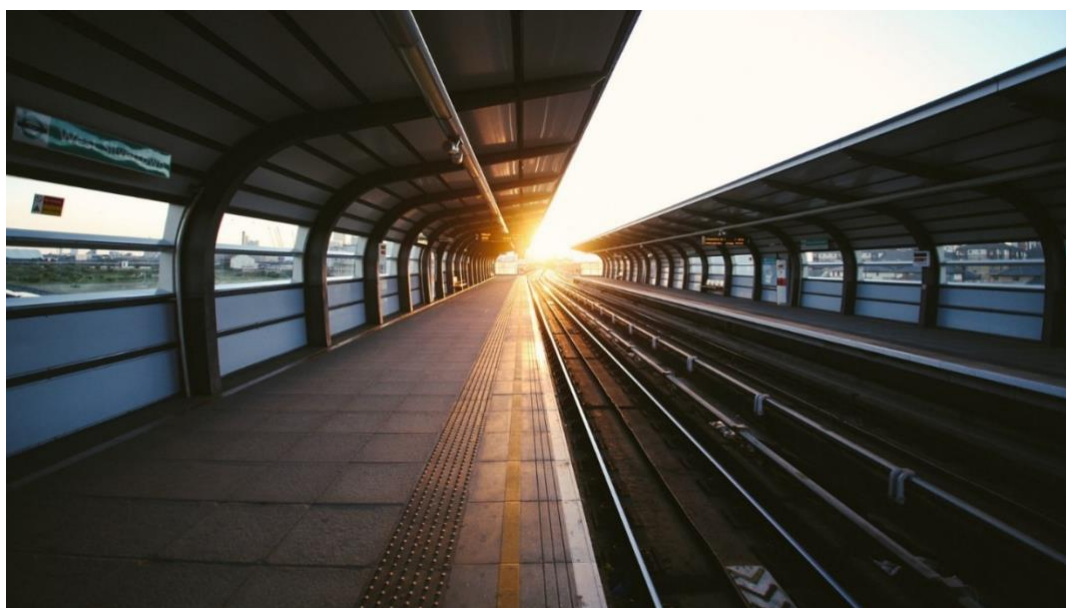


## РУКОВОДСТВО АДМИНИСТРАТОРА

---

# B4N SERVICE PLATFORM

Версия 2.3



# СОДЕРЖАНИЕ

1.	ВВЕДЕНИЕ .....	4
1.1.	О компании .....	5
1.2.	Назначение B4N Service Platform .....	6
1.3.	Условные обозначения в документе .....	6
1.4.	Термины и определения .....	6
1.5.	Обратная связь .....	9
2.	СОСТАВ ПЛАТФОРМЫ .....	10
2.1.	B4N Controller .....	11
2.2.	B4N Switch OS .....	11
2.2.1.	B4N SwitchOS для x86 платформы .....	12
2.2.2.	B4N SwitchOS для bare-metal коммутаторов .....	12
2.3.	B4N Orchestrator .....	12
2.3.1.	Оркестрация вычислительных ресурсов .....	13
2.3.2.	Оркестрация сетевыми ресурсами .....	13
2.4.	Сквозная оркестрация (VNF Manger) .....	14
3.	РАБОТА С СИСТЕМОЙ .....	15
3.1.	Установка платформы .....	16
3.2.	Референсная архитектура платформы .....	17
3.3.	Архитектура системы .....	17
3.4.	Настройка и мониторинг отказоустойчивости .....	18
4.	БАЗОВЫЕ НАСТРОЙКИ ПЛАТФОРМЫ .....	19
4.1.	Учетная запись .....	20
4.2.	Администрирование .....	20
4.2.1.	Метрики .....	20
4.2.2.	Состояние (Health) .....	21
4.2.3.	Конфигурация .....	21
4.2.4.	Управление аккаунтами .....	21
4.2.5.	Описание ролей .....	22
4.2.6.	Аудит .....	23
4.2.7.	API .....	23
4.3.	Язык интерфейса .....	23
4.4.	Feedback .....	23
4.5.	Страница по умолчанию .....	23
5.	НАСТРОЙКА ИНФРАСТРУКТУРЫ .....	25
5.1.	Создание доменов .....	25
5.2.	Создание дата-центров .....	25
5.3.	Вычислительные ресурсы .....	26
5.3.1.	Создание VIM .....	27
5.4.	Системные ресурсы .....	28
5.4.1.	Настройка интерфейса к системе мониторинга .....	28
5.4.2.	Настройка интерфейсов к VNFM .....	29
6.	СЕТЕВЫЕ РЕСУРСЫ .....	30
6.1.	Регистрация контроллеров .....	30
6.2.	Создание сервисов .....	31
6.2.1.	Узлы .....	32
6.2.2.	Коммутаторы .....	32

