



# Сценарии использования Red Hat Ansible Automation

Игорь Крошкин  
Системный архитектор  
14 сентября 2018, Москва

# Зачем нужен Ansible?

## Традиционный подход к автоматизации [1/2]

Ручное администрирование:

- Занимает много времени
- Риск возникновения ошибок
- Нет учета изменений
- Невозможно повторить

```
[root@mgmt ~]# ssh server1.example.com
root@server1.example.com's password:
Last login: Sun Jun  5 15:27:37 2016 from mgmt.example.com
[root@server1 ~]# yum install httpd
[root@server1 ~]# vi /etc/resolv.conf
.
.
.
[root@mgmt ~]# ssh server2.example.com
root@server2.example.com's password:
Last login: Sun Jun  5 15:28:37 2016 from mgmt.example.com
[root@server2 ~]# yum install httpd
[root@server2 ~]#
.
.
.
[root@mgmt ~]# ssh server3.example.com
root@server3.example.com's password:
Last login: Sun Jun  5 15:29:37 2016 from mgmt.example.com
[root@server3 ~]# yum install httpd
[root@server3 ~]# vi /etc/resolv.conf
.
.
.
etc..
```

Разные настройки приводят к несоответствию между системами

# Зачем нужен Ansible?

## Традиционный подход к автоматизации [2/2]

Использование сценариев:

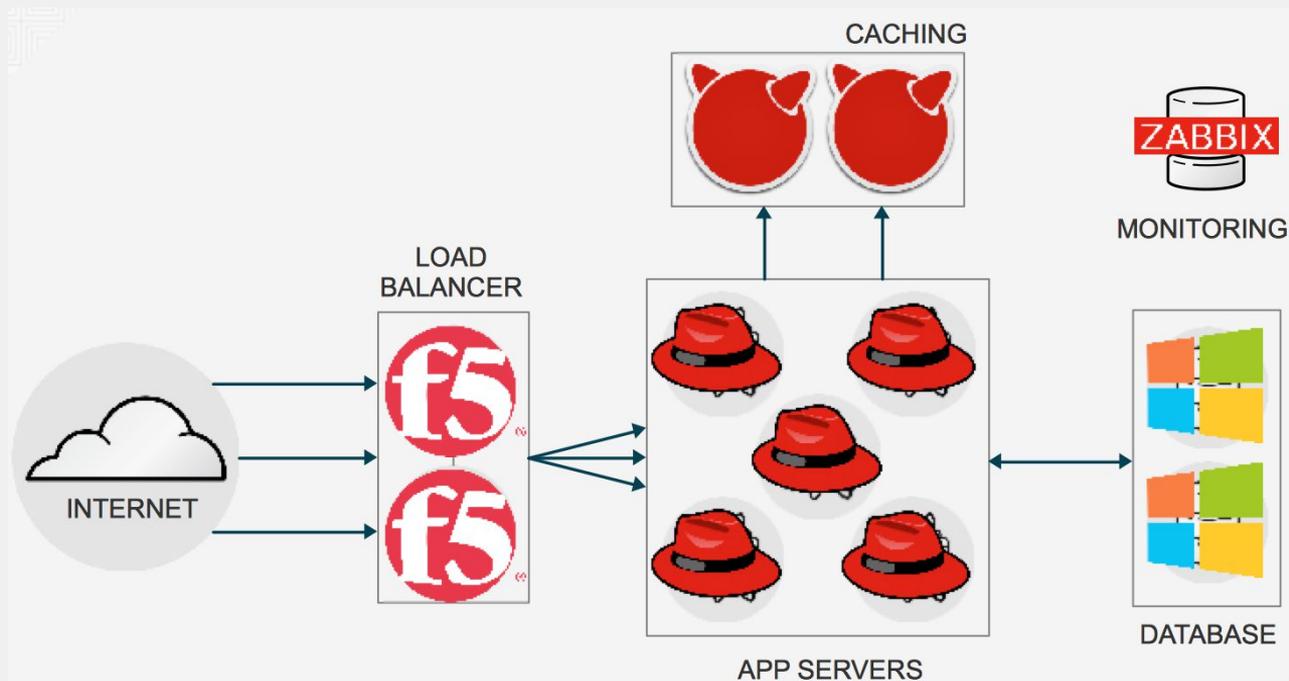
- Недолговечны
- Трудоемкость создания и внесения изменений
- Тяжело сопровождать
- Нет учета изменений

```
#!/bin/sh
HOSTS="
server1.example.com
server2.example.com
server3.example.com
db1.example.com
db2.example.com
"
for host in $HOSTS
do
# Copy DNS settings to all servers
ssh $host "sudo echo \"nameserver 8.8.8.8 >> /etc/resolv.conf\""
# Install Apache
ssh $host "sudo yum install httpd"
done
```

