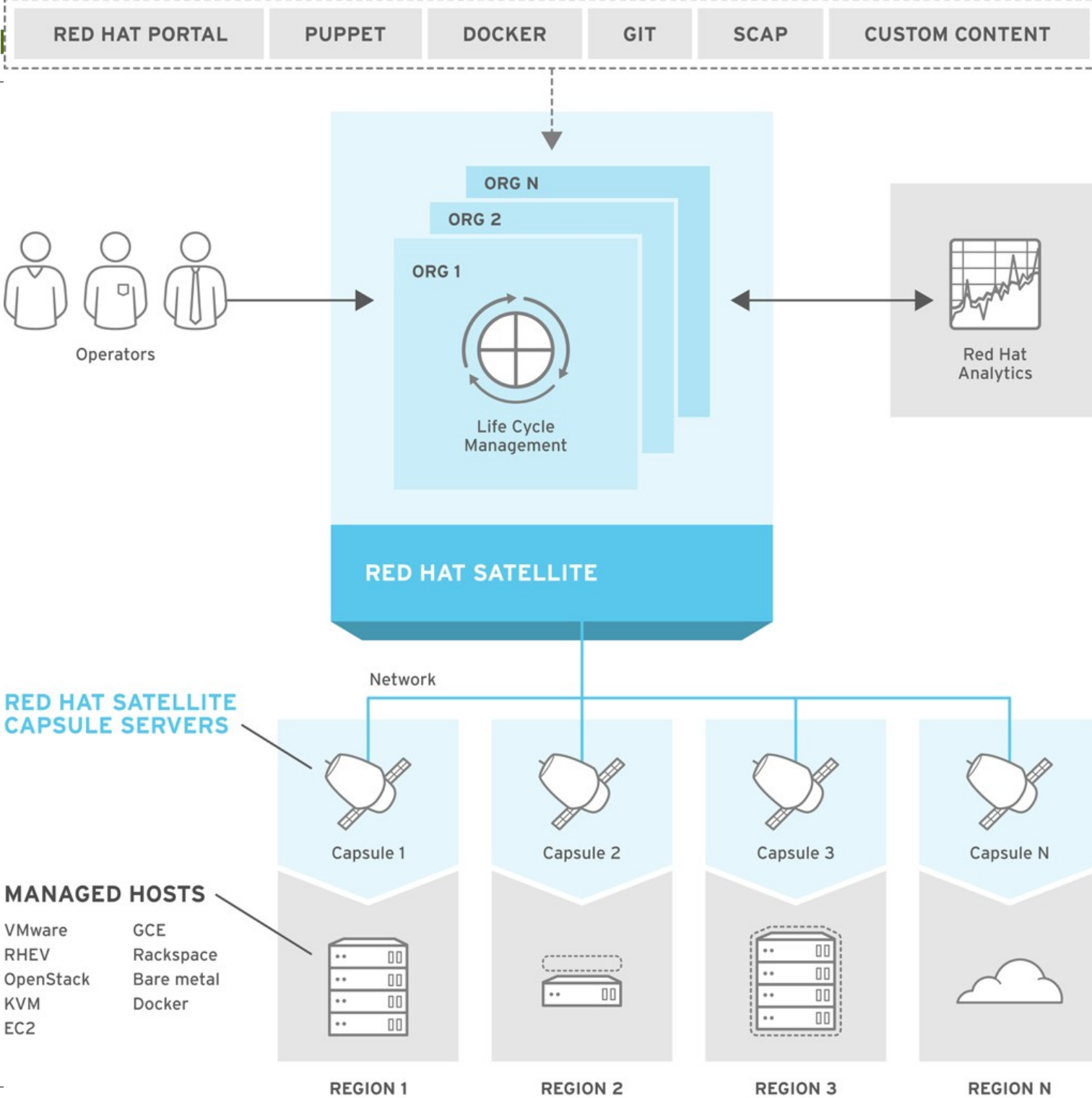


Red Hat Satellite

Red Hat Satellite

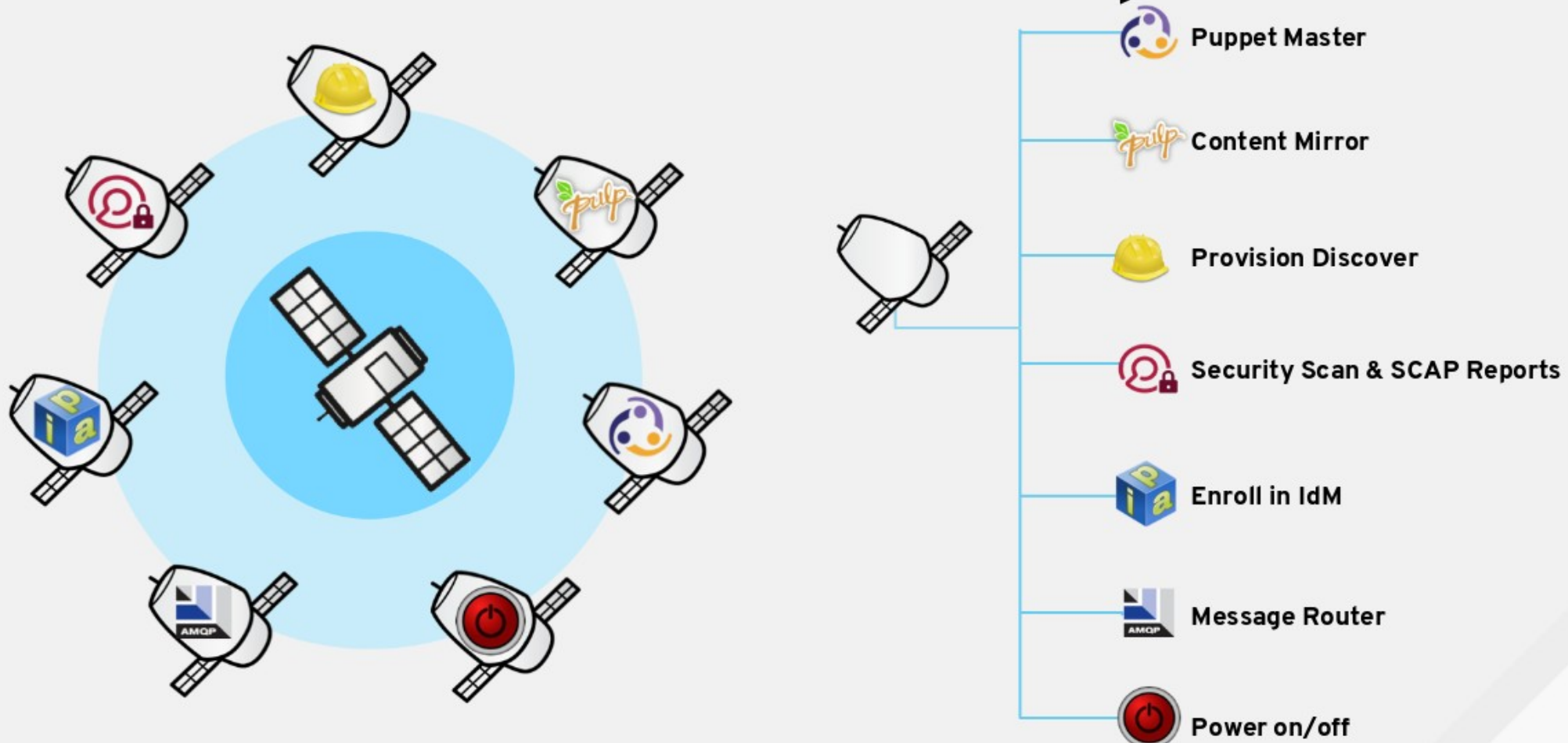
- Provisioning auf Bare Metal, Private und Öffentliche Clouds
- Deklaratives Konfigurations-Management
- Automatisierte Softwareverteilung
- Life Cycle Management
- Vereinfachtes Content Management
- Vereinheitlichtes Management
- Subskriptions Management