6.2.3.	Настройка коммутатора .....	34
6.2.4.	Сбор информации о коммутаторе .....	34
6.2.5.	Топология .....	35
6.2.6.	Сегменты .....	36
6.2.7.	QoS правила .....	36
6.2.8.	P2P туннели .....	39
6.2.9.	P2M туннели .....	40
6.2.10.	M2M туннели .....	42
6.2.11.	IP multicast источники .....	43
6.2.12.	L3 сервис .....	44
6.2.13.	Фильтры .....	46
6.2.14.	Сервис TAP (Test Access Point) .....	47
6.2.15.	Группы портов .....	48
6.2.16.	Линки .....	48
6.2.17.	Логгирование .....	49
6.2.18.	SNMP .....	50
6.2.19.	Media Fabric .....	52
7.	КАТАЛОГИ .....	53
7.1.	Наполнение каталога VNF .....	53
7.2.	Наполнение каталога шаблонами сервисов .....	54
8.	МОНИТОРИНГ .....	56
9.	ЖУРНАЛ .....	58
10.	АРЕНДАТОРЫ, ПОЛЬЗОВАТЕЛИ РЕСУРСОВ ПЛАТФОРМЫ .....	60
10.1.	Развёртывание логической топологии .....	61
11.	РЕЗЕРВНАЯ КОПИЯ. ПОИСК НЕИСПРАВНОСТЕЙ .....	62
11.1.	Создание резервной копии и восстановление (backup & restoring) .....	62
11.1.1.	Использование Web .....	62
11.1.2.	Использование интерфейса командной строки .....	62
11.2.	Поиск неисправностей (troubleshooting) .....	63
11.2.1.	Offline механизм .....	63
11.2.2.	Использование сторонних систем управления .....	63
11.2.3.	Использование интерфейса командной строки .....	63

# 1. ВВЕДЕНИЕ

Поддерживая множество клиентов в пределах сети агрегации города или района, сервис провайдеры (SP) стремятся расширить портфель предложений услуг Carrier Ethernet (CE) (в большинстве случаев это L2VPN, L3VPN, PW и Multicast сервисы). Сервис провайдеры, предлагающие сегодня услуги CE для конечных заказчиков или других провайдеров CE, могут воспользоваться услугами виртуальных сетей и облачными сервисами, чтобы дополнить существующие наборы сервисов. Другими словами, Carrier Ethernet обеспечивает базовую связь с виртуальными сетевыми сервисами, расположенными поверх. Реализация сетевых сервисов часто связана с многочисленными сложными и ручными процессами, а управление сервисами часто требует координации многих различных аппаратных и/или программных систем.

Для сокращения цикла реализации нового сервиса требуется автоматизация. Для упорядочения и расширения операций требуются новые подходы к использованию новых технологий при поддержке существующих бизнес-процессов. Таким образом, для поставщиков услуг сегодня становятся актуальными автоматизация, гибкость в обслуживании и сквозное сетевое управление. Эволюция мобильных сетей и переход на инфраструктуру мобильных сетей 5-го поколения предъявляют новые требования к транспортной сети, такие как низкая задержка, высокая пропускная способность, разбиение сети и поддержка мобильных пограничных служб и дезагрегированных архитектур EPC и IMS (разделение плоскостей управления и данных для сетевых функций). В этом случае технология программно-конфигурируемых сетей (SDN) и технология виртуализации сетевых функций (NFV) нацелены на то, чтобы помочь сервис провайдеру построить гибкую и ориентированную на сервисы сетевую инфраструктуру, которая представляет все сетевые ресурсы как единый пул абстрактных ресурсов, целостно управляемых централизованной системой (LSO). Такой подход позволяет поставщику услуг динамически распределять сетевые ресурсы, быстро создавать и управлять подключением и облачными сервисами, обеспечивая при этом сквозную гарантию обслуживания, видимость и управление. NFV, в сочетании с SDN и LSO, обеспечивает гибкость обслуживания и быстрое внедрение услуг, которые помогают создавать возможности дифференциации обслуживания и получения прибыли.

