

# Обзор и разбор возможностей RedHat Satellite



Материал подготовил: Барашков Кирилл Витальевич,  
Программист 2 категории отдела ПТК ГВЦ ОАО  
«РЖД»

- Назначение Satellite
- Ansible
- Reports
- Параметры



## Simplify infrastructure management

Red Hat Satellite can manage your entire infrastructure—not just Red Hat Enterprise Linux systems—from a single console.



## Implement your standard operating environment

Improve system consistency, reduce errors, and meet compliance standards.



## Scale your environment

Meet the demand for managing thousands of distributed systems across multiple datacenters.

### ∨ Content management

- **Content repository** any type of content made available to any host
- **Curation** of content prior to distribution
- **Distribution** of content as close as possible to the end point

### ∨ Patch management

- **Report** on hosts that need updates, fixes, or enhancements
- **Group** homogeneous systems so that you can easily work with them
- **Respond** quickly to patching requirements using scalable automation

### ∨ Provisioning management

- **Provision** to bare metal, virtual, private, and public clouds
- **Discover** non-provisioned hosts
- **Automate** using Ansible roles to perform post-provisioning steps

### ∨ Subscription management

- **Centrally manage** subscription usage
- **Maintain** accurate inventory and utilization information
- **Report** on subscription consumption






*Сервер Satellite является ключевой точкой всей инфраструктуры ГВЦ и собирает множество различной информации о узлах. Часто необходимо получить некую информацию с большого количества хостов. Satellite имеет встроенный Report Engine, который позволяет существенно упростить процесс сбора информации.*






# Report Engine


## Generating a report


This will generate a report Ansible - Ansible Inventory. Based on its definition, it can take a long time to process.


Generate at    


Send report via e-mail 


Output format   


Organization  CSV


Location  JSON


Host Group  **YAML**  
HTML


Host Collections  yes


Host Parameters  no


IPv4  yes


IPv6  no



Subnet  yes

Subnet v6  no

Smart Proxies  yes

Content Attributes  no

Facts  no

Hosts   



## Ключевые особенности Ansible

- **Безагентное.** В клиенте не установлено программное обеспечение или агент, который общается с сервером.
- **Идемпотентное.** Независимо от того, сколько раз вы вызываете операцию, результат будет одинаковым.
- **Простое и расширяемое.** Программа Ansible написана на Python и использует YAML для написания команд. Оба языка считаются относительно простыми в изучении.



# Структура Ansible

## Модули

Это небольшие программы, выполняющие определённую работу на сервере. Например, вместо запуска этой команды:

## Плагины

Ansible поставляется с несколькими удобными плагинами, и вы можете легко написать свой собственный.

## Инвентаризация хостов

Чтобы предоставить перечень хостов, нам нужно обозначить список, находящийся в файле инвентаризации. Он напоминает содержание файла hosts.

В простейшем виде он может содержать одну строку:

```
35.178.45.231 ansible_ssh_user=adminos
```



## **Playbooks**

Ansible playbooks — это способ отправки команд на удалённые компьютеры с помощью скриптов. Вместо того, чтобы индивидуально использовать команды для удалённой настройки компьютеров из командной строки, вы можете настраивать целые сложные среды, передавая скрипт одной или нескольким системам.

## **group\_vars**

Файл содержит набор переменных, например имя пользователя и пароль базы данных.





## **Роли**

Это способ сгруппировать несколько задач в один контейнер, чтобы эффективно автоматизировать работу с помощью понятной структуры каталогов.

## **Обработчики**

Представляют собой списки задач, которые на самом деле не отличаются от обычных задач, на которые ссылается глобально уникальное имя и которые оповещаются уведомлениями. Если ничто не уведомляет обработчик, он не будет запускаться. Независимо от того, сколько задач уведомляет обработчик, он запускается только один раз, после того как все задачи завершены.

## **Теги**

Если у вас playbook с большим объёмом, может быть полезно иметь возможность запускать только определённую часть его конфигурации.



# Различные варианты использования Ansible в Satellite

## Ansible Roles

Filter ... x Поиск [v] Импорт Справка

Имя	Группы узлов	Узлы	Imported at	Действия
demo-rtd	1	0	9 месяца н...	Переменные
infra-system.linux.isc-bind-setup	1	0	3 месяца н...	Переменные
infra-system.linux.os.auditd	2	0	6 месяца н...	Переменные
infra-system.linux.os.cockpit	2	0	6 месяца н...	Переменные
infra-system.linux.os.dnsmasq-resolver	0	0	3 месяца н...	Переменные

[Задания](#) > [Вызов задания](#)

Категория \*

Шаблон задания \*

Закладка

Search Query

Применимо к 0 узлу(ам)

action

module\_spec

## Шаблоны заданий

Filter ... x Поиск [v] Справка Импорт Создать шаблон задания

Имя	Фрагмент	Заблокировано	Действия
<a href="#">Ansible Roles - Ansible Default</a>		🔒	Выполнить
<a href="#">Ansible Roles - Install from Galaxy</a>		🔒	Выполнить
<a href="#">Ansible Roles - Install from git</a>		🔒	Выполнить
<a href="#">Ansible Roles - Mpuer Default</a>			Выполнить
<a href="#">Ansible - Run insights maintenance plan</a>		🔒	Выполнить
<a href="#">Ansible - Run playbook</a>		🔒	Выполнить

## Ansible Variables

Filter ... x Поиск [v] Импорт New Ansible Variable Справка

Name	Role	Тип	Imported?	Действия
dns_domain	infra-system.lin...	string	✓	Удалить
dns_search	infra-system.lin...	array	✓	Удалить
nameserver	infra-system.lin...	array	✓	Удалить
test_var	test-var	string		Удалить

100 per page 1-4 of 4 << < 1 of 1 > >>



*Ansible – это система управления конфигурации, которая преследует принцип идемпотентности. Для выполнения данного принципа необходимо избегать ручной модификации хоста, находящегося под управлением системы.*



# Параметры

Уровни иерархии параметров:

- **Глобальные параметры**

Параметры, которые применяются к узлу в целом.

- **Параметры домена**

Эти параметры переопределяют глобальные параметры и применяются на уровне домена.

- **Параметры группы узлов**

Эти параметры переопределяют глобальные параметры и параметры домена и применимы ко всем узлам в составе заданной группы.

- **Параметры узла**

Эти параметры применяются к конкретному узлу и переопределяют все вышеперечисленные параметры.

Сами параметры подразделяются на несколько категорий:

- **Простые параметры**

Пара «ключ=значение».

- **Смарт-параметры**

Сложный параметр, который помимо пары «ключ=значение» использует дополнительные аргументы, условия и проверки.

- **Параметризованные классы**

Параметры классов, импортированных с мастер-сервера Puppet.