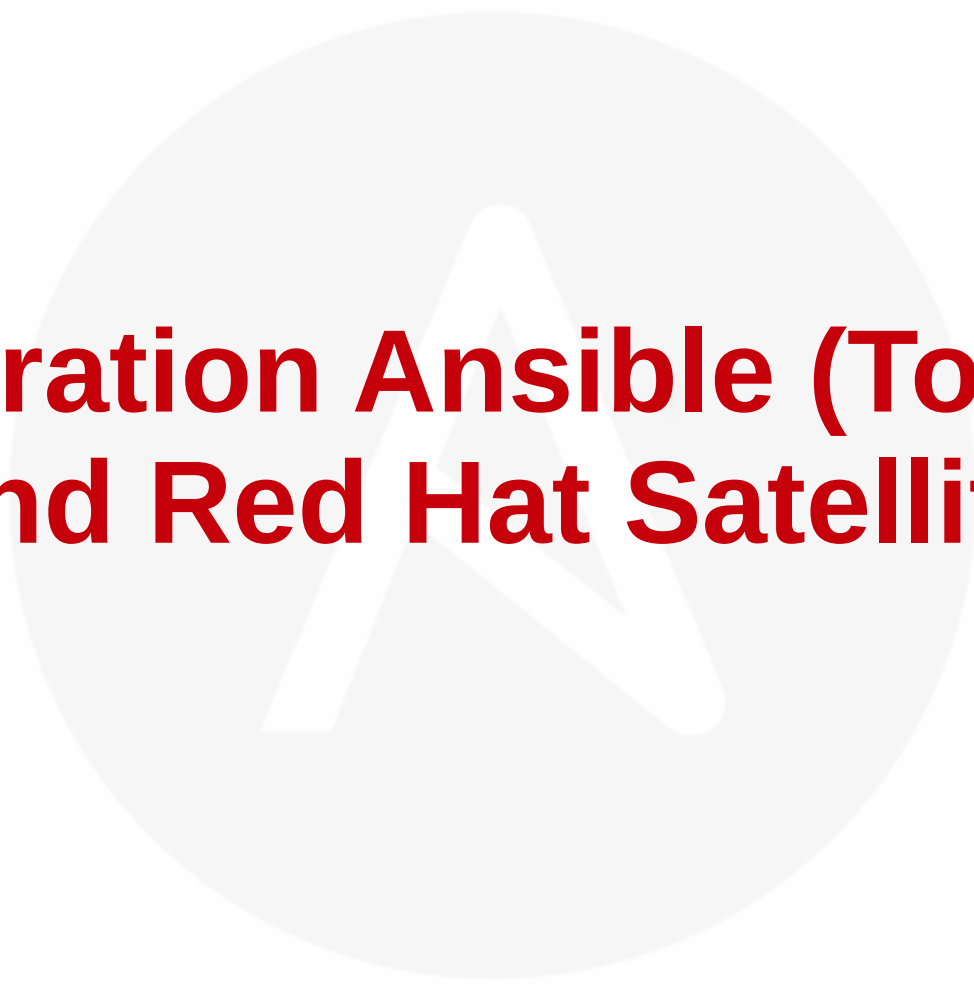


Red Hat Satellite

Federated Services with Capsules







Integration Ansible (Tower) und Red Hat Satellite

Dynamic Inventory

- Auslesen der Ansible Hosts von Red Hat Satellite inkl:
 - Hostgruppen
 - Lifecycle Environments
 - Locations
 - Content Views
 - Parameter
- Ansible Core via foreman.py Inventory Skript
- Im Ansible Tower bereits integriert
- Tipp: foreman.py von <https://github.com/ansible/ansible> für Tower verwenden, da mehr Funktionalität, aber unsupported

Puppet Class Parameters

Puppet Class	Name	Type	Value	Omit
Parent Parameters				
Name	Value			Actions
ansible_host_config_key				  Override
ansible_job_template_id		10		  Override
ansible_tower_fqdn				  Override
ansible_tower_provisioning		True		  Override
common_ntp_servers		[[{"server": "ntp1.example.com"}, {"server": "ntp2.example.com"}]]		  Override
icinga2_function		client		  Override
icinga2_host				  Override
icinga2_master				  Override
kt_activation_keys		prod - Red Hat Enterprise Linux 7 x86_64,prod - Red Hat Enterprise Linux 7 x86_64 - 3rd Party		  Override
satellite6_remove_repos		True		  Override

Host Group Parameters

Name	Value	Actions
apache_conf_dir	/opt/rh/httpd24/root/etc/httpd	   Remove
apache_document_root	/data/www	   Remove
apache_listen_port	443	   Remove
apache_pkg_name	["httpd24-httpd", "httpd24-mod_ssl"]	   Remove
apache_repo	rhel-server-rhsc1-7-rpms	   Remove

Provisioning

- Provisioning via Satellite:
 - Discovery von physikalischen Maschinen
 - Erstellen von virtuellen Maschinen / Cloud Instanzen
 - Booten via (i)PXE oder Bootdisk
 - Kickstart/PreSeed/AutoYAST-Installation
- Ausführen von Ansible Rollen oder
- Callback zu Ansible Tower

Provisioning

☰

Red Hat Satellite

Rene Koch IT

Any Location

🔔

Admin

Monitor

Content

Containers

Hosts

Configure

Infrastructure

Insights

Administer

Provisioning Templates » Edit ansible_tower_callback_script

Template Inputs Type Association History Locations Organizations Help

⚠ Warning! This template is locked. You may only change the associations. Please [clone](#) it to customize.

Name *

ansible_tower_callback_script

Default ⓘ

☒

Template *

Input Diff Preview Fullscreen

ruby

Default

```
kind: snippet
name: ansible_tower_callback_script
model: ProvisioningTemplate
snippet: true
- %>
#!/bin/sh

echo "Calling Ansible AWX/Tower provisioning callback..."
/usr/bin/curl -v -k -s --data "host_config_key=<%= host_param('ansible_host_config_key') %>" https://<%= host_param('ansible_tower_fqdn') %>/api/v2/job_templates/<%= h
echo "DONE"
```

Provisioning

TOWER

VIEW

Dashboard

Jobs

Schedules

My View

RESOURCE

Templates

Credentials

Projects

Inventories

Inventory Scripts

ACCESS

Organizations

Users

Teams

ADMINISTRATION

Credential Types

Notifications

Management Jobs

admin

TEMPLATES / Playbook: Linux: Bootstrap (custom filter)

Playbook: Linux: Bootstrap (custom filter)

DETAILS

PERMISSIONS

NOTIFICATIONS

COMPLETED JOBS

SCHEDULES

ADD SURVEY

* NAME

Playbook: Linux: Bootstrap (custom filter)

DESCRIPTION

* JOB TYPE ?

Run

PROMPT ON LAUNCH

* INVENTORY ?

Inventory: Satellite 6

PROMPT ON LAUNCH

* PROJECT ?

GIT: RK IT (@master)

* PLAYBOOK ?

linux_common.yml

CREDENTIAL ?

Linux: ansible Vault: Linux default

PROMPT ON LAUNCH

FORKS ?

20

LIMIT ?

PROMPT ON LAUNCH

* VERBOSITY ?

1 (Verbose)

PROMPT ON LAUNCH

JOB TAGS ?

PROMPT ON LAUNCH

SKIP TAGS ?

PROMPT ON LAUNCH

LABELS ?

INSTANCE GROUPS ?

JOB SLICING ?

1

SHOW CHANGES ?

ON

PROMPT ON LAUNCH

OPTIONS

Enable Privilege Escalation ?

Allow Provisioning Callbacks ?

Enable Concurrent Jobs ?

Use Fact Cache ?

PROVISIONING CALLBACK URL ?

