

**ИСПОЛЬЗОВАНИЕ АНАЛИЗА СЕТЕЙ SDN НА
ЗАНЯТИЯХ ТЕХНИЧЕСКИХ ДИСЦИПЛИН****Мурадова Алевтина Александровна**Ташкентский университет информационных технологий имени
Мухаммада аль-Хорезми, PhD, доцент кафедры
«Телекоммуникационный инжиниринг», a.muradova1982@inbox.ru**Абдужаппарова Мубарак Балтабаевна**Ташкентский университет информационных технологий имени
Мухаммада аль-Хорезми, PhD, доцент, заведующая кафедры
«Телекоммуникационный инжиниринг»,
mubarakabd846@gmail.com**Каримова Говхар Абдухалик кизи**Ташкентский университет информационных технологий имени
Мухаммада аль-Хорезми, стажёр-преподаватель кафедры
«Инфокоммуникационный инжиниринг»
<https://doi.org/10.5281/zenodo.17809505>**ARTICLE INFO**Received: 27th November 2025Accepted: 28th November 2025Online: 29th November 2025**KEYWORDS***Software-Defined Networking; SDN; анализ сетей; виртуальные топологии; сетевое моделирование; Mininet; SDN-контроллер; OpenFlow; сетевые технологии; технические дисциплины; сетевой трафик; мониторинг сети; сетевые протоколы; автоматизация сети; образовательные лаборатории***ABSTRACT**

В статье рассматриваются возможности применения анализа сетей с программно-определяемой архитектурой (SDN) в процессе преподавания технических дисциплин. Подчеркивается, что использование SDN позволяет повысить качество обучения за счет интеграции современных сетевых технологий, обеспечивающих гибкое управление трафиком, централизованный контроль и глубокую аналитическую обработку сетевых данных. Приводится обзор ключевых инструментов и платформ, таких как Mininet, Ryu и OpenDaylight, которые позволяют моделировать виртуальные топологии и исследовать поведение сетевых протоколов в лабораторных условиях. Анализируются педагогические преимущества внедрения SDN: формирование практических навыков у студентов, развитие компетенций в области сетевой автоматизации, программирования и мониторинга сетей. Делается вывод о том, что использование SDN-аналитики способствует переходу к современным образовательным практикам и формирует у обучающихся готовность к работе с реальными сетевыми инфраструктурами нового поколения.

Введение. Software-Defined Networking (SDN) - это архитектура, в которой управление сетью отделено от ее физической инфраструктуры. SDN активно используется в современных дата-центрах, облачных службах и операторами связи. В образовательном процессе SDN открывает уникальные возможности: на занятиях

студенты могут не только изучать теорию маршрутизации и коммутации, но и практически моделировать сетевое поведение, управлять трафиком и применять аналитические инструменты.

Современное инженерное образование стремительно развивается под влиянием цифровизации, роста сетевой инфраструктуры и широкого внедрения технологий программно-определяемых сетей (SDN — Software-Defined Networking). SDN постепенно становится ключевым элементом современных телекоммуникационных систем, облачных платформ и центров обработки данных, что требует от будущих специалистов глубокого понимания принципов управления сетями, анализа трафика и взаимодействия сетевых ресурсов. В этих условиях интеграция анализа SDN в образовательный процесс технических дисциплин приобретает стратегическое значение [1].

Использование SDN на занятиях позволяет переходить от традиционной модели обучения к практико-ориентированным методам, основанным на моделировании сетевых процессов, выполнении лабораторных работ в виртуальных средах и исследовании реальных сценариев функционирования сетевой инфраструктуры. Такой подход обеспечивает формирование у студентов не только теоретических знаний, но и профессиональных компетенций в области управления сетями, автоматизации конфигураций и применения аналитических инструментов для оценки сетевой эффективности. Дополнительным преимуществом является развитие у обучающихся навыков критического мышления, системного подхода и умения работать с современными сетевыми платформами, такими как OpenFlow-контроллеры, Mininet, Ryu или ONOS. Включение анализа SDN в образовательные программы способствует адаптации студентов к реальным требованиям отрасли и повышает их конкурентоспособность на рынке труда.

