



Ansible. Автоматизация управления сетевой инфраструктурой

Александр Сафонов
Архитектор решений
asafonov@redhat.com

18 апреля 2018



28,000+
stars on GitHub

HERNISH
NATURE

1600+
Ansible modules

500,000+
downloads a month

ANSIBLE POPULARITY

● Puppet
Software

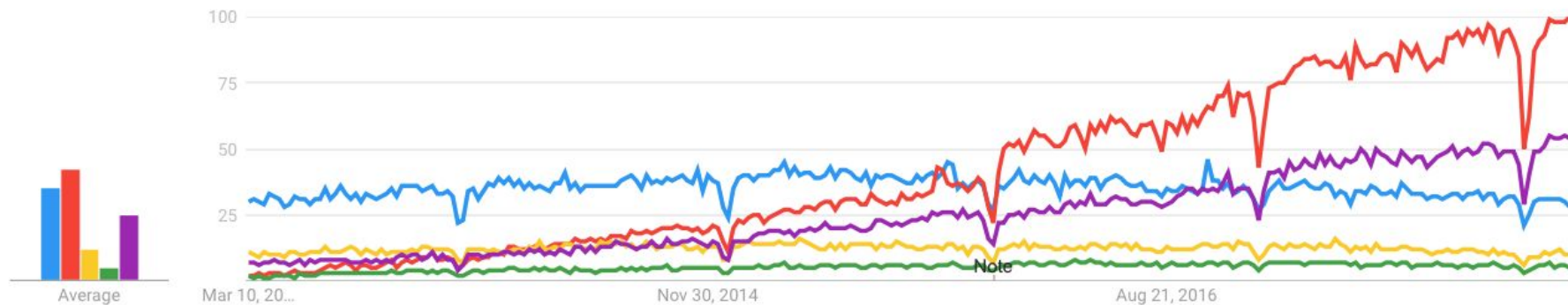
● Ansible
Software

● Chef
Topic

● Salt
Software

● DevOps
Search term

Interest over time 



ANSIBLE - ОСНОВНЫЕ 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

ANSIBLE - THE MOST POPULAR OPEN SOURCE AUTOMATION PLATFORM

Goal: Unify provisioning, configuration, and application deployment

Result: Ansible. A python-based command line engine that interprets and executes YAML-based “Playbooks” that contain one or more “plays” or tasks.