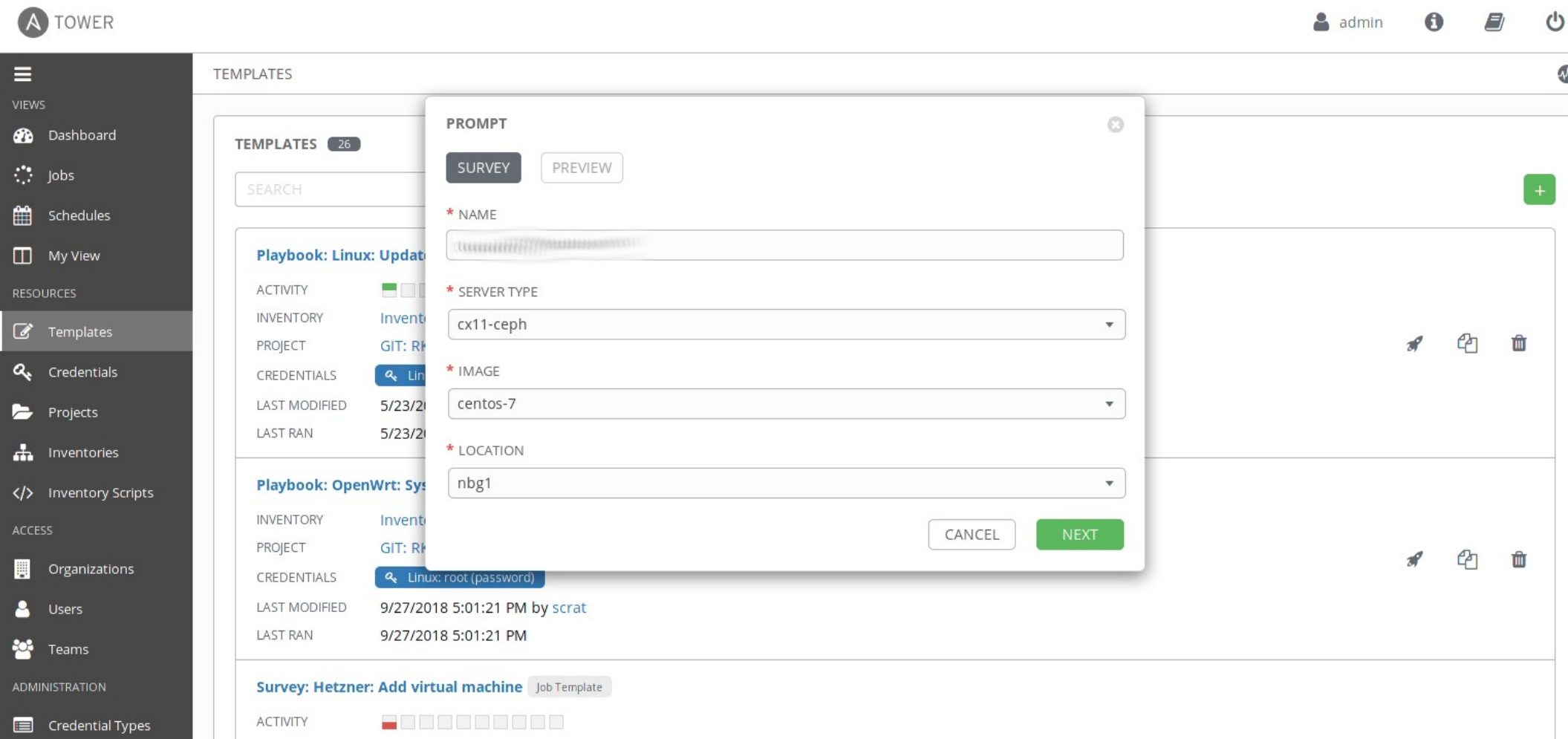
https://v2/job_templates/10/cal

* HOST CONFIG KEY ?

Provisioning

- Provisioning via Ansible Tower:
 - Eingabemaske für Parameter
 - Erstellen von virtuellen Maschinen / Cloud Instanzen
 - Klonen von Images oder
 - Provisioning via Satellite
 - Booten via (i)PXE oder Bootdisk
 - Kickstart/PreSeed/AutoYAST-Installation
- Ausführen von Ansible Playbooks nach dem Provisioning durch den Ansible Tower
 - u.a. Registrierung an Red Hat Satellite für Content Management

Provisioning



The screenshot shows the Ansible Tower web interface. The top navigation bar includes the 'TOWER' logo, a user profile 'admin', and several utility icons. The left sidebar contains navigation links for 'VIEWS' (Dashboard, Jobs, Schedules, My View) and 'RESOURCES' (Templates, Credentials, Projects, Inventories, Inventory Scripts). The main content area is titled 'TEMPLATES' and shows a list of templates with columns for 'ACTIVITY', 'INVENTORY', 'PROJECT', 'CREDENTIALS', 'LAST MODIFIED', and 'LAST RAN'. A modal window titled 'PROMPT' is open, allowing the user to create a new template. The modal has two tabs: 'SURVEY' (selected) and 'PREVIEW'. It contains four required fields: 'NAME' (text input), 'SERVER TYPE' (dropdown menu with 'cx11-ceph' selected), 'IMAGE' (dropdown menu with 'centos-7' selected), and 'LOCATION' (dropdown menu with 'nbg1' selected). At the bottom of the modal are 'CANCEL' and 'NEXT' buttons.

Provisioning

```
--  
- name: Install pre-built bootstrap RPM from {{ satellite6_fqdn }}  
  yum:  
    name: http://{{ satellite6_fqdn }}/pub/katello-ca-consumer-latest.noarch.rpm  
    state: "{{ satellite6_bootstrap_rpm_state }}"  
  notify:  
    - Refresh subscription-manager  
  when: inventory_hostname != satellite6_fqdn  
  
- name: Register host to Satellite 6  
  redhat_subscription:  
    state: "{{ satellite6_sm_state }}"  
    activationkey: "{{ satellite6_activationkey }}"  
    org_id: "{{ satellite6_org }}"  
    autosubscribe: "{{ satellite6_autosubscribe }}"  
  when: inventory_hostname != satellite6_fqdn  
  
- name: Register system to RHSM  
  redhat_subscription:  
    state: present  
    username: "{{ satellite6_rhsm_username }}"  
    password: "{{ satellite6_rhsm_password }}"  
    pool_ids: "{{ satellite6_rhsm_pool_ids }}"  
  when: inventory_hostname == satellite6_fqdn
```

Host Updates

- Satellite 6 ist Paketquelle für Red Hat und Custom/3rd Party RPMs
- Freigabe von Paketen via Content Views und Life Cycles
- Updaten über Satellite via:
 - Remote Execution Feature (SSH)
 - Katello Agent
- Updaten via Ansible Tower über Templates:
 - Paketquelle bleibt Red Hat Satellite
 - Weitere Möglichkeiten via Downtime im Monitoring, Funktionale Tests nach Update,...

Host Updates


Red Hat Satellite
Rene Koch IT
Any Location

Admin User

Monitor
Content
Containers
Hosts
Configure
Infrastructure
Insights
Administer

CCV Red Hat Enterprise Linux 7 x86_64
Publish New Version
Select Action

Content Views » CCV Red Hat Enterprise Linux 7 x86_64 » Versions

Details Versions Content Views History Tasks

Filter... Search

Version	Status	Environments	Content	Description	Actions
Version 20.0	Promoted to prod (2019-05-22 20:31:24 +0200)	prod Library dev qa	76569 Packages 8923 Errata (1150  4096  2000 )	added Satellite Tools 6.5	Promote
Version 19.0	Promoted to prod (2019-05-22 19:17:11 +0200)		76526 Packages 8922 Errata (1149  4096  2000 )	added Satellite Capsule 6.5, removed Satellite Capsule 6.4	Promote
Version 18.0	Promoted to prod (2019-05-22 16:54:28 +0200)		76553 Packages 8929 Errata (1150  4102  2000 )	Satellite Capsule 6.4.3 updated packages	Promote
Version 17.0	Promoted to prod (2019-03-04 12:42:29 +0100)		74910 Packages 8991 Errata (1095  4060  2073 )	removed CV RHV 4.2	Promote

100 per page
Showing 1 - 4 of 4
1 of 1


René Koch IT
Any Location
Admin User

Monitor
Content
Containers
Hosts
Configure
Infrastructure
Insights
Administer

Unregister Host

Content Hosts
Errata

Details
Provisioning Details
Subscriptions
Host Collections
Tasks
Packages
Errata
Module Streams
Traces
Repository Sets