Каждый использует свой язык и набор сценариев

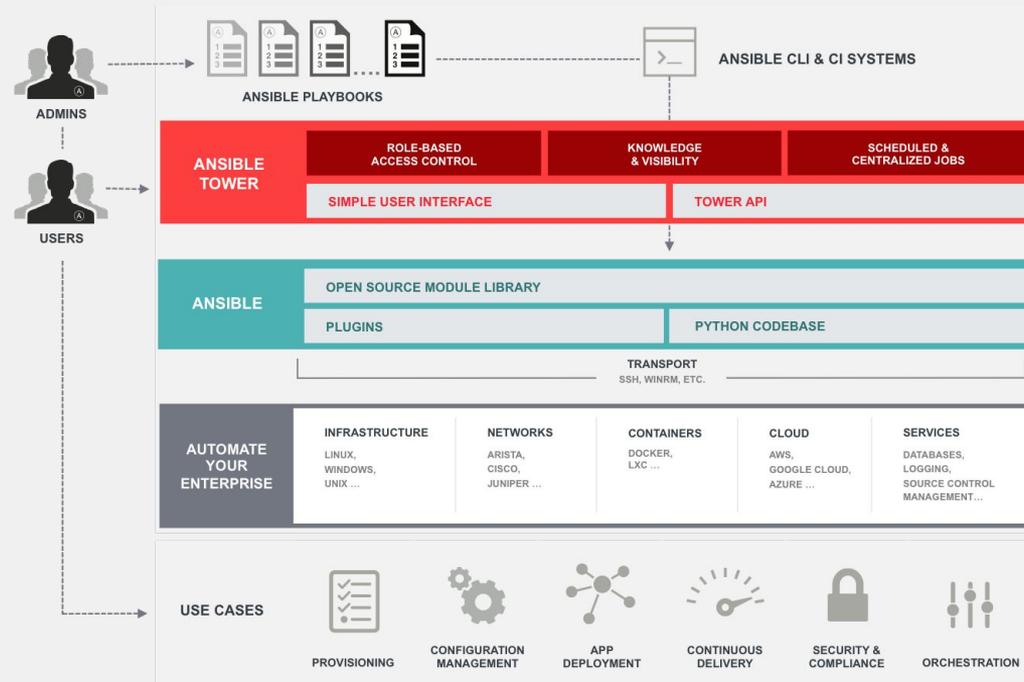
# Зачем нужен Ansible?

Жизненный цикл (развертывание, внесение изменений, обновление, обслуживание, списание) распределенного сервиса