Используя этот подход, Brain4Net разработала целостную программную платформу B4N Service Platform, которая обеспечивает следующие основные функции:

- Автоматизацию предоставления услуг Carrier Ethernet – EVC, E-Tree, E-LAN, поддержку многоуровневого QoS, а также поддержку 802.1ag L2 OAM-сервиса (Operations, Administration & Management);
- Формирование в сетях Metro Access/Aggregation Network и центрах обработки данных гибкой и отказоустойчивой унифицированной сетевой матрицы, состоящей из физических элементов сети – bare-metal коммутаторов и виртуальных коммутаторов облачной среды;
- Централизованное управление трафиком для достижения максимального использования сети и оптимального распределения нагрузки между сетевыми путями, избегая при этом перегруженных каналов. Поддерживаются автоматически рассчитанные пути (основанные на полосе пропускания) или основанные на ограничениях трафика управляемые пути;
- Централизованное управление сетевой инфраструктурой (вместо пошаговых конфигураций);
- Включение виртуализированных сетевых сервисов в облаке сервис провайдера и на сетевой инфраструктуре за пределами центра обработки данных для обеспечения комплексного управления услугами.

Решение на базе платформы обслуживания B4N обеспечивает ускорение сроков выхода на рынок новых услуг, увеличение коэффициента удержания клиентов (Retention Rate), расширение портфеля услуг и создание гибкой и адаптивной сетевой инфраструктуры, позволяя операторам:

- Сократить капитальные затраты за счёт использования в Metro сетях bare-metal коммутаторов из ODM вместо дорогостоящих и излишне функциональных MPLS-маршрутизаторов;
- Снизить OPEX за счёт автоматизации предоставления сетевых услуг, комплексного управления услугами и унификации оборудования;
- Использовать эффективные стратегии перекрёстных продаж путём включения услуг VPN-клиентов с добавленными VNF;
- Реализовать стратегию "цифровизации" и "трансформации сети" наиболее эффективным с точки зрения затрат способом.

Сетевая виртуализация – один из наиболее востребованных современных трендов, который тесно связан с такими глобальными трендами, как 5G (сети пятого поколения), IoT (интернет вещей), AR/VR (виртуальная / дополненная реальность).

Решение базируется на современных технологиях виртуализации и использует программно-определяемые сети (SDN) и виртуальные сетевые функции (VNF) для реализации запрошенного функционала различных элементов узла передачи данных.

Решение создано в соответствии с рекомендациями ETSI по построению инфраструктуры сетевой виртуализации (NFVi) и позволит оператору быстро осуществить развёртывание системы, а затем поэтапно наращивать мощность узла, минимизировав первоначальные капитальные вложения и реализовав модель Pay-As-You-Grow.

Настоящий документ содержит описание средств администрирования B4N Service Platform – продукта Brain4Net. Решения на базе B4N Service Platform могут состоять из нескольких аппаратных и программных компонентов, включая bare-metal коммутаторы с открытым исходным кодом от технологических партнёров, серверного оборудования и платформ виртуализации (OpenStack, VMware vSphere). В данном документе содержится информация, описывающая принципы работы с программным продуктом, его функциональными возможностями, а также принципами настройки. Для эффективного использования B4N Service Platform рекомендуется изучить данный документ полностью. Это поможет понять основные приёмы работы и приобрести необходимые навыки для администрирования B4N Service Platform. Документ адресован администраторам B4N Service Platform и конечным пользователям. Каждый сможет найти в нём ответы на вопросы, соответствующие его компетенции. Если в документе не найдётся ответ на интересующий вопрос, напишите в службу [технической поддержки](#) продукта, и документ будет дополнен недостающей информацией.

## 1.1. О компании

ООО "ПРОГРАММИРУЕМЫЕ СЕТИ" (торговая марка Brain4Net) – компания разработчик систем управления сетями передачи данных, обработки сетевого трафика и виртуализации сервисных приложений на базе архитектур программируемых сетей SDN (Software Defined Networks) и виртуализации сетевых функций NFV (Network Function Virtualization).

В рамках проработки проектов компания оказывает для партнёров и клиентов услуги консалтинга по разработке стратегии миграции существующих сетей и интеграции решения с существующими сетями, системами виртуализации и OSS/BSS системами с целью обеспечения максимальной эффективности не только от новых инвестиций, но и от уже ранее инсталлированной базы.

Компания аккредитована Министерством Связи РФ в реестре организаций, осуществляющих деятельность в области информационных технологий. Целевыми клиентами для решений, которые разрабатывает компания, являются операторы фиксированной и мобильной связи, крупные корпоративные сети. Решения компании позволяют значительно снизить удельные капитальные вложения при развитии сетей по сравнению с применением традиционных технологий. Компания обладает уникальным составом команды для Российского рынка. Синергия сильнейшей практической инженерной экспертизы и инновационной команды прикладных разработчиков, опыт работы с крупнейшими игроками рынка сетевых технологий позволяют:

- Разрабатывать самые передовые архитектурные решения в области SDN/NFV архитектур;
- Проактивно подготавливать программную часть к моменту появления самых последних образцов аппаратного обеспечения;
- Использовать кооперацию с другими игроками российского и международного рынка для обеспечения комплексности предлагаемых заказчикам решений.