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

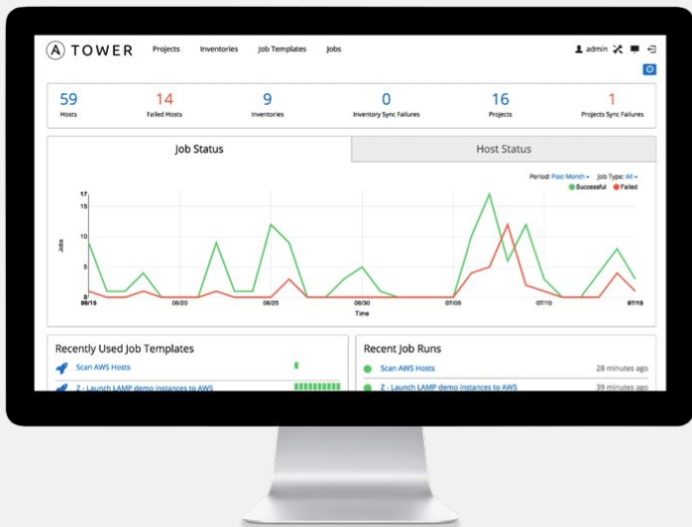
POWERFUL

AGENTLESS

FUELED BY AN INNOVATIVE **OPEN SOURCE** COMMUNITY

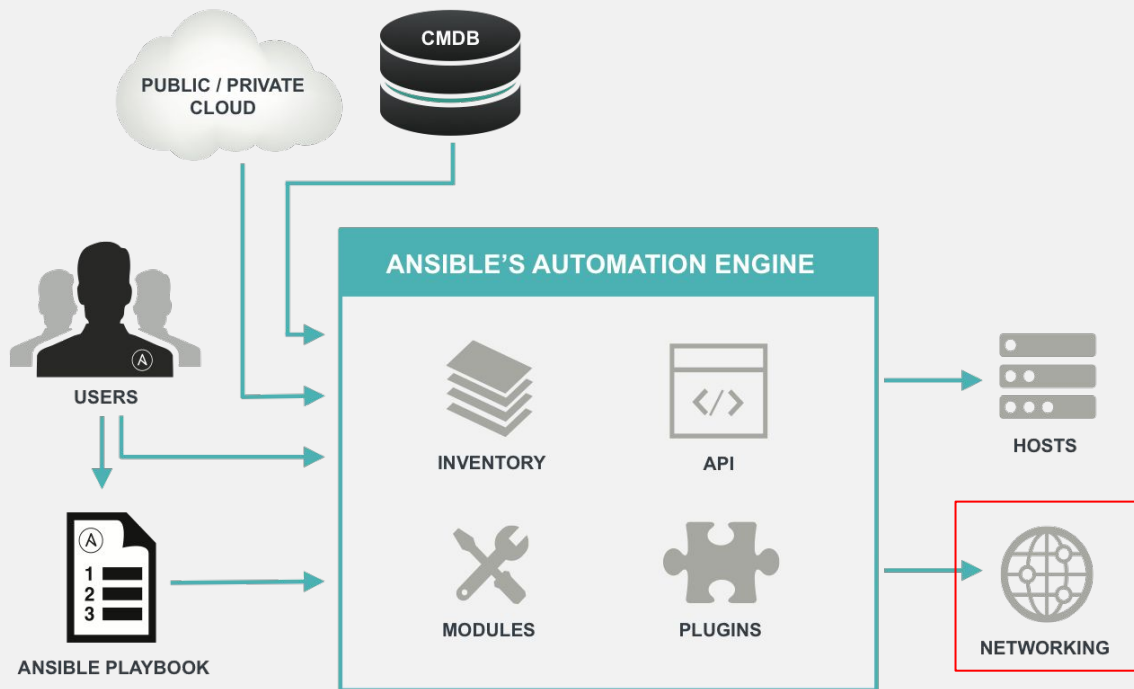
RED HAT ANSIBLE TOWER

Платформа для автоматизации, включающая в себя

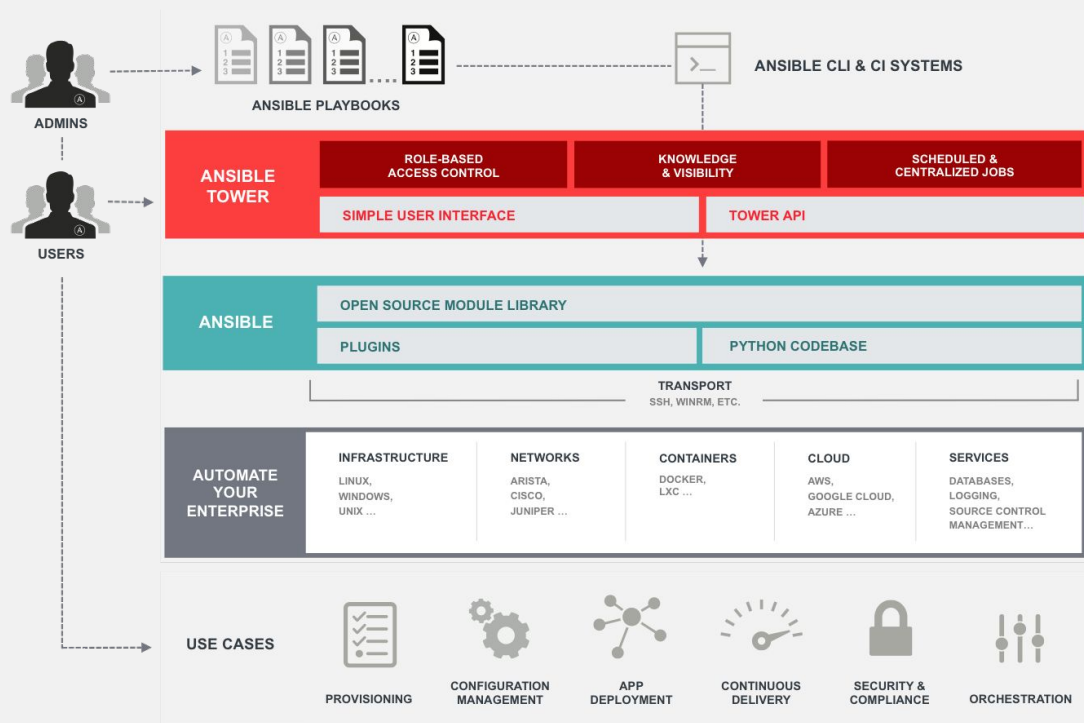


- **Язык** для автоматизации ИТ-инфраструктуры и приложений с помощью сценариев
- **Движок Ansible** для запуска и работы сценариев
- **Ansible Tower** предоставляет:
 - возможность автоматизировать рабочие процессы и определить взаимосвязь между сценариями
 - ролевая модель доступа для запуска сценариев
 - централизованное выполнение задач с регистрацией событий и их статусом
 - графический интерфейс и REST API

RED HAT ANSIBLE AUTOMATION ARCHITECTURE



RED HAT ANSIBLE TOWER ARCHITECTURE



<https://www.ansible.com/products/tower>

ANSIBLE - PROJECT & PRODUCT

Open Source
(Communities)



Red Hat Ansible Automation
(Enterprise)

OPS - IT Managers, "Teams"