Installable Errata

Current Lifecycle Envi

Search
Apply Selected
Recalculate

0 of 13 Selected

☐	Type	Id	Title	Issued
☐	✳ Bug Fix Advisory - Low	FEDORA-EPEL-2019-cd7d7b3e23	ansible-2.7.10-1.el7	5/22/19
☐	⚠ Security Advisory - Moderate	FEDORA-EPEL-2019-0911253d1e	python-django-1.11.20-1.el7	5/22/19
☐	➕ Product Enhancement Advisory - None	FEDORA-EPEL-2019-3893847a12	qpidd-cpp-1.39.0-1.el7	5/22/19
☐	✳ Bug Fix Advisory - None	FEDORA-EPEL-2019-b78513aba4	mosquitto-1.5.5-2.el7 libwebsockets-3.0.1-2.el7	5/22/19
☐	✳ Bug Fix Advisory - None	FEDORA-EPEL-2018-90207015c0	rubygem-rkerberos-0.1.3-6.el7	5/22/19
☐	✳ Bug Fix Advisory - Moderate	FEDORA-EPEL-2017-0579784f53	kobo-0.6.0-1.el7	5/22/19
☐	⚠ Security Advisory - None	FEDORA-EPEL-2016-42ecf5c111	v8-3.14.5.10-25.el7	5/22/19
☐	✳ Bug Fix Advisory - None	FEDORA-EPEL-2016-4c48bc7b08	rubygem-ffi-1.9.10-2.el7	5/22/19
☐	✳ Bug Fix Advisory - None	FEDORA-EPEL-2017-94499a3eb7	createrepo_c-0.10.0-6.el7	5/22/19
☐	✳ Bug Fix Advisory - None	FEDORA-EPEL-2015-7089	python-semantic_version-2.4.2-1.el7	5/22/19
☐	newpackage - None	FEDORA-EPEL-2016-3d23c34f99	passenger-4.0.53-4.el7	5/22/19
☐	⚠ Security Advisory - Important	FEDORA-EPEL-2019-dd9e038712	pagure-5.2-2.el7 python-amqp-2.4.0-1.el7 python-billiard-3.5.0.5-1.el7 python-celery-4.2.1-3.el7 python-kombu-4.2.2-1.el7 python-redis-2.10.6-1.el7 python-vine-1.2.0-1.el7	2/4/19
☐	newpackage	FEDORA-EPEL-2017-18b2d130e1	rubygem-sinatra-1.4.8-2.el7 rubygem-tilt-2.0.7-1.el7	4/30/17

100 per page
 Showing 1 - 13 of 13
1 of 1

Host Updates

 TOWER
 admin

VIEW

DASHBOARD

JOBS

SCHEDULES

MY VIEW

RESOURCES

TEMPLATES

CREDENTIALS

PROJECTS

INVENTORIES

INVENTORY SCRIPTS

ACCESS

ORGANIZATIONS

USERS

TEAMS

ADMINISTRATION

CREDENTIAL TYPES

NOTIFICATIONS

MANAGEMENT JOBS

INSTANCE GROUPS

JOBS / 93523 - Playbook: Linux: Update DEV Systems

DETAILS

STATUS

Successful

STARTED

5/23/2019 6:58:09 PM

FINISHED

5/23/2019 6:58:42 PM

JOB TEMPLATE

Playbook: Linux: Update DEV Systems

JOB TYPE

Run

LAUNCHED BY

admin

INVENTORY

Inventory: Satellite 6

PROJECT

GIT: RK IT (@master)

REVISION

f617cd5

PLAYBOOK

linux_updates.yml

CREDENTIAL

Linux: ansible

FORKS

10

LIMIT

foreman_lifecycle_environment_dev

VERBOSITY

1 (Verbose)

INSTANCE GROUP

tower

EXTRA VARIABLES

YAML JSON

EXPAND

1 ---

Playbook: Linux: Update DEV Systems

PLAYS 1

TASKS 2

HOSTS 5

ELAPSED 00:00:32

SEARCH

Q

KEY

2 Using /var/lib/awx/projects/_6_git_rk_it_master/ansible.cfg as config file

3 /tmp/awx_93523_6KMvEx/tmpXx9YCV did not meet host_list requirements, check plugin documentation if this is unexpected

4

5 PLAY [Update Linux hosts] ***** 18:58:23

6

7 TASK [Gathering Facts] ***** 18:58:23

8 ok: []

9 ok: []

10 ok: []

11 ok: []

12 ok: []

13

14 TASK [Install Linux updates] ***** 18:58:37

15 ok: [] => {"changed": false, "msg": "", "rc": 0, "results": ["Nothing to do here, all packages are up to date"]}

16 ok: [] => {"changed": false, "msg": "", "rc": 0, "results": ["Nothing to do here, all packages are up to date"]}

17 ok: [] => {"changed": false, "msg": "", "rc": 0, "results": ["Nothing to do here, all packages are up to date"]}

18 ok: [] => {"changed": false, "msg": "", "rc": 0, "results": ["Nothing to do here, all packages are up to date"]}

Rollen Testen mit Molecule

- Molecule ist ein Framework zum Testen von Ansible Rollen
- Kann Container benutzen (z.B. Podman oder Docker)
- Container können via Satellite freigegeben werden und versionisiert werden (Life Cycle)
- Zugriff von Molecule auf Docker Registry

Rollen Testen mit Molecule

☰

Red Hat Satellite

Rene Koch IT ▾ Any Location ▾

🔔

Admin User ▾

Monitor >

Content >

Containers >

Hosts >

Configure >

Infrastructure >

Insights >

Administer >

CV Container

Publish New Version Select Action ▾

Content Views > CV Container > Docker Repositories ⇄

Details Versions Yum Content ▾ Apt Repositories File Repositories Puppet Modules Container Images ▾ OSTree Content History Tasks

Docker Repository Selection

List/Remove Add

Filter... Search ▾

Remove Repositories

0 of 3 Selected

☐	Name	Product	Last Sync	Sync State	Content
☐	library/centos	Docker	2019-05-23 21:45:57 +0200	Success	104 Container Image Manifests 45 Container Image Manifest Lists 33 Container Image Tags
☐	rhel7-init	Docker	2019-05-23 21:45:58 +0200	Success	54 Container Image Manifests 10 Container Image Manifest Lists 23 Container Image Tags
☐	rhel8-beta/rhel-init	Docker	2019-05-23 21:45:55 +0200	Success	7 Container Image Manifests 1 Container Image Manifest Lists 3 Container Image Tags