Brain4Net является активным участником MEF, ONF, OPNFV, Intel Network Builders, Open Compute Project, Telecom Infra Project. Благодаря совместному лидерству в группе MEF OpenCS NFV, Brain4Net активно делится опытом использования сервисов CE 2.0 с OpenFlow среди членов сообщества MEF.

## 1.2. Назначение B4N Service Platform

B4N Service Platform – комплексное решение для модернизации сетевой инфраструктуры операторов связи и распределённых корпоративных сетей на базе технологий SDN и NFV. Платформа включает в себя компоненты для управления и оркестрации мультивендорной сети, организации высокоскоростной обработки трафика и виртуализации сетевых функций.

## 1.3. Условные обозначения в документе

Названия элементов интерфейса (названия пунктов меню, кнопок и пр.) выделены **полужирным** шрифтом (например, **Сохранить**, **Отмена**);

Взаимоотношений между двумя фрагментами текста, при котором можно осуществлять быстрый переход от одного фрагмента к другому, помечен стилем гиперссылки (например, [ссылка](#));

Элементы кода, переменные, программные составляющие выделены **таким шрифтом**.

## 1.4. Термины и определения

Сокращение	Расшифровка
B4N Controller	Основной элемент B4N Service Platform. Реализует функции унификации и автоматизации управления OpenFlow-коммутаторами

B4N Service Platform	Комплексное решение для модернизации сетевой инфраструктуры операторов связи и распределённых корпоративных сетей на базе технологий SDN и NFV. Платформа включает в себя компоненты для управления и оркестрации мультивендорной сети, организации высокоскоростной обработки трафика и виртуализации сетевых функций
Best Effort Service	Тип обслуживания, который не является механизмом тонкого регулирования сети и является признаком простого увеличения пропускной способности без какого-либо выделения отдельных классов трафика и регулирования
Carrier Grade NAT	Решение, реализующее функции трансляции сетевых адресов в сети оператора связи. Система нацелена на решение задачи нехватки глобальных IPv4 адресов с сохранением существующей инфраструктуры оператора связи
CSP (Communications Service Provider)	Организация предоставляющая программное обеспечение, ресурсы вычислительных платформ и прочее для других юридических и физических лиц
Differentiated Service (DiffServ)	Модель обеспечивает QoS на основе распределения ресурсов в ядре сети и определённых классификаторов, и ограничений на границе сети, комбинируемых с целью предоставления требуемых услуг. В этой модели вводится разделение трафика по классам, для каждого из которых определяется свой уровень QoS
Integrated Service (IntServ)	Модель интегрированного обслуживания сети. Обеспечивает сквозное (End-to-End) качество обслуживания, гарантируя необходимую пропускную способность. IntServ использует для своих целей протокол резервирования сетевых ресурсов RSVP, который обеспечивает выполнение требований ко всем промежуточным узлам
L2VPN (Layer 2 Virtual Private Network)	Протокол туннелирования второго уровня. В компьютерных сетях туннельный протокол, использующийся для поддержки виртуальных частных сетей. Главное достоинство L2TP состоит в том, что этот протокол позволяет создавать туннель не только в сетях IP, но и в таких, как ATM, X.25 и Frame Relay
L3VPN (Layer 3 Virtual Private Network)	Протокол туннелирования третьего уровня. Сервис виртуальной частной сети, который предоставляется операторами связи наряду с L2-VPN и VPLS
LSO (Lifecycle Service Orchestration)	Представляет собой набор стандартов и продуктов, обеспечивающих оркестрацию и управление сетевым оборудованием, программным обеспечением телекоммуникационных платформ таких как системы поддержки бизнеса (BSS) и операционной поддержки системы (OSS)
MEN (Metro Ethernet Network)	Участок сети оператора связи построенная на технологии Ethernet в масштабах города. Служит для связи сетей доступа оператора с источниками сервисов, предоставляемых оператором
NFV (Network Functions Virtualization)	Технология виртуализации для переноса сетевых функций и сервисов в виде аппаратных решений, со специализированных аппаратных платформ на стандартные серверы
OpenFlow	Протокол управления процессом обработки данных, передающихся по сети передачи данных маршрутизаторами и коммутаторами, реализующий технологию программно-конфигурируемой сети