Цели использования SDN в образовательном процессе- применение SDN-технологий на занятиях технических дисциплин направлено на: формирование практических навыков работы с современными сетевыми архитектурами; освоение принципов централизованного управления сетью; развитие компетенций в области сетевой автоматизации и программирования; изучение методов анализа трафика и мониторинга сетевых процессов; подготовку студентов к работе в профессиональной сфере IT и телекоммуникаций [2].

Методы. Основные методы обучения с использованием SDN. 1. Моделирование сетей в виртуальной среде. Метод: использование симуляторов и виртуальных стендов (Mininet, EVE-NG, GNS3). Ценность: позволяет построить многослойную SDN-топологию без дорогостоящего оборудования. Операции, выполняемые студентами: создание виртуальных коммутаторов Open vSwitch; подключение к SDN-контроллеру; генерация сетевого трафика; настройка правил маршрутизации.

2. Работа с SDN-контроллерами. Метод: изучение API контроллеров (Ryu, ONOS, OpenDaylight) и взаимодействия с ними. Практика: получение статистики OpenFlow; настройка потоковых таблиц; управление сетевыми политиками через REST API.

3. Программирование сетевых приложений. Метод: внедрение мини-проектов на Python для разработки простых SDN-приложений. Результаты обучения: понимание

принципов автоматизации; навыки написания контроллерных скриптов; создание собственных механизмов маршрутизации и QoS.

4. Анализ трафика и мониторинг. Метод: сбор и исследование данных о сети. Инструменты: Wireshark, sFlow/NetFlow-мониторинг, встроенная аналитика контроллеров. Изучается: задержки, потери, пропускная способность; распределение потоков; загрузка узлов и каналов [3].

5. Исследование сетевых политик и QoS. Метод: применение приоритетов, ограничений скорости, фильтрации. Задачи студентов: анализ поведения трафика при изменении политики; оценка качества обслуживания; выявление узких мест.

Результаты. Формы проведения занятий. 1. Лабораторные работы. Наиболее эффективный формат. Примеры работ: Построение SDN-сети в Mininet. Изучение OpenFlow-правил и их влияние на маршрутизацию. Мониторинг сети через контроллер и системные утилиты. Реализация простой SDN-политики QoS [4].

2. Практикумы и тренинги. Короткие интенсивные занятия для отработки конкретных навыков: настройка API запросов; анализ логов и статистики; отладка сетевых приложений.

3. Проектная деятельность. Студенты создают собственные мини-проекты: разработка алгоритма маршрутизации; автоматизация сети на Python; визуализация SDN-топологий; анализ сетевого поведения при сбоях.

4. Организация учебного процесса. Материально-техническое обеспечение. Для проведения занятий достаточно: ПК/ноутбуков студентов; виртуализированных SDN-платформ (Mininet); установленного контроллера (Ryu как оптимальный старт); инструментов мониторинга (Wireshark и встроенные средства) [5].

Педагогические рекомендации. начинать с простых топологий и постепенно усложнять задания; сочетать теорию с немедленной практикой; демонстрировать реальный трафик, показывая работу сети «вживую»; включать сценарии анализа ошибок и необычных ситуаций; поощрять экспериментирование с правилами и политиками сети [6].

5. Практическая значимость метода. Использование анализа SDN позволяет: освоить современные сетевые подходы, востребованные в телекоммуникациях и IT; обучать студентов работе с реальными инструментами инженерной профессии; внедрять элементы DevNet/NetOps-культуры в образование; формировать компетенции, соответствующие требованиям индустрии 4.0 [7,8].