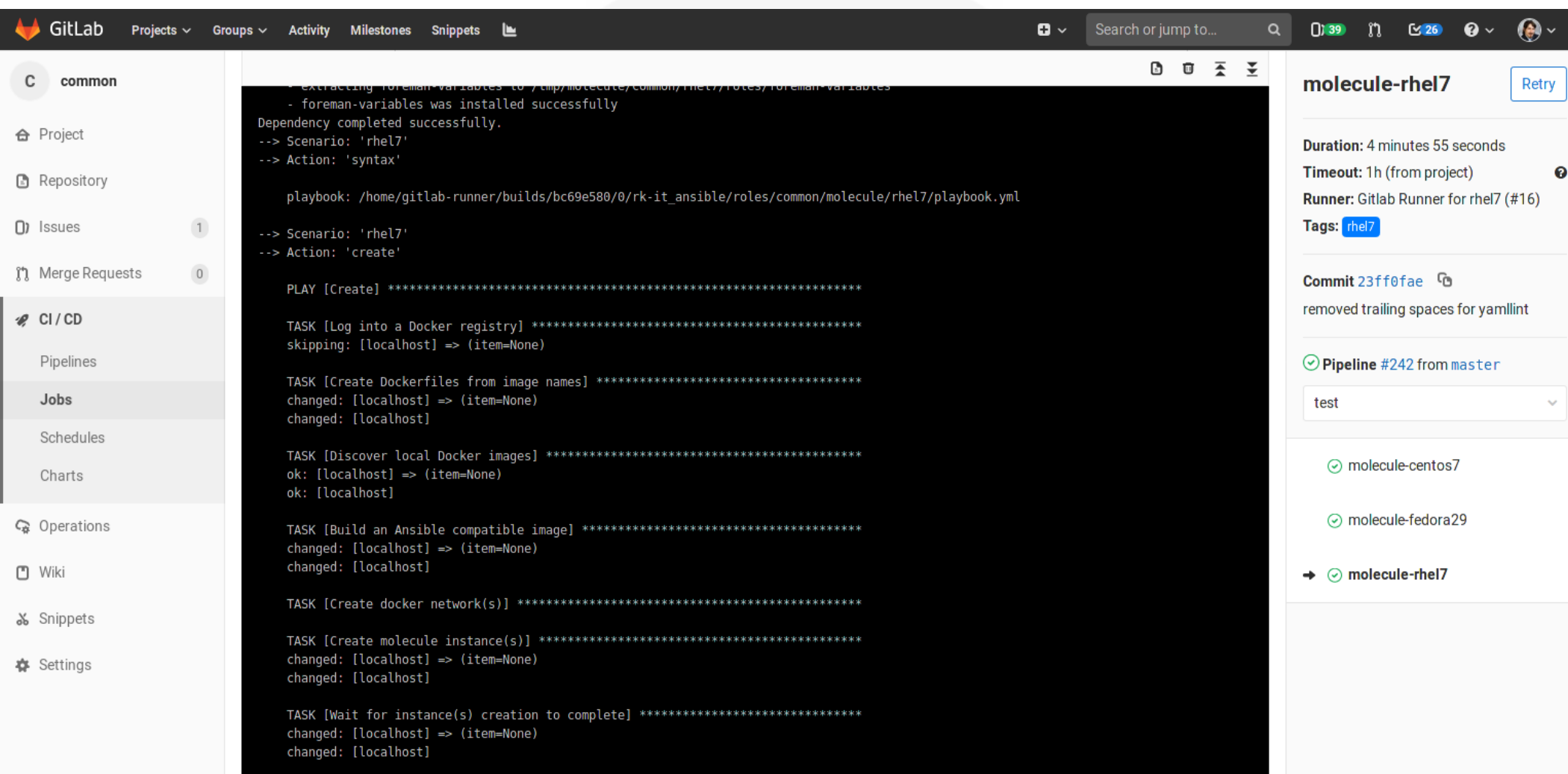
100 ▾ per page

Showing 1 - 3 of 3 << < 1 of 1 > >>

Rollen Testen mit Molecule

```
--  
dependency:  
  name: galaxy  
driver:  
  name: docker  
lint:  
  name: yamllint  
platforms:  
  - name: rhel7  
    image: "rk-it-prod-cv_container-docker-rhel7-init"  
    registry:  
      url: "https://registry-1.docker.io:5000"  
    privileged: true  
    volume_mounts:  
      - "/sys/fs/cgroup:/sys/fs/cgroup:rw"  
    command: "/usr/sbin/init"  
provisioner:  
  name: ansible  
  lint:  
    name: ansible-lint  
  options:  
    skip-tags: skip_molecule  
scenario:  
  name: rhel7  
verifier:  
  name: testinfra  
  lint:  
    name: flake8  
~
```

Rollen Testen mit Molecule



The screenshot displays the GitLab CI/CD interface. On the left, the sidebar shows the 'CI / CD' section with 'Jobs' selected. The main area shows the output of a Molecule test for the 'molecule-rhel7' job. The output indicates that the test was successful, with a duration of 4 minutes 55 seconds and a timeout of 1h. The runner used is 'Gitlab Runner for rhel7 (#16)'. The commit hash is '23ff0fae'. The pipeline is '#242 from master'. The job 'molecule-rhel7' is highlighted as successful.

```

- extracting foreman-variables to /tmp/molecule/common/rhel7/roles/foreman-variables
- foreman-variables was installed successfully
Dependency completed successfully.
--> Scenario: 'rhel7'
--> Action: 'syntax'

playbook: /home/gitlab-runner/builds/bc69e580/0/rk-it_ansible/roles/common/molecule/rhel7/playbook.yml

--> Scenario: 'rhel7'
--> Action: 'create'

PLAY [Create] *****

TASK [Log into a Docker registry] *****
skipping: [localhost] => (item=None)

TASK [Create Dockerfiles from image names] *****
changed: [localhost] => (item=None)
changed: [localhost]

TASK [Discover local Docker images] *****
ok: [localhost] => (item=None)
ok: [localhost]

TASK [Build an Ansible compatible image] *****
changed: [localhost] => (item=None)
changed: [localhost]

TASK [Create docker network(s)] *****

TASK [Create molecule instance(s)] *****
changed: [localhost] => (item=None)
changed: [localhost]

TASK [Wait for instance(s) creation to complete] *****
changed: [localhost] => (item=None)
changed: [localhost]
  
```

molecule-rhel7 [Retry](#)

Duration: 4 minutes 55 seconds
Timeout: 1h (from project)
Runner: Gitlab Runner for rhel7 (#16)
Tags: [rhel7](#)

Commit [23ff0fae](#) [🔗](#)
 removed trailing spaces for yamllint

✓ **Pipeline #242** from master

test

✓ molecule-centos7
 ✓ molecule-fedora29
 → ✓ **molecule-rhel7**

Compliance mit OpenSCAP

- Tools zum Definieren, Überwachen und Einhalten einer Security-Baselines.
- Policies definieren den einzuhaltenden Standard (z.B. PCI DSS, STIGs, USGCB,...)
- Policies können angepasst werden
- Ein SCAP-Workflow kann folgendermaßen aussehen:
 - Konfiguration von SCAP Daemon via Ansible
 - Holen der SCAP Policies vom Red Hat Satellite
 - Scheduling via Cron und Senden der Ergebnisse an den Red Hat Satellite
 - Compliance-Anpassungen via Ansible

Compliance mit OpenSCAP

 Red Hat Satellite

Rene Koch IT
 Any Location

 Admin User

Monitor

Content

Containers

Hosts

Configure

Infrastructure

Insights

Administer

Compliance Reports

Show log messages:

All messages

Back

Delete

Host details

View full report

Download XML in bzip

Download HTML

Reported at 19 May 03:32 for policy Standard System Security Profile for Red Hat Enterprise Linux 7 through

Severity	Message	Resource	Result	Actions
Unknown	Disable At Service (atd)	xccdf_org.ssgproject.content_rule_service_atd_disabled	pass	Hosts failing this rule
Unknown	Disable Network Router Discovery Daemon (rdisc)	xccdf_org.ssgproject.content_rule_service_rdisc_disabled	pass	Hosts failing this rule
Unknown	Disable Automatic Bug Reporting Tool (abrt)	xccdf_org.ssgproject.content_rule_service_abrt_disabled	pass	Hosts failing this rule
Unknown	Disable Odd Job Daemon (oddjobd)	xccdf_org.ssgproject.content_rule_service_oddjobd_disabled	pass	Hosts failing this rule
Unknown	Disable ntpdate Service (ntpdate)	xccdf_org.ssgproject.content_rule_service_ntpdate_disabled	pass	Hosts failing this rule
Unknown	Disable Apache Qpid (qpidd)	xccdf_org.ssgproject.content_rule_service_qpidd_disabled	pass	Hosts failing this rule
Medium	Disable the Automounter	xccdf_org.ssgproject.content_rule_service_autofs_disabled	pass	Hosts failing this rule
Unknown	Add nodev Option to /dev/shm	xccdf_org.ssgproject.content_rule_mount_option_dev_shm_nodev	pass	Hosts failing this rule
Unknown	Add nosuid Option to /dev/shm	xccdf_org.ssgproject.content_rule_mount_option_dev_shm_nosuid	pass	Hosts failing this rule
Unknown	Ensure All SUID Executables Are Authorized	xccdf_org.ssgproject.content_rule_file_permissions_unauthorized_suid	fail	Hosts failing this rule
Unknown	Verify that All World-Writable Directories Have Sticky Bits Set	xccdf_org.ssgproject.content_rule_dir_perms_world_writable_sticky_bits	pass	Hosts failing this rule
Medium	Ensure No World-Writable Files Exist	xccdf_org.ssgproject.content_rule_file_permissions_unauthorized_world_writable	pass	Hosts failing this rule
Unknown	Ensure All SGID Executables Are Authorized	xccdf_org.ssgproject.content_rule_file_permissions_unauthorized_sgid	fail	Hosts failing this rule

Checklist scap_org.open-scap_datastream_from_xccdf_ssg-rhel7-xccdf-1.2.xml / scap_org.open-scap_cref_ssg-rhel7-xccdf-1.2.xml

Title **Guide to the Secure Configuration of Red Hat Enterprise Linux 7**

Customization None selected

Profile Standard System Security Profile for Red Hat Enterprise Linux 7 (51) Customize

Target ☒ Local Machine ☐ Remote Machine (over SSH)

Rules

Expand all

- ▶ Disable Apache Qpid (qpidd)
- ▶ Disable Network Router Discovery Daemon (rdisc)
- ▶ Disable Automatic Bug Reporting Tool (abrt)
- ▶ Disable ntpdate Service (ntpdate)
- ▶ Disable Odd Job Daemon (oddjobd)
- ▶ Disable At Service (atd)
- ▶ Disable the Automounter
- ▶ Add nosuid Option to /dev/shm
- ▶ Add nodev Option to /dev/shm
- ▶ Ensure No World-Writable Files Exist
- ▶ Verify that All World-Writable Directories Have Sticky Bits Set
- ▶ Ensure All SGID Executables Are Authorized
- ▼ Ensure All SUID Executables Are Authorized

The SUID (set user id) bit should be set only on files that were installed via authorized means. A straightforward means of identifying unauthorized SGID files is determine if any were not installed as part of an RPM package, which is cryptographically verified. Investigate the origin of any unpackaged SUID files.