Bottom-Up
Influence

Top-Down
Strategy

NETOPS - Network Operations

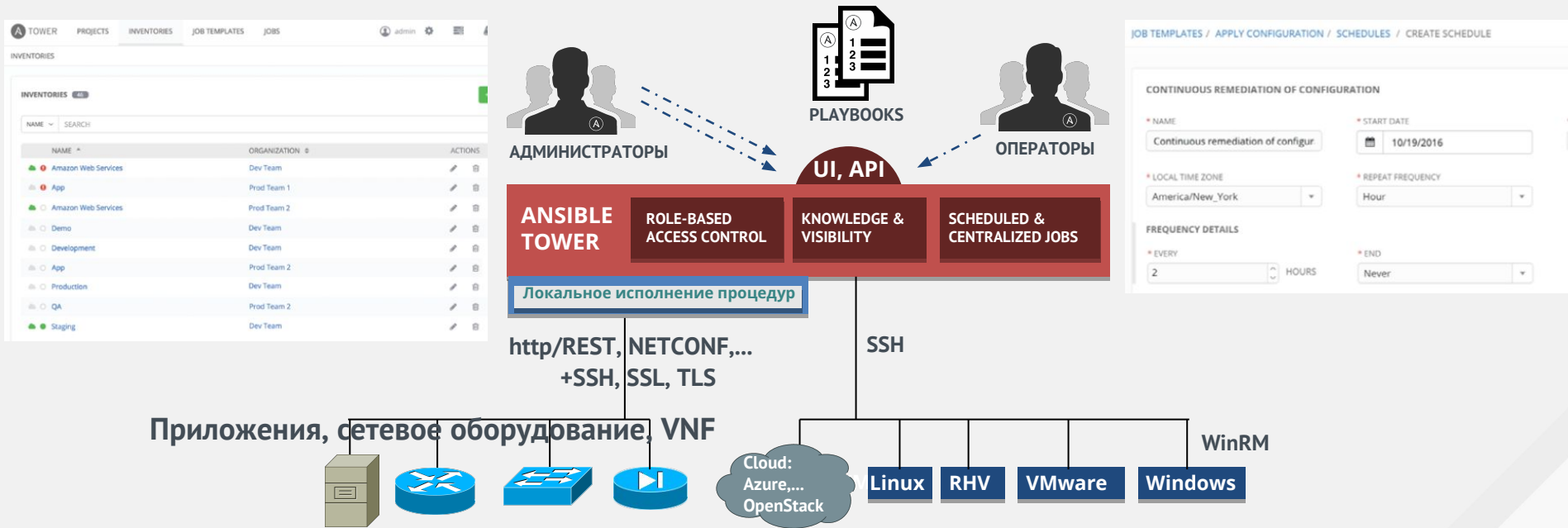


DEV - Playbook Authors, "Individuals"

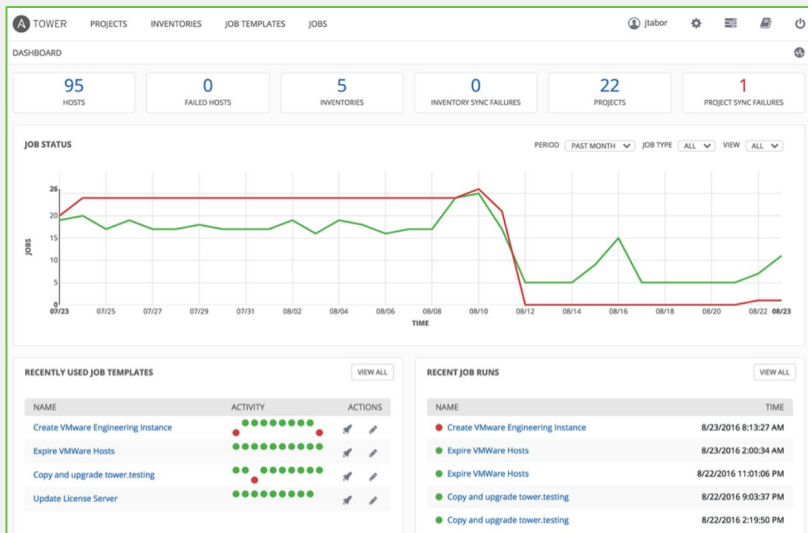
RED HAT ANSIBLE TOWER

RED HAT ANSIBLE TOWER

INFRASTRUCTURE-AS-A-CODE // УПРАВЛЕНИЕ КОНФИГУРАЦИЯМИ



Ansible Tower - Автоматизация



- Запуск сценариев нажатием одной кнопки
- Импорт сценариев из SCM
- CLI, REST API, remote execution, интеграции с CI/CD tools

The screenshot shows the Ansible Tower REST API interface. At the top, there is a search bar for REST API. Below this is the REST API endpoint: GET /api/format-api. The response is shown as a JSON object with the following structure:

```
HTTP 200 OK
Allow: GET, HEAD, OPTIONS
Content-Type: application/json
Vary: Accept
X-API-Time: 0.001s

{
  "available_versions": {
    "v1": "/api/v1/"
  },
  "description": "Ansible Tower REST API",
  "current_version": "/api/v1/"
}
```

The screenshot shows the Ansible Tower Run Command interface. At the top, there is a navigation bar with the TOWER logo and tabs for Projects, Inventories, Job Templates, and Jobs. Below this is the Run Command section. The Module is set to yum. The Arguments section is empty. The Host is set to adhoc.png. The Machine Credential is set to Banzai ssh key. The Verbose checkbox is checked. The Verbose level is set to Normal. There are buttons for Launch and Reset.

Ansible Tower - Контроль и RBAC

The screenshot shows the Ansible Tower interface for a job named 'APPLY STANDARD CONFIGURATION'. The left sidebar contains a 'DETAILS' panel with the following information:

- STATUS: Successful
- 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
- REVISION: 108a25f7964503c6bf46593e93528846e68d836b
- PLAYBOOK: apply-configuration.yml
- MACHINE CREDENTIAL: Weyland-Yutani ssh key
- FORKS: 0

The main area shows the job progress with a search bar and a list of tasks. The task 'TASK [Ensure users are present]' is expanded, showing a list of changes:

```
11 TASK [Ensure users are present]
12 changed: [ec2-54-160-241-172.compute-1.amazonaws.com]
13 changed: [ec2-54-91-190-27.compute-1.amazonaws.com]
14 changed: [ec2-54-227-29-23.compute-1.amazonaws.com]
15 changed: [ec2-54-160-241-172.compute-1.amazonaws.com]
16 changed: [ec2-54-227-29-23.compute-1.amazonaws.com]
```

The screenshot shows the 'CLOUDFORMS' details page. The 'DETAILS' tab is active, showing the following information:

- *NAME: CloudForms
- DESCRIPTION: Inventory Script
- ORGANIZATION: Bit63
- *TYPE: Red Hat CloudForms
- *CLOUDFORMS HOST: https://cloudforms07.bit63.net
- *USERNAME: admin
- *PASSWORD: SHOW [masked]

Below the details is the 'Help Desk | Add Permissions' dialog. It prompts the user to 'Please select resources from the lists below.' and provides a search bar and a table of resources:

NAME	DESCRIPTION
<input type="checkbox"/>	Demo Job Template

- Регистрация всех действия в БД
- Статус выполнения сценариев
- История выполненных задач
- Управление на уровне организаций, рабочих групп
- Хранение пользовательских данных в зашифрованном виде
- Не надо иметь учетки на клиентах