OSS/BSS (Operation Support System/Business Support System)	Система поддержки операций/система поддержки бизнеса. Общее наименование для категории прикладного программного обеспечения внутренних бизнес-процессов операторов связи
OTT (Over the Top)	Метод предоставления видеослужб через Интернет, часть технологии IPTV. Термин OTT означает доставку видеосигнала от провайдера контента на устройство пользователя (приставку, компьютер, мобильный телефон) по сетям передачи данных, часто без прямого контакта с оператором связи, в отличие от традиционных услуг IPTV, которые предоставляются как правило только через управляемую самим оператором сеть с гарантированным QoS (QoE)
QoS (Quality of service)	Определяет качество обслуживания сети. Обозначает вероятности прохождения пакета между двумя точками сети
SDN (Software-defined networking)	Программно-конфигурируемая сеть. Сеть передачи данных, в которой уровень управления сетью отделён от устройств передачи данных и реализуется программно, одна из форм виртуализации сети
VLAN (Virtual Local Area Network)	Логическая ("виртуальная") локальная компьютерная сеть, представляет собой группу хостов с общим набором требований, которые взаимодействуют так, как если бы они были подключены к ширококестельному домену, независимо от их физического местонахождения. VLAN имеет те же свойства, что и физическая локальная сеть, но позволяет конечным станциям группироваться вместе, даже если они не находятся в одной физической сети. Такая реорганизация может быть сделана на основе программного обеспечения вместо физического перемещения устройств
Bare-metal коммутатор	Типовой ethernet коммутатор от производителя без загруженной операционной системы. Передача данных основана на контроллере ASIC.
x86 платформа	Стандартная коммерческое вычислительное оборудование (сервер) на базе процессора Intel.



## 1.5. Обратная связь

С вопросами по использованию продукта B4N Service Platform, пожеланиями или предложениями свяжитесь со специалистами компании Brain4Net:

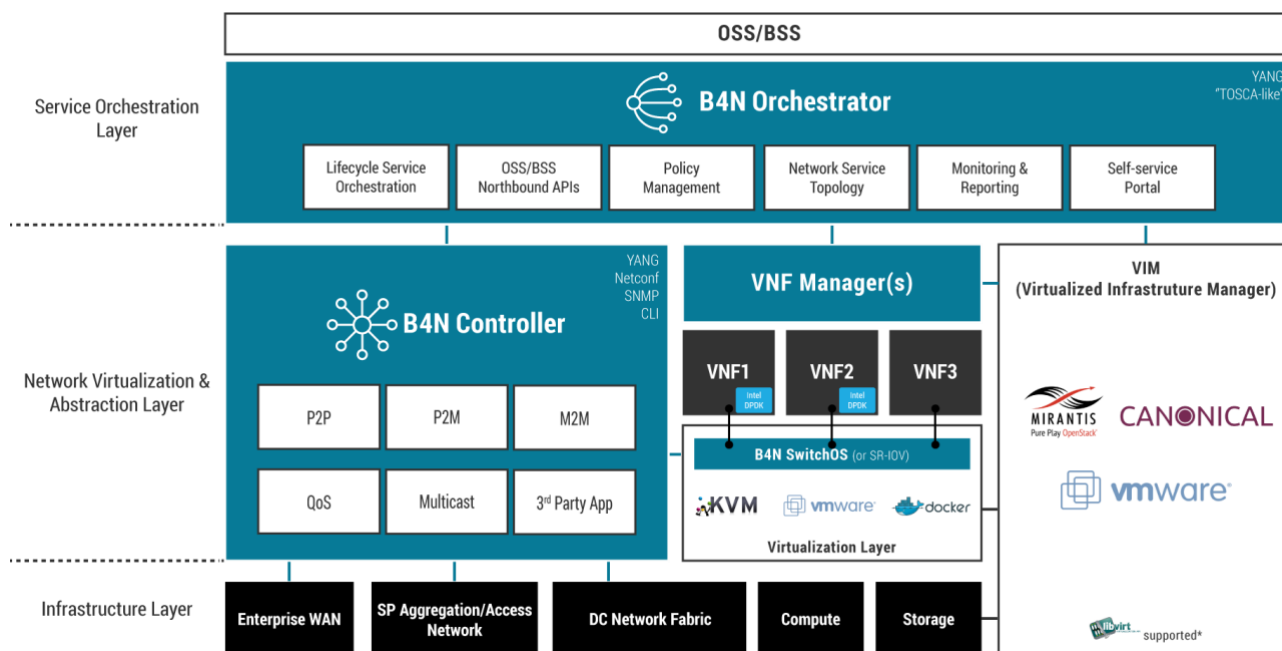
- E-mail: [need@brain4net.ru](mailto:need@brain4net.ru);
- Телефон: +7.499.649.4646.