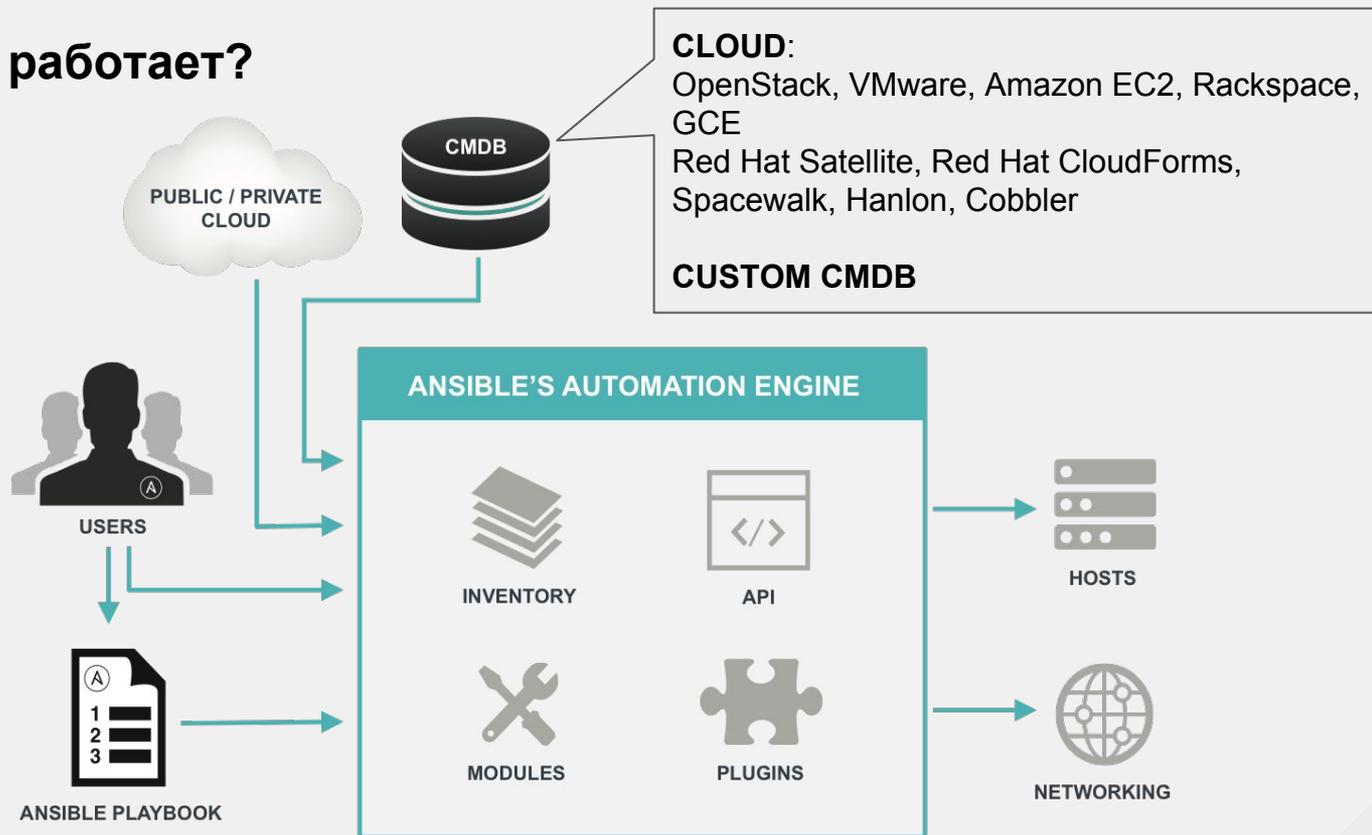
# Red Hat Ansible Automation

## Архитектура



# Red Hat Ansible Automation

Как это работает?



# Red Hat Ansible Automation

## Преимущества



### ПРОСТОТА В ИСПОЛЬЗОВАНИИ

Простой язык на основе YAML

Не требует навыков программирования

Последовательное выполнение задач

Автоматизация для всех



### ФУНКЦИОНАЛЬНЫЕ ВОЗМОЖНОСТИ

Развертывание приложений

Управление изменениями

Автоматизация рабочих процессов

Оркестрация между различными платформами



### БЕЗОПАСНОСТЬ

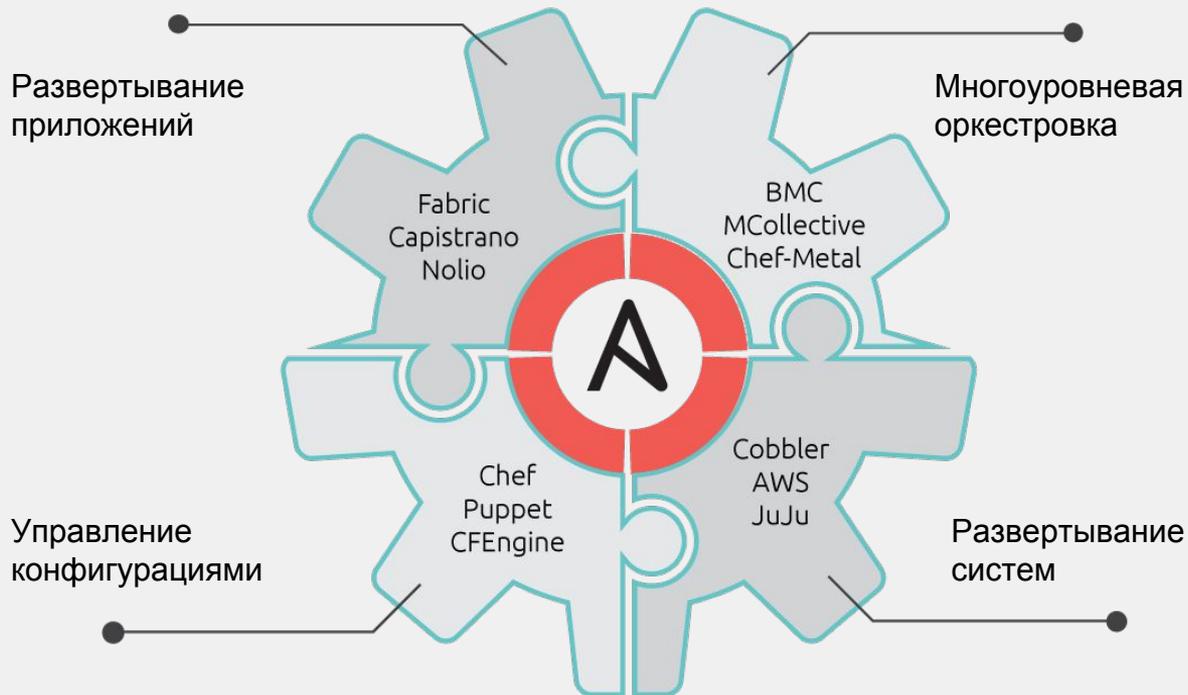
Не требует агента на клиенте

OpenSSH & WinRM в качестве транспорта

Эскалация привилегий

# Red Hat Ansible Automation

## Сценарии использования



# Red Hat Ansible Automation

## 1250+ модулей

[Docs](#) » [Module Index](#)