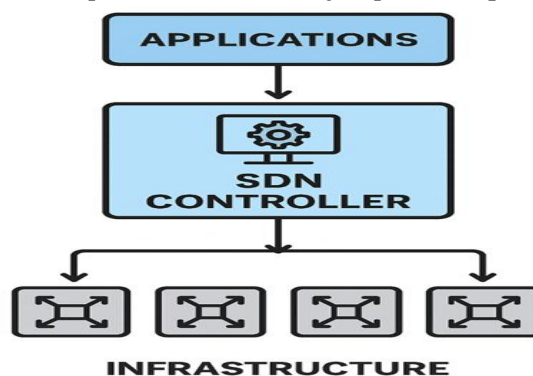


Рис.1. Структура сети SDN

Значение SDN в технических дисциплинах/ Почему SDN стало актуальным: растущая сложность сетевых инфраструктур. Появление виртуализации и облаков. Необходимость централизованного управления. Высокие требования к гибкости и аналитике сети.

Что даёт SDN в учебном процессе: Практика работы с контроллерами (OpenDaylight, ONOS, Ryu). Изучение сетевой автоматизации. Построение виртуальных топологий (Mininet). Проведение анализа трафика и логов. Реализация собственных алгоритмов маршрутизации [9,10].

Анализ сетей SDN в учебной среде

Основные направления анализа: 1) Аналитика трафика: анализ потоков OpenFlow, оценка загрузки каналов, выявление узких мест (bottlenecks), применение инструментов: Wireshark, sFlow, NetFlow. 2) Мониторинг состояния сети: метрики задержки, потерь, пропускной способности, контроль работы коммутаторов и линков, использование REST API SDN-контроллера. 3) Анализ поведения приложений: как приложения управляют сетью, тестирование политик QoS, настройка приоритетов данных.

4) Анализ безопасности SDN: обнаружение аномалий, анализ атак типа DoS, spoofing (информативно, без опасных деталей), обучение работе с сетевыми политиками безопасности [11,12].

Обсуждение. Лабораторные занятия с анализом SDN. Примеры учебных работ: 1. Построение SDN-топологии в Mininet: Создание виртуальной сети, Подключение к контроллеру Ryu. 2. Анализ сетевых потоков OpenFlow: Просмотр таблиц потоков, Изменение правил маршрутизации. 3. Мониторинг SDN-сети: Получение статистики через REST API, Визуализация трафика. 4. Эксперимент по QoS в SDN: Ограничение скорости, Фильтрация и маркировка трафика. 5. Анализ отказоустойчивости: имитация отказа узлов, изучение реакции контроллера. 6. Педагогические преимущества использования SDN. Для студентов: более глубокое понимание сетевых процессов, развитие навыков моделирования, работа с современными инструментами, подготовка к реальной инженерной практике. Для преподавателя: возможность демонстрации сложных процессов на практике, создание гибких лабораторных стендов, развитие проектной и исследовательской деятельности студентов. 7. Рекомендации по внедрению SDN в обучение: использовать виртуальные лаборатории (Mininet, GNS3, EVE-NG), включать проектные задания, применять открытые контроллеры для обучения (Ryu — лучший старт), вводить базовые уроки по Python для написания SDN-приложений, комбинировать теорию с практическим анализом логов и трафика [13].

Заключение. Использование анализа SDN на занятиях технических дисциплин делает обучение более современным, практикоориентированным и востребованным. Студенты учатся понимать архитектуру сетей будущего, управлять трафиком, выполнять мониторинг и настроить сетевые политики — навыки, которые очень востребованы в ИТ и телекоммуникациях. Использование технологий SDN в образовательном процессе значительно повышает качество подготовки специалистов в области сетевых технологий, поскольку позволяет изучать современные

архитектуры и механизмы сетевого управления, применяемые в индустрии. Централизованная модель управления SDN предоставляет студентам уникальную возможность исследовать сетевые процессы на уровне контроллера и понимать взаимосвязи между приложениями, политиками и сетевым поведением.