## 2. СОСТАВ ПЛАТФОРМЫ

Платформа B4N Service Platform включает в себя:

- **B4N SwitchOS** – OpenFlow-агент для bare-metal коммутаторов или виртуальный коммутатор для работы на x86 платформе;
- **B4N Orchestrator** – единый интерфейс для управления платформой. Отвечает за выделение ресурсов, установку и активацию виртуальных сетевых функций, создание базовых сетевых сервисов, которые требуются для построения цепочки сервисов от начальной до конечной точки. Оркестратор выполняет также функции интеграции с внешними системами (OSS/BSS, OpenStack с KVM, VMWare, RHEV или Hyper-V и др.);
- **B4N Controller** – высокопроизводительный кластер SDN-контроллеров с поддержкой OpenFlow v1.3;
- **B4N VNF Manager** – модуль реализующий целостную оркестрацию сервисов или виртуальных сетевых функций. и интеграцию с VIM.

B4N Service Platform – программная платформа, базирующаяся на синергии SDN, NFV и оркестрации сервисов обеспечивая сквозную оркестрацию сервисов и управление мультивендорной сетевой инфраструктурой:



### ТРЕБОВАНИЯ ПОД СЕРВЕРЫ ДЛЯ УПРАВЛЯЮЩИХ УЗЛОВ

#### SDN transport без служебных VNF, L2-сервисы

Switches	CPU	RAM	HDD	NICs
16 switches	6	24G	150GB	2x10G with offloads
32 switches	8	32G	150GB	2x10G with offloads
64 switches	12	32G	150GB	2x10G with offloads

#### SDN transport с R-VNF, до 1 млн маршрутов на домен

Switches	CPU	RAM	HDD	NICs
16 switches	8	24G	150GB	2x10G with offloads
32 switches	10	32G	150GB	2x10G with offloads
64 switches	14	32G	150GB	2x10G with offloads

Полная спецификация и описание дизайна решения доступно в “Руководстве по дизайну” и на [web-сайте](#).

## 2.1. B4N Controller

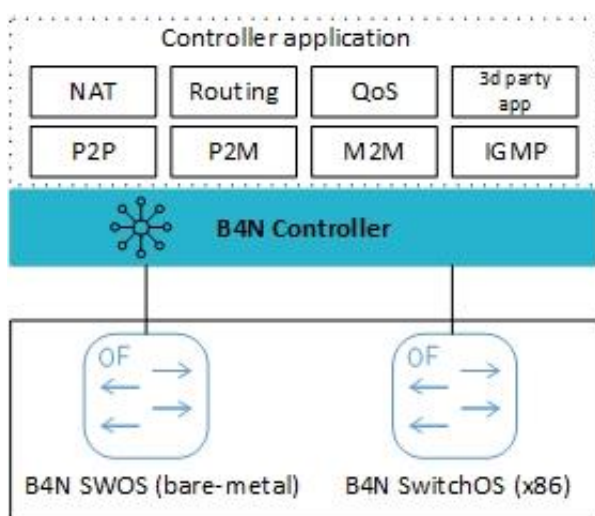


Рис 2.1 B4N Controller

B4N Controller – SDN контроллер с поддержкой OpenFlow v. 1.3.5 протокола. Продукт предназначен для создания надёжной, отказоустойчивой единой сетевой структуры, состоящей из физических и виртуальных сетевых элементов для сетей Metro Ethernet и ЦОД. B4N Controller ключевая часть портфеля B4N Service Platform, которая обеспечивает управление топологией, полный набор Carrier Ethernet EVCs (E-Line, E-LAN, E-Tree EVCs), Traffic Engineering и QoS. Продукт позволяет оператору связи обеспечить путь доставки трафика клиента от сети доступа к платформе NFV расположенной в ЦОД и тесно интегрировать сервисы сети агрегации Metro с услугами NFV или другими облачными услугами, работающими в центре обработки данных.

B4N Controller может выступать в качестве много доменного SDN контроллера для WAN сети и центра обработки данных. Как SDN контроллер центра обработки данных, B4N Controller, обеспечивает единое управление для физических bare-metal OF коммутаторов и виртуальных коммутаторов в гипервизоре, образуя единую сеть ЦОД без каких-либо наложенных технологий. Такой подход позволяет снизить нагрузку на сеть, улучшить управление сетью и упростить поиск неисправностей.