- ▶ Verify File Hashes with RPM
- ▶ Verify and Correct File Permissions with RPM
- ▶ Ensure /var/log/audit Located On Separate Partition

0% (0 results, 51 rules selected)

Generate remediation role

☐ Dry run ☐ Fetch remote resources ☐ Remediate

Scan

bash
ansible

Scheduling von Ansible Runs

- Satellite 6 kann Ansible Runs schedulen
 - Limitiert auf Ausführen von Rollen und im Satellite registrierte Systeme
 - Notifizierung via E-Mail
- Ansible Tower hat Scheduling-Engine
 - Für Ansible Systeme (Linux/Windows/Switches/Storages/...)
 - Notifizierung via E-Mail, Slack, IRC, HipChat, Webhook,...
 - Scheduling von Playbooks, Check-Runs, Workflows
- Ziel: Sicherstellen, dass Standards eingehalten werden

Scheduling von Ansible Runs

TOWER admin ? 📄 🔌

SCHEDULES

SCHEDULED JOBS 11

SEARCH 🔍 KEY

NAME	TYPE	NEXT RUN	ACTIONS
☒ Schedule: Playbook: Linux: Bootstrap DEV Systems (00 23 ***)	Playbook Run	5/24/2019 11:00:00 PM	✎ 🗑
☒ Schedule: Playbook: Linux: Bootstrap QA Systems (10 23 ***)	Playbook Run	5/24/2019 11:10:00 PM	✎ 🗑
☒ Schedule: Playbook: Linux: Bootstrap PROD Systems (40 23 ***)	Playbook Run	5/24/2019 11:40:00 PM	✎ 🗑
☒ Schedule: Playbook: Linux: Install Development Time Tracking (00 02 ***)	Playbook Run	5/25/2019 2:00:00 AM	✎ 🗑
☒ Schedule: Playbook: Linux: Sync Development SuiteCRM to Time Tracking (00 ***)	Playbook Run	5/24/2019 10:00:00 AM	✎ 🗑
☒ Schedule: Playbook: Linux: Satellite 6 Repository Sync (03 19 ***)	Playbook Run	5/24/2019 7:03:00 PM	✎ 🗑

Job History

- Wichtig im Unternehmens-Umfeld ist das Nachweisen wann wurde was geändert
- History wird von Red Hat Satellite und Ansible Tower geführt

 Red Hat Satellite

Rene Koch IT
 Any Location


 Admin User

 Monitor

 Content

 Containers

 Hosts

 Configure

 Infrastructure

Job invocations

Description	Search Query	Status	Succeeded	Failed	Pending	Total hosts	Start
Run ansible roles	name ^ (loki...	failed	0	1	0	1	2 months ago
Run ansible roles	name ^ (loki...	failed	0	1	0	1	2 months ago
Run ansible roles	name ^ (loki...	failed	0	1	0	1	2 months ago
Run ansible roles	name ^ (loki...	failed	0	1	0	1	2 months ago
Run ansible roles	name ^ (comp...	failed	0	1	0	1	6 months ago

100 per page

1-5 of 5

1 of 1

Job History

 TOWER

 admin
 



VIEWS

Dashboard

Jobs

Schedules

My View

RESOURCES

Templates

Credentials

Projects

Inventories

Inventory Scripts

ACCESS

Organizations

Users

Teams

ADMINISTRATION

Credential Types

Notifications

Management Jobs

Instance Groups

JOBS

36965

SEARCH

Q

KEY

93894 - Playbook: Linux: Sync Production SuiteCRM to Time Tracking

Playbook Run

STARTED 5/24/2019 9:27:42 AM

FINISHED 5/24/2019 9:28:23 AM

JOB TEMPLATE

Playbook: Linux: Sync Production SuiteCRM to Time Tracking

INVENTORY

Inventory: Satellite 6

PROJECT

GIT: RK IT (@production)

CREDENTIALS

Linux: ansible

Vault: Linux default

93896 - Inventory: Satellite 6 - Source:

Inventory Sync

STARTED 5/24/2019 9:26:28 AM

FINISHED 5/24/2019 9:27:41 AM

INVENTORY

Inventory: Satellite 6

93895 - GIT: RK IT (@production)

SCM Update

STARTED 5/24/2019 9:26:28 AM

FINISHED 5/24/2019 9:26:36 AM

PROJECT

GIT: RK IT (@production)

93889 - Playbook: Linux: Sync Production SuiteCRM to Time Tracking

Playbook Run

STARTED 5/24/2019 9:12:42 AM

FINISHED 5/24/2019 9:13:22 AM

JOB TEMPLATE

Playbook: Linux: Sync Production SuiteCRM to Time Tracking

INVENTORY

Inventory: Satellite 6

PROJECT

GIT: RK IT (@production)

CREDENTIALS

Linux: ansible

Vault: Linux default

- VIEW
- Dashboard
- Jobs
- Schedules
- My View
- RESOURCES
- Templates
- Credentials
- Projects
- Inventories
- Inventory Scripts
- ACCESS
- Organizations
- Users
- Teams
- ADMINISTRATION
- Credential Types
- Notifications
- Management Jobs
- Instance Groups
- Applications
- Settings

JOBS / 93576 - Playbook: Linux: Bootstrap DEV Systems

DETAILS

STATUS ● Successful

STARTED 5/23/2019 8:16:06 PM

FINISHED 5/23/2019 8:21:17 PM

JOB TEMPLATE [Playbook: Linux: Bootstrap DEV Systems](#)

JOB TYPE Run

LAUNCHED BY admin

INVENTORY [Inventory: Satellite 6](#)

PROJECT ● GIT: RK IT (@master)

REVISION 789e76a 📄

PLAYBOOK linux_common.yml

CREDENTIAL 🔍 Linux: ansible 🔍 Vault: Linux default

FORKS 20

LIMIT foreman_lifecycle_environment_dev

VERBOSITY 1 (Verbose)

INSTANCE GROUP tower

EXTRA VARIABLES ? YAML JSON EXPAND

1	---
---	-----

Playbook: Linux: Bootstrap DEV Systems

PLAYS 1 TASKS 113 HOSTS 5 ELAPSED 00:05:10 📄 ✖

SEARCH

Q KEY

changed:true ✖ CLEAR ALL

```

...
1181
1182 RUNNING HANDLER [linux-telegraf : Reload telegraf] ***** 20:21:13
*****
1183 changed: [1183] => {"changed": true, "name": "telegraf",
"state": "started", "status": {"ActiveEnterTimestamp": "Thu 2019-05-23 18:20:
52 UTC", "ActiveEnterTimestampMonotonic": "95681420148", "ActiveExitTimestamp
Monotonic": "0", "ActiveState": "active", "After": "network.target system.sli
ce systemd-journald.socket basic.target", "AllowIsolate": "no", "AmbientCapab
ilities": "0", "AssertResult": "yes", "AssertTimestamp": "Thu 2019-05-23 18:2
0:52 UTC", "AssertTimestampMonotonic": "95681419204", "Before": "multi-user.t
arget shutdown.target", "BlockIOAccounting": "no", "BlockIOWeight": "18446744
073709551615", "CPUAccounting": "no", "CPUQuotaPerSecUsec": "infinity", "CPUS
chedulingPolicy": "0", "CPUSchedulingPriority": "0", "CPUSchedulingResetOnFor
k": "no", "CPUShares": "18446744073709551615", "CanIsolate": "no", "CanReload
": "yes", "CanStart": "yes", "CanStop": "yes", "CapabilityBoundingSet": "1844
6744073709551615", "ConditionResult": "yes", "ConditionTimestamp": "Thu 2019-
05-23 18:20:52...
1184
1185 PLAY RECAP ***** 20:21:13
*****
1186 [1186] : ok=57 changed=0 unreachable=0 fail
ed=0
1187 [1187] : ok=57 changed=0 unreachable=0 fail
ed=0
1188 [1188] : ok=57 changed=6 unreachable=0 fail
ed=0

```

Git Checkouts

- Ansible Tower automatisiert Git Checkouts durch:
 - Unterstützung für Branches
 - Sicheres Speichern von Credentials
 - Git Checkouts wenn Playbook gestartet wird sowie Caching
- Red Hat Satellite hat (zum Stand heute) keine Ansible-Git-Integration, muss manuell gemacht werden

Git Checkouts

 TOWER
 admin

VIEWS

Dashboard

Jobs

Schedules

My View

RESOURCES

Templates

Credentials

Projects

Inventories

Inventory Scripts

ACCESS

Organizations

Users

Teams

PROJECTS / GIT: RK IT (@master)

GIT: RK IT (@master)

DETAILS

PERMISSIONS

NOTIFICATIONS

JOB TEMPLATES

SCHEDULES

* NAME

GIT: RK IT (@master)

DESCRIPTION

just for development systems

* ORGANIZATION

Q RK-IT

* SCM TYPE

Git

SOURCE DETAILS

* SCM URL ?

ssh://git@

SCM BRANCH/TAG/COMMIT

master

SCM CREDENTIAL

Q GIT: ansible

SCM UPDATE OPTIONS

☒ Clean ?