Ansible Tower - Управление

INVENTORIES / MANAGE CLOUD STAGING SERVERS / EDIT

CLOUD SERVERS

DETAILS NOTIFICATIONS

* 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: YAML JSON

JOB TEMPLATES SCHEDULES / JOB TEMPLATE SCHEDULES.EDIT

DAILY REMEDIATION

* NAME: Daily remediation

* START DATE (MM/DD/YYYY): 10/03/2016

* START TIME (HH24:MM:SS): 01:23:45

* LOCAL TIME ZONE: America/New_York

* REPEAT FREQUENCY: Day

Demo Job Template | SURVEY ON

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

- Динамическая инвентаризация клиентов
- Запуск сценариев по расписанию
- Использование опросника для задания доп. параметров
- Уведомления в режиме реального времени

Ansible Tower - Управление 'workflow'

EDIT WORKFLOW

KEY: — On Success — On Fail — Always P Project Sync I Inventory Sync TOTAL TEMPLATES 6

START → Demo Project (P) → (On Success) → Demo Job Template → Demo Job Template
(On Fail) → Demo Job Template
(Always) → Demo Job Template → Demo Job Template

DEMO JOB TEMPLATE

JOBS PROJECT SYNC INVENTORY SYNC

SEARCH 🔍 KEY

NAME ▲

Demo Job Template INFO

ITEMS 1 - 1 OF 1

* TYPE

On Success
 On Failure

CANCEL SELECT

Provision → Configure → Deploy → Scale

Build → Test → Promote → Verify → Deploy

Встроенный редактор рабочих процессов / workflow:

- Возможность группировать сценарии в зависимости от условий и результатов выполнения предыдущего
- Ролевая модель и результат выполнения задач

RED HAT ANSIBLE NETWORKING



**MANAGING NETWORKS
HASN'T CHANGED
IN 30 YEARS.**

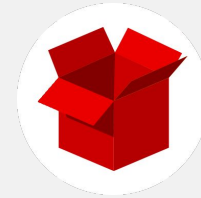
WHY HASN'T NETWORKING CHANGED?

Networking vendors are the trusted advisors



PEOPLE

- Domain specific skill sets
- Vendor oriented experience
- Siloed organizations
- Legacy operational practices



PRODUCTS

- Infrastructure-focused features
- Baroque, CLI-based interfaces
- Siloed technologies
- Monolithic, proprietary platforms

WHAT IS THE PRIMARY METHOD OF MAKING NETWORK CHANGES IN YOUR ENVIRONMENT?

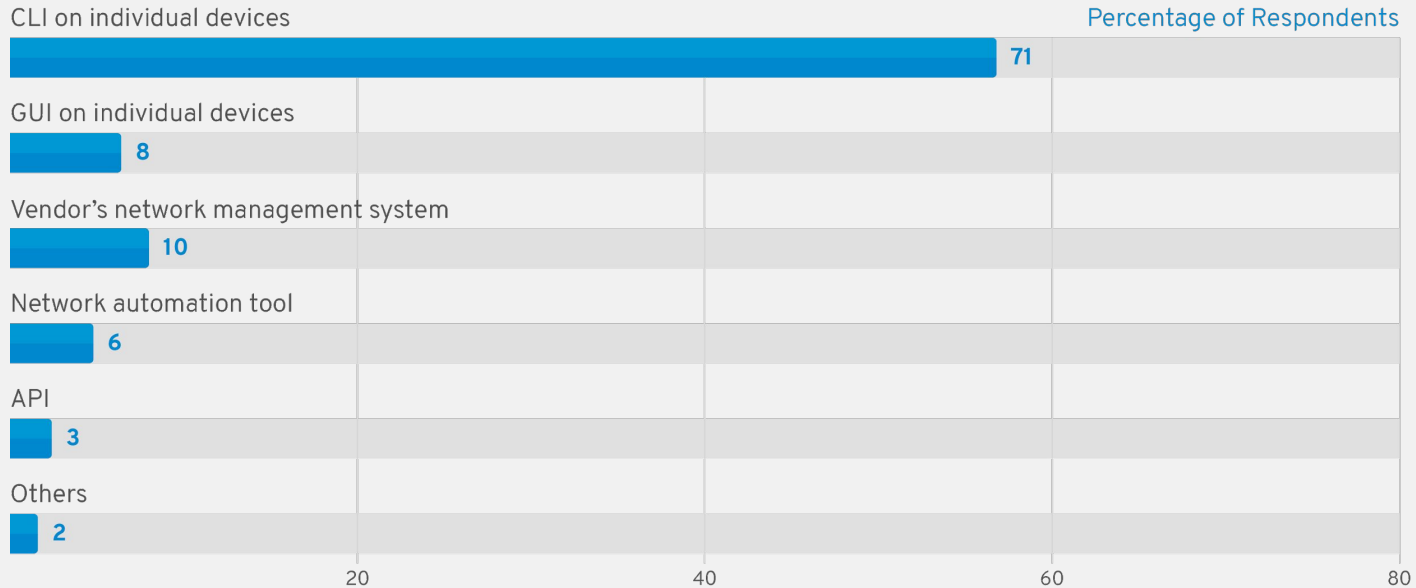


Figure 1
Primary Method for Making Network Changes

NETWORK AUTOMATION PROGRESS

Полный список поддерживаемых Red Hat Ansible Engine Networking Add-On платформ и модулей:
<https://access.redhat.com/solutions/3184741>