## 2.2. B4N Switch OS

B4N SwitchOS выполняет роль многоплатформенного OpenFlow коммутатора, который может быть реализован либо как виртуальный коммутатор, работающий на x86 платформе, либо как OF агент для bare-metal коммутаторов с поддержкой OF-DPA, обеспечивая единый пользовательский интерфейс для обеих версий и более гибкую программную архитектуру для сетей сервис-провайдеров. B4N SwitchOS обеспечивает высокоскоростную плоскость передачи данных для сети с поддержкой SDN, возможности быстрого развёртывания сетей и сетевых сервисов, и простоты использования и поддержки сторонних контроллеров SDN:

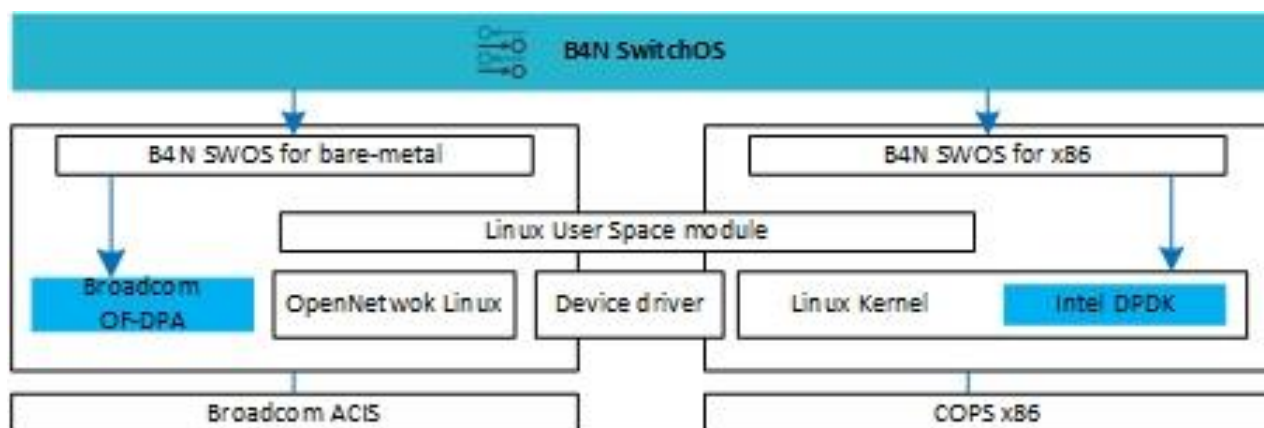


Рис 2.2 B4N Switch OS

К заказу доступны две различные версии B4N SwitchOS для различных платформ: [vSwitch для x86](#) платформы [и OpenFlow агент](#) для bare-metal коммутаторов.

### 2.2.1. B4N SwitchOS для x86 платформы

B4N SwitchOS для платформы x86 – высокопроизводительный виртуальный коммутатор, реализованный с применением Intel DPDK (Data Plane Development Kit), архитектурных решений и инновационных алгоритмов Brain4Net, что значительно повышает производительность B4N SwitchOS по сравнению с существующими на рынке виртуальными коммутаторами. B4N SwitchOS имеет ряд преимуществ:

- Предсказуемую задержку передачи трафика;
- Собственный механизм синхронизации классификаторов механизма RCU OVS;
- Отсутствие блокировки пакетов в классификаторе;
- Поддержка QoS/HQoS, Group Chaining и Metering;
- Поддержка 802.1q и 802.1ad маркировки VLAN (tagging);
- Простой перенос на новую платформу.

### 2.2.2. B4N SwitchOS для bare-metal коммутаторов

B4N SwitchOS для bare-metal коммутаторов выполнен в виде OpenFlow-агент для bare-metal коммутаторов с поддержкой ONIE, построенный на Debian/Ubuntu дистрибутивах с применением SDN SDK. Работа агента протестирована на совместимость с ограниченным списком производителей. Подробный список поддерживаемых устройств на [web сайте](#).

Ключевые функции:

- Поддержка OpenFlow v1.3.5;
- Поддержка QoS, queues & metering, select groups and group chaining;
- Поддержка расширений OpenFlow для HQoS;
- Ускорение плоскости передачи данных через использование OF-DPA (OpenFlow Data Path Acceleration);
- Поддержка ONIE (Open Network Install Environment);
- Поддержка Broadcom Trident II ASIC;

## 2.3. B4N Orchestrator

B4N Orchestrator обеспечивают сквозную оркестрацию и управление SDN доменов. Оркестратор включает в себя:

- Интерфейс управления инфраструктурой – сетевыми сервисами (через SDN-контроллеры) и вычислительными ресурсами;
- Каталог виртуальных сетевых функций (VNF);
- Конструктор сервисов;
- Web портал для Заказчиков, позволяющий управлять выделенными им ресурсами, логической топологией и сервисами;
- Северный (Northbound) интерфейс REST API для интеграции с оркестраторами других производителей и системами OSS/BSS.