### Module Index

- [All Modules](#)
- [Cloud Modules](#)
- [Clustering Modules](#)
- [Commands Modules](#)
- [Crypto Modules](#)
- [Database Modules](#)
- [Files Modules](#)
- [Identity Modules](#)
- [Inventory Modules](#)
- [Messaging Modules](#)
- [Monitoring Modules](#)
- [Net Tools Modules](#)
- [Network Modules](#)
- [Notification Modules](#)
- [Packaging Modules](#)
- [Remote Management Modules](#)
- [Source Control Modules](#)
- [Storage Modules](#)
- [System Modules](#)
- [Utilities Modules](#)
- [Web Infrastructure Modules](#)
- [Windows Modules](#)

### at - Schedule the execution of a command or script file via the at command.

New in version 1.5.

- [Synopsis](#)
- [Requirements \(on host that executes module\)](#)
- [Options](#)
- [Examples](#)
  - [Status](#)
  - [Support](#)

#### Synopsis

- Use this module to schedule a command or script file to run once in the future.
- All jobs are executed in the 'a' queue.

#### Requirements (on host that executes module)

- at

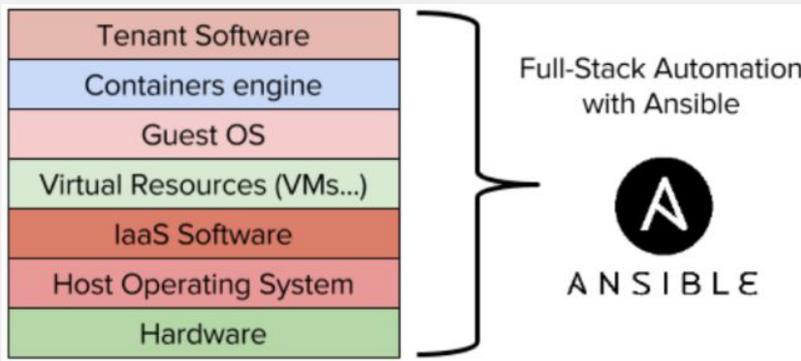
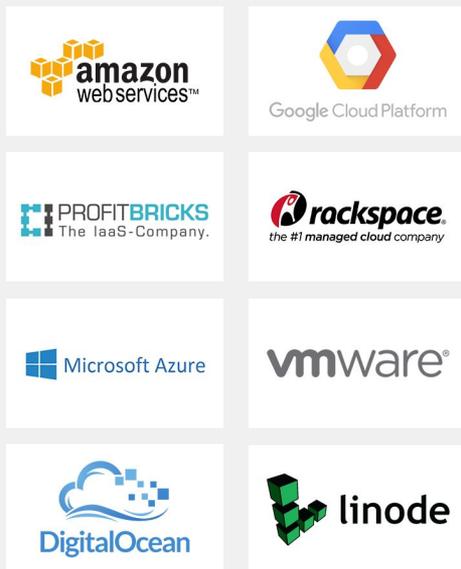
#### Options

parameter	required	default	choices	comments
command	no			A command to be executed in the future.
count	yes			The count of units in the future to execute the command or script file.

[http://docs.ansible.com/modules\\_by\\_category.htm](http://docs.ansible.com/modules_by_category.htm)

# Управление облачными провайдерами

300+ модулей



# Управление сетевыми устройствами

## 450+ модулей

- A10
- Apstra
- Arista EOS (cli, eAPI), CVP
- Aruba Networks
- AVI Networks
- Big Switch Networks
- Cisco ACI, AireOS, ASA, IOS, IOS-XR, NX-OS
- Citrix Netscaler
- Cumulus Linux
- Dell OS6, OS9, OS10
- Exoscale
- F5 BIG-IP
- Fortinet FortiOS
- Huawei
- Illumos
- Juniper Junos
- Lenovo
- Ordnance
- NETCONF
- Netvisor
- Openswitch
- Open vSwitch (OVS)
- Palo Alto PAN-OS
- Nokia SR OS
- VyOS

<https://access.redhat.com/solutions/3184741>

# Управление ОС Microsoft Windows

## 50+ модулей

- winrm используется для удаленного управления с помощью PowerShell
- PowerShell 3.0 используется для большинства модулей
- win\_chocolatey
  - установка, настройка, обновление пакетов
- win\_service
  - управление Windows-службами
- win\_firewall
- win\_user, win\_domain\_user
- win\_domain\_controller
- script
  - Выполнение произвольного сценария PowerShell

[http://docs.ansible.com/ansible/latest/list\\_of\\_windows\\_modules.html](http://docs.ansible.com/ansible/latest/list_of_windows_modules.html)