<https://www.ansible.com/overview/networking>

7 Platforms
28 Modules

2.1

May 2016

17 Platforms
141 Modules

2.2

Oct 2016

29 Platforms
267 Modules

Persistent
Connections

NETCONF
Support

2.3

Apr 2017

33 Platforms
463 Modules

Declarative
Intent

Aggregate
Resources

Platform
Agnostic

2.4

Sep 2017

40 Platforms
572 Modules

network_cli
connection
plug-in

NETCONF
connection
plug-in

Better
Logging

XML
Filters

More
Docs

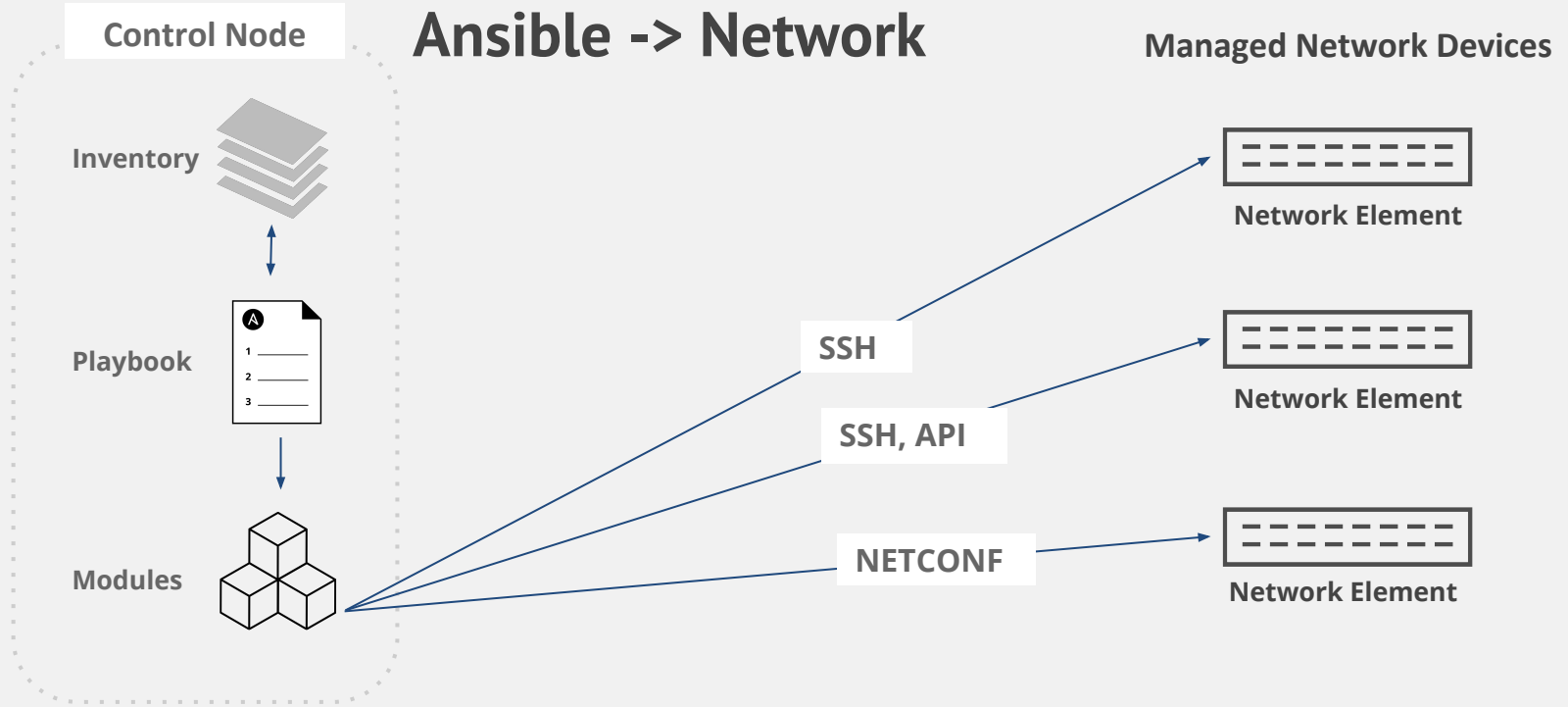
2.5

Mar 2018

NETWORK MODULES: BUILT-IN DEVICE ENABLEMENT

- A10
- Apstra AOS
- Arista EOS (cli, eAPI), CVP
- Aruba Networks
- AVI Networks
- Big Switch Networks
- Brocade Ironware
- Cisco ACI, AireOS, ASA, IOS, IOS-XR, NSO, NX-OS
- Citrix Netscaler
- Cumulus Linux
- Dell OS6, OS9, OS10
- Exoscale
- F5 BIG-IP
- Fortinet FortIOS, FMGR
- Huawei
- Illumos
- Infoblox NIOS
- Juniper Junos
- Lenovo CNOS, ENOS
- Mellanox ONYX
- Ordnance
- NETCONF
- Netvisor
- Openswitch
- Open vSwitch (OVS)
- Palo Alto PAN-OS
- Nokia NetAct, SR OS
- VyOS

Ansible -> Network



Modules:

Handles execution of remote system commands

Control Node:

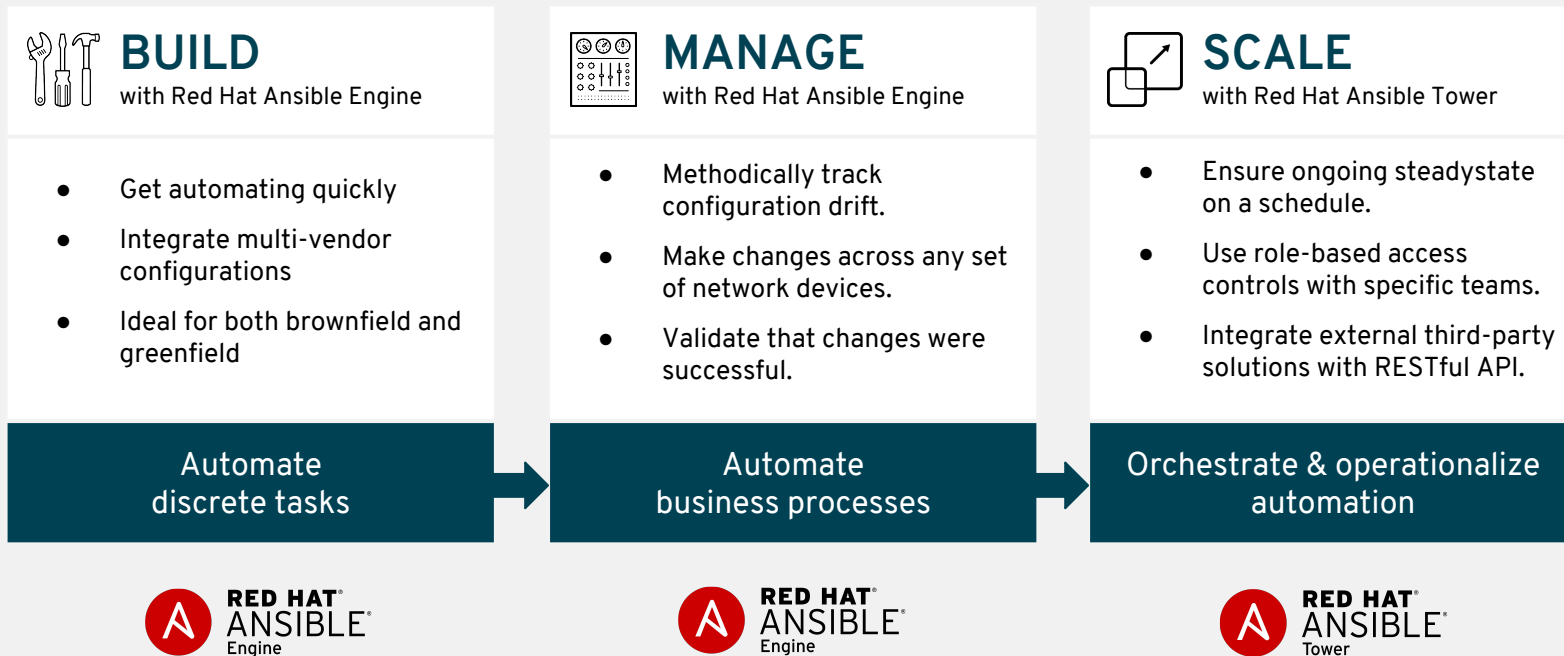
Any client system (server, laptop, VM) running Linux or Mac OSX

Managed Nodes (Inventory):

A collection of endpoints being managed via SSH or API.

Ansible - NETWORKING AUTOMATION

Configure, validate, & ensure continuous compliance for physical network devices



NETWORK AUTOMATION WITH ANSIBLE

PLAN

Ansible automation baked in as part of design and architecture

BUILD

Build engineers use Ansible (jobs/playbooks) to set up new configurations across multi-vendor networking platform, operating systems, private and public cloud.

RUN

Network Operations use Ansible to run or schedule jobs for specific use cases eg. scan

Heterogenous mix of networking vendors

ARISTA



JUNIPER
NETWORKS

cumulus networks

paloalto
NETWORKS

CITRIX
NetScaler

Decoupling the data model from the execution enables:

Multi-vendor orchestration

Enhanced network operational model through scalable, repeatable patterns

Integration into agile DevOps workflows



Ansible Playbooks

Define Intent, Policy, Architecture



Execution

Apply across device type, vendors

NETWORK SERVICE AUTOMATION EXAMPLE

Data Center and WAN Network Devices. > 10,000 Multi-vendor Devices – NXOS, JunOS, IOS-XR, EOS

Before Red Hat Automation Tools:

- Manual operator translation of daily changes to security (ACL) rules
- Not Compliant with Operational Business Rules

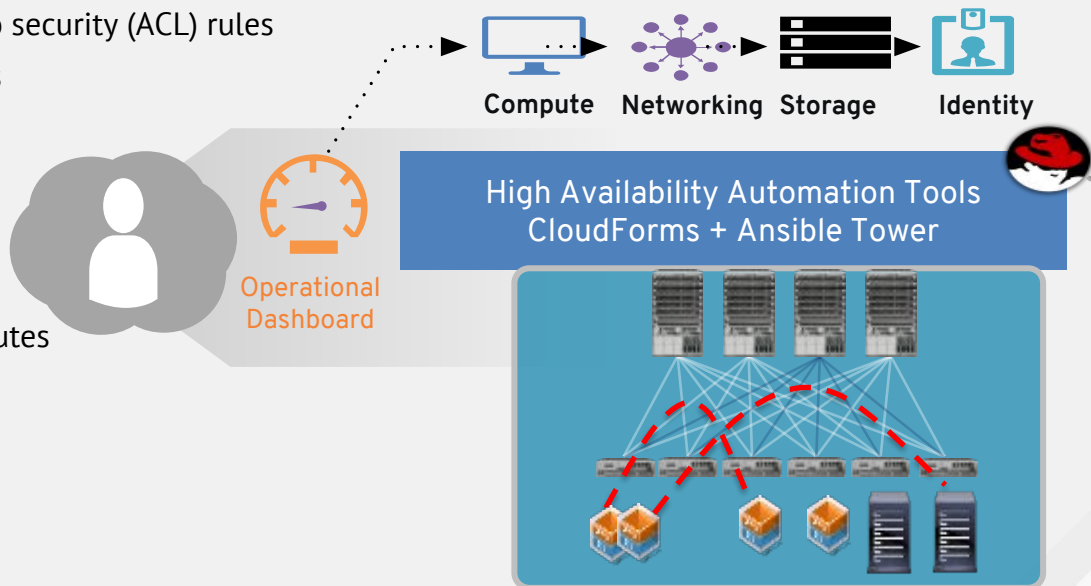
After:

- Ansible Tower + CloudForms
- Automated Rule Playbooks
- Daily Remediation of Business Rules
- Hours to Provision One Device Reduced to Minutes

Key Benefits:

- Fully Automated, Open, Secure
- Saves Time – More Productive Teams
- Fewer Mistakes / Errors
- Significant Reductions in Complexity
- Apply Across Other Service Deployments

Faster Deployments -> Lower Costs



SDN vs Automation (Ansible)

BENEFIT	SDN	AUTOMATION
Reconfigure the network from a central point	✓	✓
Reduced vendor lock in with commodity hardware	?	✓
Leverage existing infrastructure	✗	✓
Decouple control plane from data plane	✓	✓
Reduced opex/capex costs	?	✓

TRANSFORMING THE ENTERPRISE



**PHYSICAL NETWORK
STATE VALIDATION**



**DATA CENTER
FABRIC AUTOMATION**



**HYBRID CLOUD
NETWORK INTERCONNECT**

40

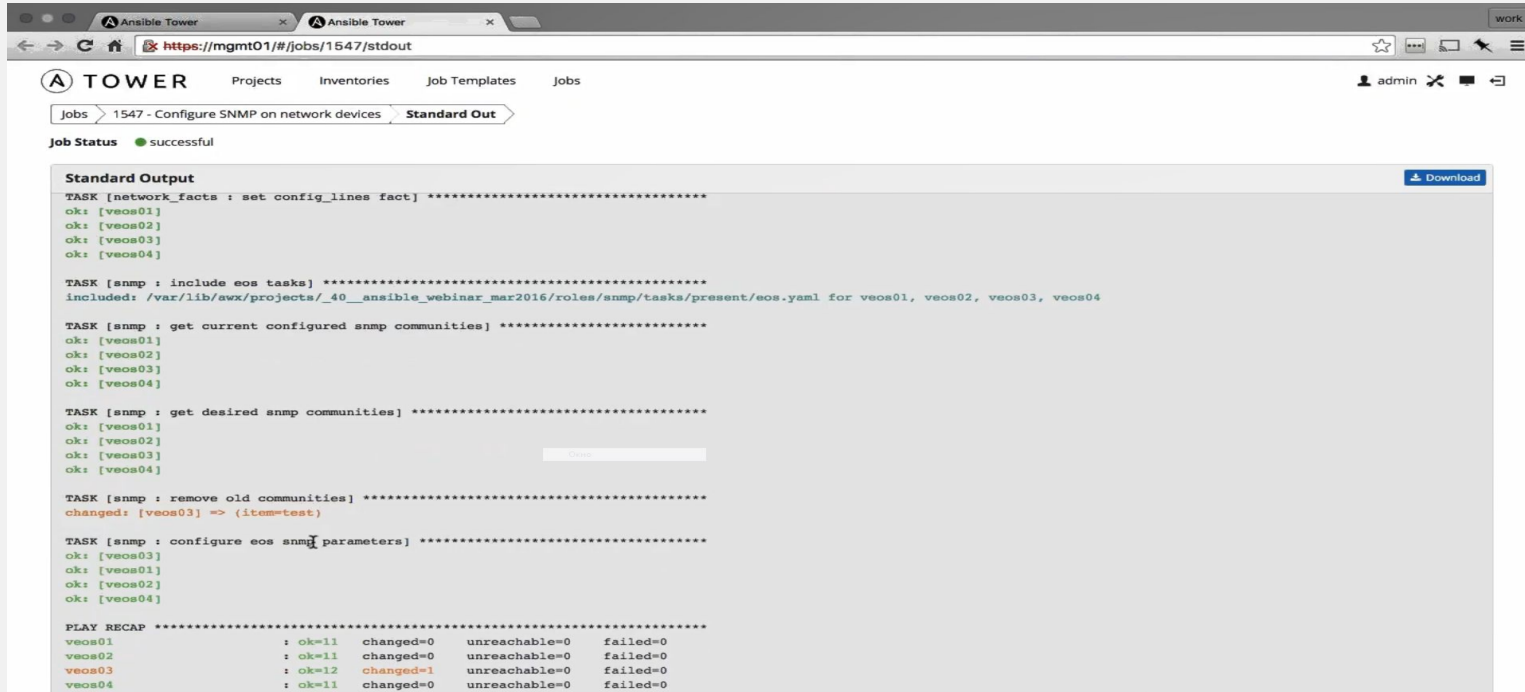
Networking
platforms

570+

Networking
modules

ansible.com/networking

Ansible Tower - UI



The screenshot displays the Ansible Tower web interface. The browser address bar shows the URL `https://mgmt01/#/jobs/1547/stdout`. The navigation menu includes 'TOWER', 'Projects', 'Inventories', 'Job Templates', and 'Jobs'. The user 'admin' is logged in. The breadcrumb trail is 'Jobs > 1547 - Configure SNMP on network devices > Standard Out'. The job status is 'successful'. The main content area shows the 'Standard Output' of the job, which includes the following tasks and their results:

```
TASK [network_facts : set config_lines fact] *****
ok: [veos01]
ok: [veos02]
ok: [veos03]
ok: [veos04]

TASK [snmp : include eos tasks] *****
included: /var/lib/awx/projects/_40_ansible_webinar_mar2016/roles/snmp/tasks/present/eos.yaml for veos01, veos02, veos03, veos04

TASK [snmp : get current configured snmp communities] *****
ok: [veos01]
ok: [veos02]
ok: [veos03]
ok: [veos04]

TASK [snmp : get desired snmp communities] *****
ok: [veos01]
ok: [veos02]
ok: [veos03]
ok: [veos04]

TASK [snmp : remove old communities] *****
changed: [veos03] => (item=test)

TASK [snmp : configure eos snmp parameters] *****
ok: [veos03]
ok: [veos01]
ok: [veos02]
ok: [veos04]

PLAY RECAP *****
veos01      : ok=11  changed=0  unreachable=0  failed=0
veos02      : ok=11  changed=0  unreachable=0  failed=0
veos03      : ok=12  changed=1  unreachable=0  failed=0
veos04      : ok=11  changed=0  unreachable=0  failed=0
```