☒ Delete on Update ?

☒ Update Revision on Launch ?

CACHE TIMEOUT (SECONDS) ?

300

CANCEL

SAVE

51/63

Secure Linux Administration Conference 05/2019



Git Checkouts

 TOWER
 admin

VIEWS

Dashboard

Jobs

Schedules

My View

RESOURCES

Templates

Credentials

Projects

Inventories

Inventory Scripts

ACCESS

Organizations

Users

Teams

ADMINISTRATION

Credential Types

TEMPLATES / Playbook: Linux: Bootstrap PROD Systems

Playbook: Linux: Bootstrap PROD Systems

DETAILS

PERMISSIONS

NOTIFICATIONS

COMPLETED JOBS

SCHEDULES

ADD SURVEY

* NAME

Playbook: Linux: Bootstrap PROD Systems

DESCRIPTION

* JOB TYPE ?

Run

PROMPT ON LAUNCH

* INVENTORY ?

Inventory: Satellite 6

PROMPT ON LAUNCH

* PROJECT ?

GIT: RK IT (@production)

* PLAYBOOK ?

linux_common.yml

CREDENTIAL ?

Linux: ansible Vault: Linux default

PROMPT ON LAUNCH

FORKS ?

20

LIMIT ?

foreman_lifecycle_environment_prod

PROMPT ON LAUNCH

* VERBOSITY ?

1 (Verbose)

PROMPT ON LAUNCH

JOB TAGS ?

PROMPT ON LAUNCH

SKIP TAGS ?

PROMPT ON LAUNCH

LABELS ?

INSTANCE GROUPS ?

JOB SLICING ?

1

SHOW CHANGES ?

ON

PROMPT ON LAUNCH

OPTIONS

Enable Privilege Escalation ?

Allow Provisioning Callbacks ?

Enable Concurrent Jobs ?

Use Fact Cache ?

PROVISIONING CALLBACK URL ?

https://.../43/api/v2/job_templates/28/cal

Passwörter und Privileges

- Ein Problem von Ansible Core ist, dass Zugriff auf SSH-Keys, API-Credentials, Vault-Passwörter,... irgendwo hinterlegt werden müssen oder interaktiv eingegeben werden müssen.
- Ansible Tower kann Credentials sicher speichern
- Benutzer können Jobs starten ohne Credentials zu wissen bzw. sich auf einem System einloggen zu können
- Rechte-Management für alle Tower-Elemente
- Use Case: Deployment von Web-Anwendung in AWS-Cloud von Applikations-Verantwortlichen
 - Tower hat Cloud-Credentials und SSH-Key hinterlegt
 - App-Verantwortlicher benötigt nur SSH-Zugriff mit wenigen sudo-Rechten für operative Tätigkeiten (wenn überhaupt)

Passwörter und Privileges

TOWER

admin



CREDENTIALS / EDIT CREDENTIAL

Linux: ansible

DETAILS

PERMISSIONS

* NAME ?

Linux: ansible

DESCRIPTION ?

ansible user

ORGANIZATION

RK-IT

* CREDENTIAL TYPE ?

Machine

TYPE DETAILS

USERNAME

ansible

PW

PASSWORD

SHOW

☐ Prompt on launch

PW

SSH PRIVATE KEY

REPLACE

ENCRYPTED

PRIVATE KEY PASSPHRASE

☐ Prompt on launch

SHOW

PW

PRIVILEGE ESCALATION METHOD ?

sudo

PRIVILEGE ESCALATION USERNAME

PRIVILEGE ESCALATION PASSWORD

☐ Prompt on launch

SHOW

PW

CANCEL

SAVE

Passwörter und Privileges

 TOWER

admin
 



TEMPLATES / Playbook: Linux: Bootstrap (custom filter)

Playbook: Linux: Bootstrap (custom filter)

DETAILS
 PERMISSIONS
 NOTIFICATIONS
 COMPLETED JOBS
 SCHEDULES
 ADD SURVEY

* NAME

DESCRIPTION

* JOB TYPE ? ☐ PROMPT ON LAUNCH

Run

* INVENTORY ? ☐ PROMPT ON LAUNCH

* PROJECT ?

* PLAYBOOK ?

linux_common.yml

CREDENTIAL ? ☐ PROMPT ON LAUNCH

FORKS ?

20

LIMIT ? ☒ PROMPT ON LAUNCH

* VERBOSITY ? ☐ PROMPT ON LAUNCH

1 (Verbose)

JOB TAGS ? ☐ PROMPT ON LAUNCH

SKIP TAGS ? ☐ PROMPT ON LAUNCH

LABELS ?

INSTANCE GROUPS ?

JOB SLICING ?

1

SHOW CHANGES ? ☐ PROMPT ON LAUNCH

ON

OPTIONS

☐ Enable Privilege Escalation ?
 ☒ Allow Provisioning Callbacks ?
 ☐ Enable Concurrent Jobs ?
 ☐ Use Fact Cache ?

PROVISIONING CALLBACK URL ?

http://192.168.1.100:443/api/v2/job_templates/10/cal

* HOST CONFIG KEY ?

Passwörter und Privileges

TOWER

demo-user

VIEW

Dashboard

Jobs

Schedules

My View

RESOURCES

Templates

Credentials

Projects

Inventories

Inventory Scripts

ACCESS

Organizations

MY VIEW

JOB TEMPLATES 1

SEARCH

Q

KEY

Survey: Hetzner: Add virtual machine

Job Template

ACTIVITY

INVENTORY

PROJECT

LAST MODIFIED

LAST RAN

Inventory: Satellite 6

GIT: RK IT (@master)

