

# Управление конфигурацией

---

15 апреля 2026 г.

### Наш CI/CD pipeline работает:

- Git push → автоматический деплой через GitHub Actions
- Приложение деплоится через Docker Compose

### Но есть проблема:

Сервер настраивается **полуавтоматически**

### Проблемы (полу)ручной настройки серверов:

- **Ошибки** — человеческий фактор при настройке
- **Дрейф конфигурации** — серверы постепенно расходятся в настройках
- **Масштабирование** — нужно 10 серверов? 10 раз повторяем

**Вопрос:** как автоматизировать настройку серверов?

### Baked:

- Заранее готовим образ с настроенным ПО (руками или скриптами)
- Разворачиваем сервер из образа

**Примеры:** Docker, Packer и др.

### Fry:

- Готовим скрипты по настройке
- Каждый сервер настраивается через скрипт

**Примеры:** Bash скрипты, Ansible, Chef, и др.

**Ansible** — инструмент для автоматизации настройки серверов

## Основные особенности:

- **Agentless** — не нужно ставить агенты на сервера
- **SSH** — использует обычное SSH соединение
- **YAML** — конфигурация в простом формате
- **Идемпотентность** — можно запускать много раз безопасно

**Принцип работы:** описываем желаемое состояние сервера в файлах, Ansible приводит сервер к этому состоянию

**Inventory** — список серверов для управления

- IP адреса или доменные имена
- Группировка серверов (веб-сервера, БД, и т.д.)

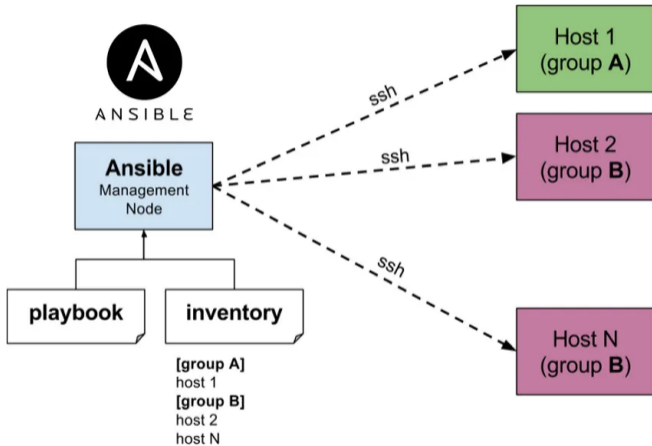
**Playbook** — сценарий действий в YAML формате

- Последовательность задач (tasks)
- Применяется к группам серверов из inventory

**Modules** — готовые блоки для типовых задач

- apt, yum — установка пакетов
- copy, template — работа с файлами
- service — управление сервисами

# Ansible архитектура



## Что посмотрим:

1. Установка Ansible
2. Ansible адhoc
3. Ansible Playbook

## Демо-репозиторий:

<https://github.com/prafdin/devops-course-demos/tree/ansible>

## Что посмотрим:

- Статический и динамический inventory файл
- Плейбук для подготовки сервера к деплою

## Демо-репозиторий:

<https://github.com/prafdin/devops-demo-website/tree/ansible-demo>