Анализ SDN-сетей, включающий мониторинг трафика, изучение таблиц потоков, QoS-политик и сетевой телеметрии, способствует развитию инженерных навыков, необходимых для работы в современной телекоммуникационной инфраструктуре и облачных средах. Виртуальные лаборатории (Mininet, Ryu и др.) обеспечивают доступную и гибкую среду для моделирования сложных топологий без необходимости использования дорогостоящего оборудования. Интеграция SDN в учебные программы укрепляет связь теории и практики, формируя у студентов способность решать реальные инженерные задачи: от построения сетей до разработки собственных SDN-приложений.

Рекомендации. Постепенно вводить SDN в учебные курсы, начиная с базовых понятий централизованного управления сетью и OpenFlow, постепенно переходя к сложным практическим задачам и разработке приложений. Использовать виртуальные среды моделирования, такие как Mininet, для регулярного проведения лабораторных работ, позволяя студентам самостоятельно строить топологии, изучать поведение трафика и тестировать сетевые политики. Применять открытые SDN-контроллеры (Ryu, ONOS, OpenDaylight) как основную учебную платформу для лабораторных занятий и проектных работ. Развивать навыки программирования у студентов, включая написание простых SDN-приложений на Python, что способствует пониманию принципов автоматизации и алгоритмизации сетевого управления. Включать проектные и исследовательские задания, такие как разработка собственных механизмов маршрутизации, построение визуализации SDN-топологий, анализ поведения сети в условиях отказов и нагрузок. Интегрировать инструменты анализа трафика (Wireshark, sFlow/NetFlow) для формирования навыков мониторинга и диагностики сетевых состояний. Поддерживать междисциплинарный подход, связывая SDN с кибербезопасностью, облачными технологиями, виртуализацией и DevNet-практиками

Список литературы:

1. Кузнецов, А. В., & Егоров, В. Н. Программно-определяемые сети: архитектура, протоколы и решения. Москва: Горячая Линия – Телеком, 2020.
2. Климов, С. Ю., & Харитонов, Д. В. Архитектура и управление SDN/OpenFlow. // Журнал «Сети и системы связи», №4, 2019.
3. Sezer, S., Scott-Hayward, S., et al. Are We Ready for SDN? Implementation Challenges and Opportunities. // IEEE Communications Magazine, 51(11), 2013.
4. Kreutz, D., Ramos, F., Verissimo, P. Software-Defined Networking: A Comprehensive Survey. // Proceedings of the IEEE, 103(1), 2015.
5. ONF (Open Networking Foundation). Software-Defined Networking Architecture. ONF TR-521, 2022.
6. Feamster, N., Rexford, J., Zegura, E. The Road to SDN: An Intellectual History of Programmable Networks. // ACM SIGCOMM, 2014.

7. Goransson, P., Black, C., & Culver, T. Software Defined Networks: A Comprehensive Approach. 2nd ed. Morgan Kaufmann, 2020.
8. Михеев, А. В. Виртуальные лаборатории для обучения сетевым технологиям на базе Mininet. // Вестник технического университета, №7, 2020.
9. SdnHub. SDN Tutorials and Labs (Mininet, POX, Ryu, ONOS). 2023. (Учебные материалы для практических занятий).
10. ONOS Project Documentation. ONOS SDN Controller Architecture and Tutorials. 2024. Ryu SDN Framework. Official Developer Guide. 2023.
11. R. Braga, E. Mota, A. Passito. Lightweight DDoS Flooding Attack Detection Using NOX/OpenFlow. IEEE LCN, 2010.
12. Горошевский С.В. Программно-определяемые сети: архитектура и технологии SDN/OpenFlow. Учебное пособие.
13. Медведев С., Гаврилов А. Программно-определяемые сети и виртуализация функций сетей (NFV). Телекоммуникации, 2019