# Red Hat Ansible Tower

## Возможности



### АВТОМАТИЗАЦИЯ ДЛЯ КАЖДОГО

#### УПРАВЛЕНИЕ

Графический интерфейс,  
запуск задач по расписанию

#### КОНТРОЛЬ

Прозрачность, статус и  
результат выполнения задач

#### РОЛЕВАЯ МОДЕЛЬ

Делегирование полномочий  
и самообслуживание

#### ПРОСТОТА

Все используют один общий  
язык

#### МОЩНЫЕ ВОЗМОЖНОСТИ

REST API, поддержка  
сложных сценариев

#### ОТСУТСТВИЕ АГЕНТА

Предсказуемость,  
надежность, безопасность

ИСПОЛЬЗУЕТ ДВИЖОК ANSIBLE

# Сценарии использования Red Hat Ansible Tower

## Управление шаблонами задач

- Шаблоны задач включают в себя проекты, списки узлов, учетные данные для запуска задач
- Роли: Admin, Execute, Read

The screenshot shows the 'NEW JOB TEMPLATE' form in the Red Hat Ansible Tower interface. The form is titled 'NEW JOB TEMPLATE' and has a breadcrumb trail 'TEMPLATES / CREATE JOB TEMPLATE'. It features several tabs: 'DETAILS' (selected), 'PERMISSIONS', 'NOTIFICATIONS', 'COMPLETED JOBS', and 'ADD SURVEY'. The form is organized into columns and rows of input fields and controls. Key fields include: 'NAME' (text input), 'DESCRIPTION' (text input), 'JOB TYPE' (dropdown menu with 'Run' selected), 'INVENTORY' (text input with search icon and 'PROMPT ON LAUNCH' checkbox), 'PROJECT' (text input with search icon and 'Demo Project' selected), 'PLAYBOOK' (dropdown menu with 'Choose a playbook' selected), 'CREDENTIAL' (text input with search icon and 'PROMPT ON LAUNCH' checkbox), 'FORKS' (dropdown menu with 'DEFAULT' selected), 'LIMIT' (text input and 'PROMPT ON LAUNCH' checkbox), 'VERBOSITY' (dropdown menu with '0 (Normal)' selected), 'INSTANCE GROUPS' (text input with search icon), 'JOB TAGS' (text input and 'PROMPT ON LAUNCH' checkbox), 'SKIP TAGS' (text input and 'PROMPT ON LAUNCH' checkbox), 'LABELS' (text input), and 'SHOW CHANGES' (checkbox with 'OFF' button). At the bottom, there is an 'OPTIONS' section with four checkboxes: 'Enable Privilege Escalation', 'Allow Provisioning Callbacks', 'Enable Concurrent Jobs', and 'Use Fact Cache'.



# Сценарии использования Red Hat Ansible Tower

## Контроль

- Регистрация всех действия в БД
- Статус выполнения сценариев
- История выполненных задач

The screenshot displays the Red Hat Ansible Tower web interface. At the top, there are navigation tabs: TOWER, PROJECTS, INVENTORIES, TEMPLATES, and JOBS. Below this, the breadcrumb path is 'JOBS / 248 - APPLY STANDARD CONFIGURATION'. The main content area is divided into two panels. The left panel, titled 'DETAILS', shows the following information: STATUS: Successful (indicated by a green dot); STARTED: 2/23/2017 4:45:29 PM; FINISHED: 2/23/2017 4:47:25 PM; TEMPLATE: Apply standard configuration; JOB TYPE: Run; INVENTORY: Research Servers; PROJECT: Atmospheric Processor (indicated by a green dot); REVISION: 108a25f7964503c6bf46593e93528846e68d836b; PLAYBOOK: apply-configuration.yml; MACHINE CREDENTIAL: Weyland-Yutani ssh key; FORKS: 0. The right panel, titled 'APPLY STANDARD CONFIGURATION', shows a search bar and a list of tasks. The first task is expanded, showing its output: 'TASK [Ensure users are present]' followed by several lines of 'changed' messages for different EC2 instances, such as 'changed: [ec2-54-160-241-172.compute-1.amazonaws.com: u'apone', u'uid': 1200}]'.

# Сценарии использования Red Hat Ansible Tower

## Ролевая модель доступа

- Управление на уровне организаций, рабочих групп
- Предоставление гранулярного доступа к объектам

The screenshot displays the 'DEVELOPERS | ADD PERMISSIONS' interface. It features a search bar and a list of resources. The 'DEVELOPERS' role is selected, and the 'PERMISSIONS' tab is active. A table shows the permissions assigned to the role:

NAME	TYPE	ROLE	ACTIONS
Demo Project	Project	Update	✕
Demo Inventory	Inventory	Use	✕
Demo Job Template	Job Template	Execute	✕

At the bottom right of the table, it indicates 'ITEMS 1 - 3'. A '+ ADD PERMISSIONS' button is visible in the top right corner of the modal.

