

# Вопросы к зачету «Открытые информационные системы. DevOps»

4 марта 2026 г.

## 1 Что такое DevOps?

1. Что такое DevOps и какие проблемы он решает в современной разработке ПО?
2. Каковы ключевые цели и принципы DevOps?
3. В чем заключается роль DevOps-инженера и как она соотносится с культурой DevOps в команде?
4. Приведите примеры кейсов, демонстрирующих необходимость внедрения DevOps практик.

## 2 Веб-сервер, прокси и туннелирование

1. Что такое веб-сервер и каков его основной принцип работы?
2. Объясните разницу между forward proxy и reverse proxy. Приведите примеры использования каждого.
3. Что такое туннелирование и какие задачи оно позволяет решать?
4. Как работает SSH-туннелирование? Опишите сценарий его применения.
5. Что такое FRP (Fast Reverse Proxy) и какие задачи решает этот инструмент?

## 3 Git в DevOps

1. Какую роль занимает Git в DevOps?
2. Почему в DevOps-репозитории хранится не только исходный код, но и другие артефакты? Приведите примеры.
3. Как Git используется в качестве триггера для автоматизации?

## 4 CI/CD пайплайн

1. Что такое CI/CD пайплайн и из каких типичных шагов он состоит?
2. Сравните CI и CD. Каковы цели и задачи каждого из этих процессов?
3. Назовите основные компоненты GitHub Actions (Workflow, Job, Step, Action, Runner).
4. Какие преимущества дают готовые CI/CD платформы (Jenkins, GitHub Actions, GitLab CI) по сравнению с собственными решениями?

## 5 Управление конфигурацией

1. Какие проблемы решает автоматизация управления конфигурацией?
2. Сравните два подхода к управлению конфигурацией: "Baked" (образы) и "Fry" (скрипты). Назовите их плюсы и минусы.
3. Что такое Ansible и каковы его ключевые особенности (agentless, SSH, YAML, идиempotentность)?
4. Что такое "дрейф конфигурации" и как инструменты управления конфигурацией помогают его избежать?

## 6 Контейнеризация

1. Что такое контейнеризация и какие проблемы она решает?
2. Сравните контейнеры и виртуальные машины. В чем их ключевые различия?
3. Назовите основные понятия Docker: Image, Container, Dockerfile. Опишите их взаимосвязь.
4. Что такое Docker Registry и зачем он нужен?

## 7 Docker Compose

1. Что такое Docker Compose и какую проблему он решает?
2. Каковы преимущества использования Docker Compose для управления многоконтейнерными приложениями?
3. Назовите основные команды Docker Compose для управления жизненным циклом приложения.
4. Как с помощью Docker Compose можно масштабировать отдельный сервис?

5. Какие существуют альтернативы Docker Compose для оркестрации контейнеров?

## 8 Packer

1. Что такое Packer и для чего он используется?
2. Объясните, как работает Packer. Опишите его основные компоненты (Builder, Provisioner, Template).
3. Что такое Virtual Appliance?
4. Как Packer интегрируется в CI/CD пайплайн для автоматизации создания образов?

## 9 Cloud computing

1. Что такое облачные вычисления (Cloud computing)?
2. Сравните модели облачных вычислений: On-Premise, IaaS, PaaS, SaaS.
3. В чем разница между публичным и приватным облаком? Приведите примеры провайдеров.
4. Назовите основные преимущества использования облачных технологий.

## 10 Infrastructure as Code

1. Что такое Infrastructure as Code (IaC)?
2. Какие преимущества дает IaC по сравнению с ручным управлением инфраструктурой?
3. Что такое Terraform и для чего он используется?
4. Опишите основной рабочий процесс в Terraform (init, plan, apply).