5/23/2019 6:45:08 PM by admin

5/23/2019 6:45:08 PM

JOBS 0

SEARCH

Q

KEY

MY JOBS

ALL JOBS

PROMPT

SURVEY

PREVIEW

* NAME

* SERVER TYPE

cx11-ceph

* IMAGE

centos-7

* LOCATION

nbg1

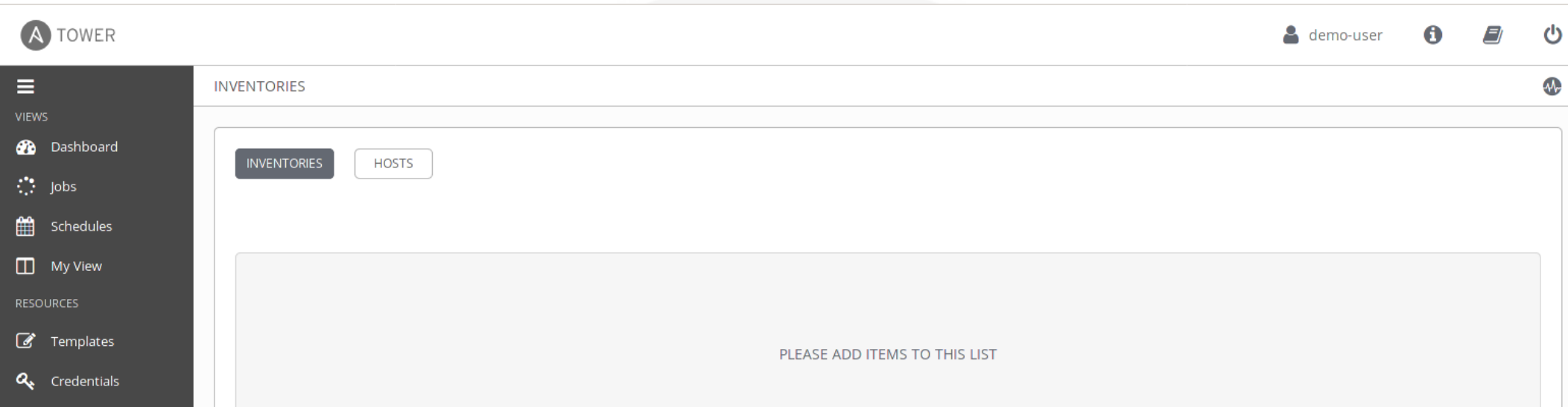
CANCEL

NEXT

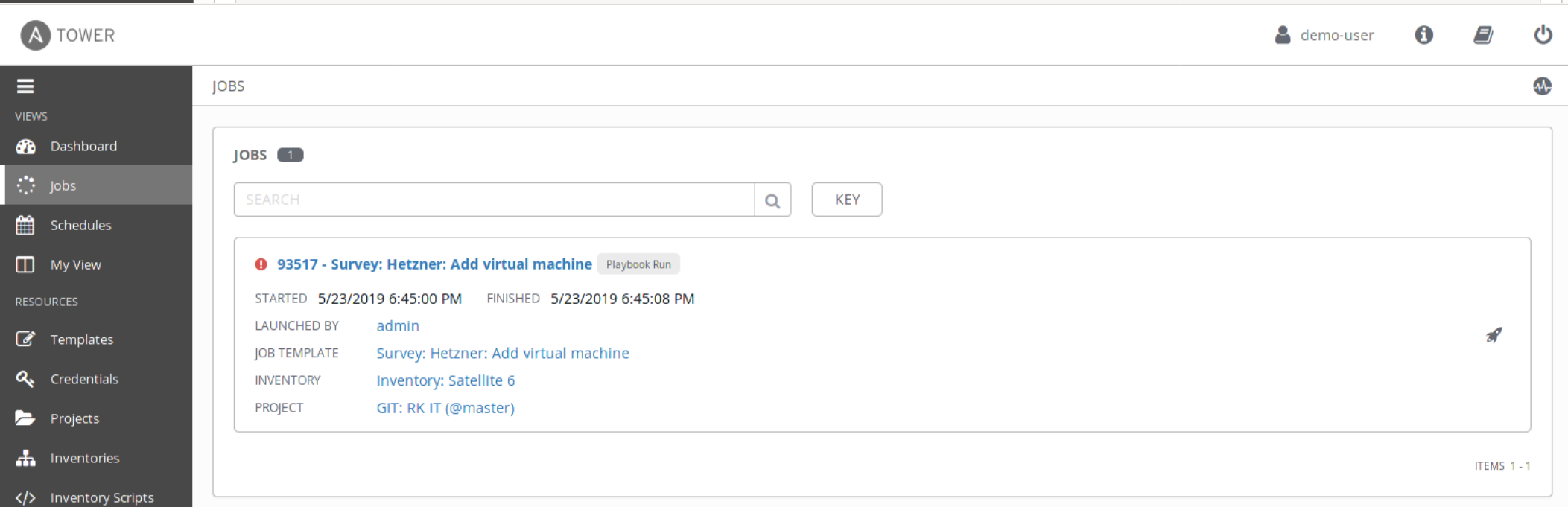
56/63

Secure Lin

Passwörter und Privileges



The screenshot shows the Tower web interface. The top navigation bar includes the Tower logo, a user profile for 'demo-user', and icons for help, settings, and power. The left sidebar contains a menu with 'VIEWS' (Dashboard, Jobs, Schedules, My View) and 'RESOURCES' (Templates, Credentials). The main content area is titled 'INVENTORIES' and features two tabs: 'INVENTORIES' (active) and 'HOSTS'. Below the tabs is a large grey box with the text 'PLEASE ADD ITEMS TO THIS LIST'.



The screenshot shows the Tower web interface for the 'JOBS' page. The top navigation bar is identical to the previous screenshot. The left sidebar is also identical, but the 'Jobs' item is highlighted. The main content area is titled 'JOBS' and features a search bar and a 'KEY' button. Below these is a job entry for '93517 - Survey: Hetzner: Add virtual machine' with a 'Playbook Run' status. The job details are as follows:

STARTED	5/23/2019 6:45:00 PM	FINISHED	5/23/2019 6:45:08 PM
LAUNCHED BY	admin		
JOB TEMPLATE	Survey: Hetzner: Add virtual machine		
INVENTORY	Inventory: Satellite 6		
PROJECT	GIT: RK IT (@master)		

At the bottom right of the job entry, it says 'ITEMS 1 - 1'.

Workflow-Editor

- Definieren eines Workflows durch Kombinieren von Playbooks
- Beispiel:
 - Abteilung Infrastruktur erstellt Playbook für GCE-Deployment
 - Abteilung Linux/Unix erstellt Playbook für RHEL-7 Bootstrapping
 - Abteilung Datenbanken erstellt Playbook für PostgreSQL-Installation
 - Workflow kombiniert diese Playbooks für das Deployment einer PostgreSQL-DB in der Google-Cloud

Workflow Editor

WORKFLOW VISUALIZER | Workflow: Linux: Build Katello Client packages for armhfp



TOTAL NODES 2 

START

Playbook: Linux: Install an...

Playbook: Linux: Build pack...

Playbook: Linux: Install and Configure Mock

JOB | PROJECT SYNC | INVENTORY SYNC

SEARCH  KEY

NAME ^	
☐ Check: Linux: Bootstrap (custom filter)	
☐ Playbook: Linux: Add ansible user (root password)	
☐ Playbook: Linux: Add ansible user (root ssh key)	
☐ Playbook: Linux: Bootstrap (custom filter)	
☐ Playbook: Linux: Bootstrap DEV Systems	
☐ Playbook: Linux: Bootstrap PROD Systems	
☐ Playbook: Linux: Bootstrap QA Systems	
☐ Playbook: Linux: Build packages for armhfp	
☐ Playbook: Linux: Configure Workstations	
☒ Playbook: Linux: Install and Configure Mock	

< 1 2 3 > PAGE 1 OF 3 ITEMS 1 - 10 OF 26

CANCEL SELECT

Workflow Editor

WORKFLOW VISUALIZER | Workflow: Linux: Build Katello Client packages for armhfp



TOTAL NODES 2 

START

Playbook: Linux: Install an...

Playbook: Linux: Build pack...

Playbook: Linux: Build packages for armhfp

JOBS

PROJECT SYNC

INVENTORY SYNC

SEARCH



KEY

NAME ^

- ☐ Check: Linux: Bootstrap (custom filter) ?
- ☐ Playbook: Linux: Add ansible user (root password) ?
- ☐ Playbook: Linux: Add ansible user (root ssh key) ?
- ☐ Playbook: Linux: Bootstrap (custom filter) ?
- ☐ Playbook: Linux: Bootstrap DEV Systems ?
- ☐ Playbook: Linux: Bootstrap PROD Systems ?
- ☐ Playbook: Linux: Bootstrap QA Systems ?
- ☒ Playbook: Linux: Build packages for armhfp ?
- ☐ Playbook: Linux: Configure Workstations ?
- ☐ Playbook: Linux: Install and Configure Mock ?

< 1 2 3 >

PAGE 1 OF 3

ITEMS 1 - 10 OF 26

CANCEL

SELECT

Fragen und Antworten

DI (FH) René Koch
rkoch@rk-it.at
Freelancer

Secure Linux Administration Conference, 29.05.2019

Zusammenfassung

- Ansible ist ein mächtiges Automatisierungs-Tools für diverse Systeme (nicht nur Linux)
- Red Hat Satellite ist System-Management-Tool für (Red Hat) Linux-Systeme
- Ansible Tower erweitert Ansible um RBAC, Scheduling, Web-UI und REST-API
- Ansible Tower und Red Hat Satellite kombinieren Provisioning, Content Management, Subscription Management, Compliance, Konfigurations-Management und Deployment mit RBAC, Scheduling und Web-Interfaces
- Nächste Ausbaustufe: Red Hat CloudForms (kombiniert mit Red Hat Satellite und Ansible Tower)



Danke für Ihre Aufmerksamkeit!

DI (FH) René Koch
rkoch@rk-it.at
Freelancer

Secure Linux Administration Conference, 29.05.2019