# Сценарии использования Red Hat Ansible Tower

## Управление списком узлов

- Динамическая инвентаризация клиентов
  - Rackspace, GCE, Amazon EC2, Azure
  - VMware vCenter
  - Red Hat Satellite
  - Red Hat CloudForms
- Запуск сценариев по расписанию
- Уведомления в режиме реального времени

The screenshot displays the Red Hat Ansible Tower web interface. The top navigation bar includes 'TOWER', 'PROJECTS', 'INVENTORIES', 'TEMPLATES', 'JOBS', and a user profile 'admin'. The breadcrumb trail indicates the current page is 'INVENTORIES / MANAGE CLOUD STAGING SERVERS / EDIT'.

The main content area is titled 'CLOUD SERVERS' and contains several configuration sections:

- DETAILS** (selected) and **NOTIFICATIONS** tabs.
- CLOUD SERVERS** section:
  - NAME:** Cloud servers
  - DESCRIPTION:** (empty)
  - SOURCE:** Amazon EC2
  - CLOUD CREDENTIAL:** Amazon keys
  - REGIONS:** US East (Northern Virginia)
  - INSTANCE FILTERS:** tag:Name=\*staging\*
  - ONLY GROUP BY:** (empty)
  - UPDATE OPTIONS:**  Overwrite,  Overwrite Variables,  Update on Launch
- VARIABLES** section:  YAML,  JSON
- DAILY REMEDIATION** section:
  - NAME:** Daily remediation
  - START DATE (MM/DD/YYYY):** 10/03/2016
  - START TIME (HH:MM:SS):** 01:23:45
  - LOCAL TIME ZONE:** America/New\_York
  - REPEAT FREQUENCY:** Day
- FREQUENCY DETAILS** section:
  - EVERY:** 1 DAYS
  - END:** Never
- SCHEDULE DESCRIPTION** section:
  - SCHEDULE DESCRIPTION:** every day
  - OCCURRENCES (limited to first 10):** 10/03/2016 01:23:45 EDT, 10/04/2016 01:23:45 EDT, 10/05/2016 01:23:45 EDT
  - DATE FORMAT:** LOCAL TIME (selected), UTC

# Сценарии использования Red Hat Ansible Tower

## Smart Inventory

INVENTORIES HOSTS

SEARCH

KEY SMART INVENTORY

Create a new Smart Inventory from results.

NAME	INVENTORY
<span>ON</span> 52.6.114.136	AWS
<span>ON</span> 54.164.137.84	ben_inventory_test, AWS

# Сценарии использования Red Hat Ansible Tower

## Хранение учетных данных

- Учетные данные хранятся в зашифрованном виде
- Пользователю нет необходимости иметь учетную запись на целевой системе
- Типы учетных записей:
  - Machine, Network, Source Control
  - Vault
  - Amazon, GCE, Azure, OpenStack
  - Red Hat Satellite, Red Hat CloudForms, Red Hat Insights
  - Red Hat Virtualization, VMware vCenter

The screenshot displays the configuration page for a credential in Red Hat Ansible Tower. The form is titled 'Mail Servers Credential' and is set to 'Machine' type. It includes fields for 'NAME', 'DESCRIPTION', and 'ORGANIZATION' (set to 'Default'). Under 'TYPE DETAILS', there are sections for 'USERNAME' (set to 'playbook'), 'PASSWORD' (with a 'SHOW' button and an 'Ask at runtime?' checkbox), and 'PRIVATE KEY PASSPHRASE' (with a 'SHOW' button and an 'Ask at runtime?' checkbox). The 'PRIVILEGE ESCALATION' is set to 'Sudo', with fields for 'PRIVILEGE ESCALATION USERNAME' and 'PRIVILEGE ESCALATION PASSWORD' (with a 'SHOW' button and an 'Ask at runtime?' checkbox). A 'VAULT PASSWORD' section also has a 'SHOW' button and an 'Ask at runtime?' checkbox. The 'PRIVATE KEY' field is marked as '\$encrypted\$'.

# Сценарии использования Red Hat Ansible Tower

## Provisioning callbacks

- Запуск сценария через REST API, CLI, tower-cli
- Используются для начальной конфигурации либо периодического запуска с помощью cron

OPTIONS	PROVISIONING CALLBACK URL 	HOST CONFIG KEY 
<input type="checkbox"/> Enable Privilege Escalation 	<input type="text" value="https://10.42.0.42:443/api/v1/job_templates/5/callb"/>	<input type="text"/>
<input checked="" type="checkbox"/> Allow Provisioning Callbacks 		

- ```
./request_tower_configuration.sh -h
Usage: ./request_tower_configuration.sh <options>
Request server configuration from Ansible Tower.
```
- ```
root@localhost:~$ curl -f -H 'Content-Type: application/json' -XPOST \
    -d '{"host_config_key": "5a8ec154832b780b9bdef1061764ae5a",
    "extra_vars": "{\\"foo\\": \\"bar\\"}"}' \
    http://<Tower server name>/api/v2/job_templates/1/callback
```