<https://www.ansible.com/resources/webinars-training/automating-your-network>

Ansible Tower - UI

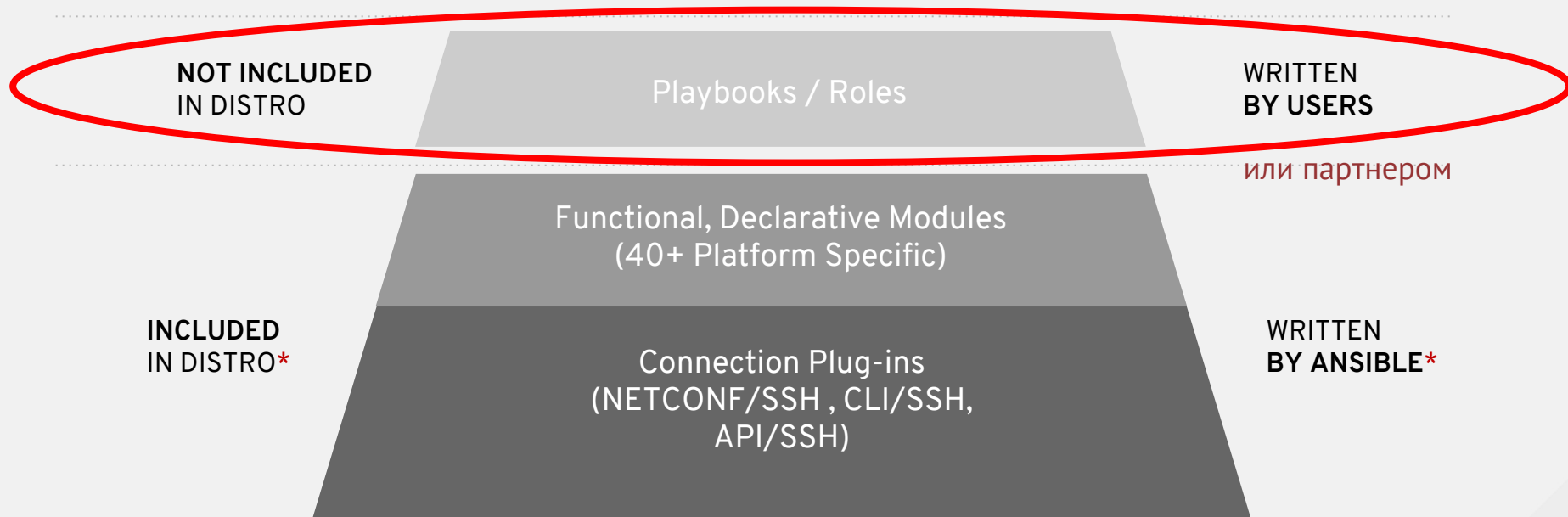
The screenshot shows the Ansible Tower web interface. The browser address bar displays `https://mgmt01/#/job_templates`. The navigation menu includes 'Projects', 'Inventories', 'Job Templates', and 'Jobs'. The user is logged in as 'admin'. A search bar is located at the top left of the content area. The main content is a table of job templates:

Name	Description	Status	Actions
Ad-hoc config push	Push configuration onto a specified network host		
Check system services compliance nodes	Checks that all network nodes are in compliance with system services		
Configure base system on network devices			
Configure eAPI on Arista EOS devices	Push configure to Arista EOS nodes to enable eAPI		
Configure Loopback interfaces on network devices			
Configure NTP on network devices	Push NTP settings to all network devices		
Configure SNMP on network devices	Push SNMP settings to all network devices		
Get device names and OS versions	Collect device configured names and OS version		
Get device running-config			
Manage IP system settings			
Push NXOS node system services	Pushes system services configuration to NXOS node		

Page 1 of 1 (11 items)

<https://www.ansible.com/resources/webinars-training/automating-your-network>

Ansible for Networking



*Can also be created and maintained by the community

Ansible для Networks - Кому и зачем

- Часть DevOps и Трансформационной истории
 - Кроме контейнеров актуальна и автоматизация legacy, а Network - наиболее legacy
 - Можно дополнять Ansible для ИТ, реализовывать параллельно, или, наоборот, первым
- Альтернатива/дополнение к вендорским EMS/NMS
- Альтернатива вендорским SDN
 - Вместо закупки контроллера и доп.лицензий

Типовые сценарии применения

- Новые дата-центры и проекты модернизации существующих
- Территориально-распределенная инфраструктура с типовыми блоками устройств

Важно

- В конкурентах только 'Do Nothing' и DIY. Популярный для VM puppet несопоставим, различные SNMP средства удачно дополняются Ansible, начиная с настройки SNMP и агентов на устройстве
- Стоимость мала на фоне затрат на сетевое оборудование и обслуживающих его людей



Спасибо



plus.google.com/+RedHat



facebook.com/redhatinc



linkedin.com/company/red-hat



twitter.com/RedHatNews



youtube.com/user/RedHatVideos