# Сценарии использования Red Hat Ansible Tower

## Управление рабочими процессами

- Встроенный редактор рабочих процессов
- Возможность группировать сценарии в зависимости от условий и результатов выполнения предыдущего

The screenshot displays the Red Hat Ansible Tower interface. On the left, a workflow diagram shows a 'START' node leading to an 'Ansible Playbook' node. From this node, the flow branches into two paths: one leading to 'Deploy Database' and 'Deploy WebServers', and another leading to 'Deploy Load Balancer'. The 'Deploy Database' and 'Deploy WebServers' nodes are connected to 'Deploy Load Balancer'. Below the 'Ansible Playbook' node, there are two more nodes: 'Open-ssh' and 'Apache Setup Template', which are also connected to the main flow.

On the right, the 'ANSIBLE PLAYBOOKS' section is visible. It includes tabs for 'JOBS', 'PROJECT SYNC', and 'INVENTORY SYNC'. Below these is a search bar with a 'KEY' button. A list of playbooks is shown, each with a radio button and an 'INFO' button:

- Apache Setup Template
- Demo Job Template
- Deploy Database
- Deploy Load Balancer
- Deploy WebServers

At the bottom of the list, there are navigation controls: '< 1 2 >' (with '2' highlighted), 'PAGE 1 OF 2', and 'ITEMS 1 - 5 OF 7'. There are also 'CANCEL' and 'SELECT' buttons.

# Сценарии использования Red Hat Ansible Tower

## Использование опросника для определения переменных

TEMPLATES / Demo Job Template

### Demo Job Template

DETAILS

COMPLETED JOBS

PERMISSIONS

NOTIFICATIONS

ADD SURVEY

TEMPLATES / DEMONSTRATION OF AWS STACK DEPLOY

### DEMONSTRATION OF AWS STACK DEPLOY

DETAILS

PERMISSIONS

NOTIFICATIONS

ADD SURVEY

WORKFLOW EDITOR

\* NAME

Demonstration of AWS stack deploy

DESCRIPTION

Demo Job Template | SURVEY

#### ADD SURVEY PROMPT

\* PROMPT

How many instances you need to deploy?

DESCRIPTION

JBOSS EAP instances to deploy

\* ANSWER VARIABLE NAME

eap\_count

\* ANSWER TYPE

Integer

MINIMUM

1

MAXIMUM

100

DEFAULT ANSWER

10

REQUIRED

CANCEL

+ ADD

Demo Job Template | SURVEY

#### ADD SURVEY PROMPT

\* PROMPT

Please enter user password:

DESCRIPTION

JBOSS EAP user password

\* ANSWER VARIABLE NAME

eap\_user\_password

\* ANSWER TYPE

Password

MINIMUM LENGTH

8

MAXIMUM LENGTH

32

DEFAULT ANSWER

SHOW

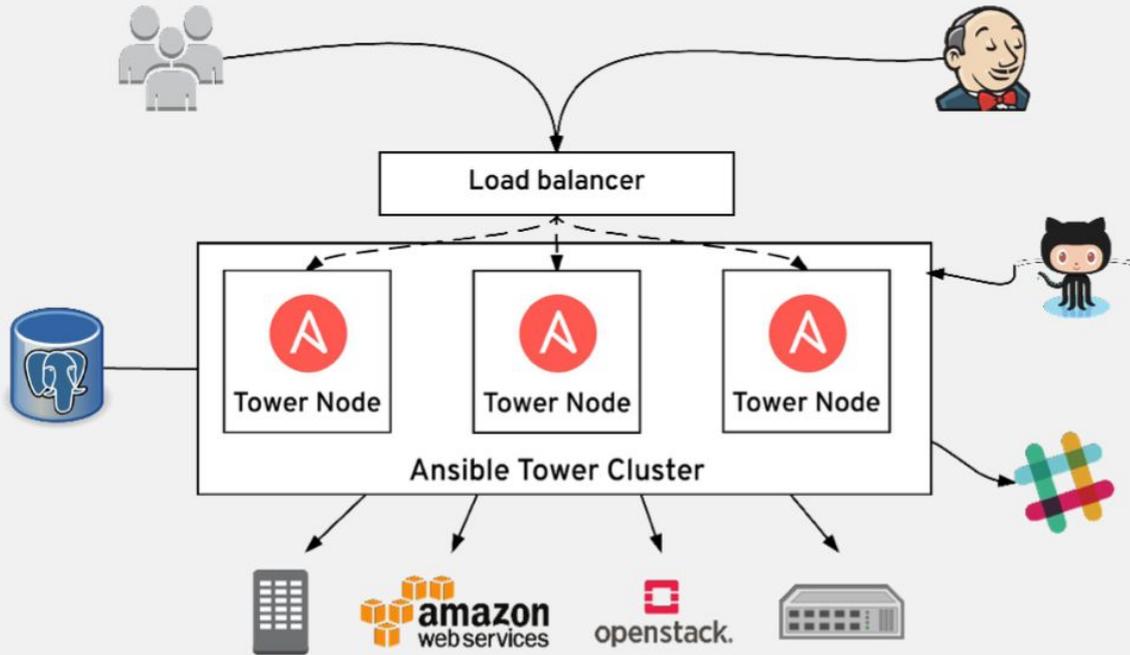
REQUIRED

CANCEL

+ ADD

# Сценарии использования Red Hat Ansible Tower

## Кластеризация и высокая доступность



# Интеграция с Red Hat CloudForms

## Совместные возможности

- Вызов сценариев Ansible:
  - В методах автоматизации
  - В действиях, определенных в созданных кнопках
  - В сервисном каталоге (Playbook as a Service)
  - В политиках безопасности как реакция на событие

The screenshot displays the Red Hat CloudForms Management Engine dashboard. The 'Current Status' section shows 35 Total Services, 0 Retiring Soon, 35 Current Services, and 0 Retired Services. The 'Featured Services' section includes: 2 Servers with RHEL 7.2 (Private Cloud Services), Amazon - Build a VM (User Public Cloud Services), Azure Wordpress Multi (Public Cloud Services), and AWS WP Scale Group (Public Cloud Services). An inset window titled 'Adding a new Service Catalog Item' shows a dropdown menu for 'Catalog Item Type' with options: Amazon, Ansible Playbook, AnsibleTower, Azure, Generic, and Google. Below the dashboard, a diagram labeled 'SERVICE' illustrates a stack: VM + ANSIBLE + VM + ANSIBLE.

# Интеграция с Red Hat Satellite

## Совместные возможности

- Использование Red Hat Satellite для динамической инвентаризации клиентов
  - [https://github.com/theforeman/foreman\\_ansible\\_inventory/](https://github.com/theforeman/foreman_ansible_inventory/)
- Возможность группировать узлы на основе принадлежности к hostgroup, location
- Использование данных facter в качестве переменных
- Автоматический запуск сценариев при развертывании нового узла, используя callback URL
- В будущем планируется поддержка сценариев Ansible для управления конфигурациями наряду с Puppet

### CREATE GROUP

DETAILS

NOTIFICATIONS

\* NAME

Demo

DESCRIPTION

SOURCE

Red Hat Satellite 6

# Интеграция с Red Hat Insights

## Совместные возможности

- Проактивное использование лучших практик Red Hat
- Возможность выбора узлов и событий
- Генерация сценариев из интерфейса Red Hat Insights

The screenshot displays the Red Hat Insights interface with a focus on system management and vulnerability assessment. A red box highlights the 'Actions' menu and a table of systems. The 'Actions' menu includes options: 'Create a new Plan/Playbook', 'Add to existing Plan/Playbook', and 'Unregister'. The table lists systems with columns for 'System Name', 'Last Check In', and 'Status'. Below this, another table shows vulnerability rules with columns for 'Action', 'Total Risk', 'Ansible', and 'Systems'. A third table at the bottom shows system details with columns for 'System', 'Last check in', and 'Status'. A red box highlights the 'GENERATE PLAYBOOK' button at the bottom right.

System Name	Last Check In	Status
ansible1tronik-insights440.atl.redhat.com	22 minutes ago	12 Actions
ansible2tronik-insights440.atl.redhat.com	22 minutes ago	12 Actions
ansible3tronik-insights440.obfuscated.host	22 minutes ago	12 Actions
atlantatronik-insights303.atl.redhat.com	a few seconds ago	11 Actions
insights-sat.tronik-insights440.atl.redhat.com	8 minutes ago	1 Action

Action	Total Risk	Ansible	Systems
OpenSSH vulnerable to remote password guessing attack (CVE-2015-5600)	High	Low	2
Remote code execution vulnerability in libresolv via crafted DNS response (CVE-2015-7547)	High	Low	3
Kernel vulnerable to man-in-the-middle via payload injection	High	Low	3

System	Last check in	Status
ansible1tronik-insights440.atl.redhat.com	an hour ago	—
ansible2tronik-insights440.atl.redhat.com	an hour ago	—
ansible3tronik-insights440.obfuscated.host	an hour ago	—



# THANK YOU



[plus.google.com/+RedHat](https://plus.google.com/+RedHat)



[facebook.com/redhatinc](https://facebook.com/redhatinc)



[linkedin.com/company/red-hat](https://linkedin.com/company/red-hat)



[twitter.com/RedHatNews](https://twitter.com/RedHatNews)



[youtube.com/user/RedHatVideos](https://youtube.com/user/RedHatVideos)