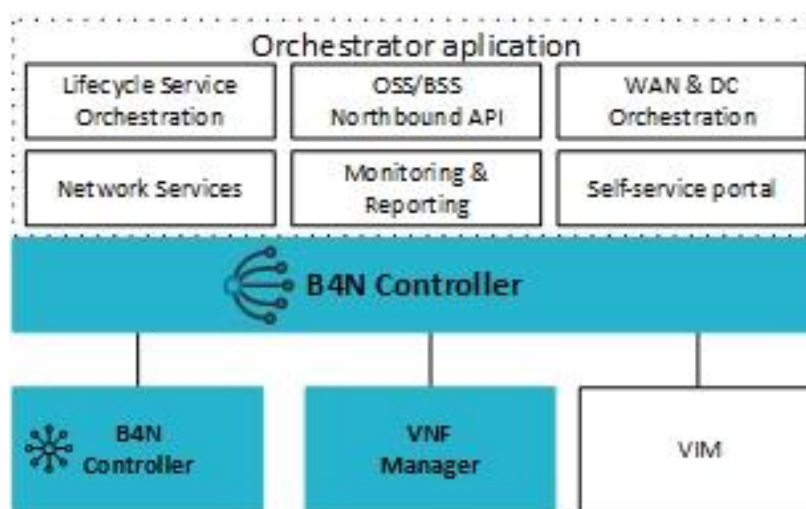


Рис 2.3 B4N Оркестратор

### 2.3.1. Оркестрация вычислительных ресурсов

B4N Оркестратор координирует виртуальные вычислительные ресурсы независимо от того, на каких физических серверах они развёрнуты, обеспечивает централизованное управление и мониторинг через интеграцию с внешней системой мониторинга. Данные мониторинга позволяют своевременно проводить автоматическое масштабирование и восстановление сетевых функций. Оркестратор обеспечивает восстановление виртуальной сетевой функции в случае сбоя.

Управление вычислительными ресурсами реализовано с помощью менеджеров виртуальной инфраструктуры (VIM) других производителей. В настоящий момент поддерживается OpenStack, который, в свою очередь, может управлять KVM, VMware, RHEV и Hyper-V.

Возможность группировки виртуальных ресурсов несколькими VIM, объединёнными в домен, помогает гранулировать ресурсы пользователей в соответствии с административным делением.

### 2.3.2. Оркестрация сетевыми ресурсами

B4N Оркестратор, с помощью SDN контроллера позволяет построить надёжную и отказоустойчивую единую сетевую фабрику, состоящую из физических элементов сети - коммутаторов с поддержкой OpenFlow и виртуальных коммутаторов. Физические коммутаторы используются в сетях Metro Ethernet на уровне агрегации и в сетевой фабрике ЦОД, виртуальные коммутаторы применяются в облачных

средах операторов (Telco Cloud). Это позволяет обеспечить связь между точками подключения клиента и создать сквозную SDN связность, распространяющуюся от сети доступа сервис провайдера к исходящим портам центра обработки данных оператора. Такой подход обеспечивает гарантированную доставку трафика клиентов от сети доступа к среде NFV, тем самым тесно интегрировать сервисы городской (Metro) сети с услугами с добавленной стоимостью – NFV, обеспечивая точный контроль трафика на всем пути его прохождения.

## 2.4. Сквозная оркестрация (VNF Manger)

VNF Manager как часть платформы B4N Service Platform полностью соответствует архитектуре ETSI NFV MANO. NFV Management & Orchestration (MANO) устанавливает стандарты управления и оркестрации ресурсов NFV в рамках облачных центров обработки данных. Эти ресурсы включают в себя вычислительные, сетевые ресурсы, системы хранения данных, виртуальные машины и др.

VNF Manager совместно с NFVO (является частью Оркестратора) - позволяют интегрировать в систему виртуальные сетевые функции, созданные другими производителями. В каждом ЦОД разворачивается отдельный экземпляр VNF Manager. С помощью оркестратора осуществляется централизованное управление распределённой архитектурой VNF Manager и администрирование виртуальных функций на протяжении их жизненного цикла. Возможность быстрого портирования виртуальных функций с применением стандартных VNF-дескрипторов позволяет сервис-провайдерам предлагать своим клиентам широкий набор виртуальных функций, коммерческих или с открытым кодом.

Основные функции NFVO и VNF Manager:

- Управление жизненным циклом VNF (создание, обновление, удаление),
- Выбор места размещения высокопроизводительных и высоконагруженных VNF,
- Начальная конфигурация VNF,
- Мониторинг VNF,
- Обеспечение доступности - автоматический перезапуск VNF в случае необходимости на другом хосте или в другом ЦОД,
- Обеспечение масштабируемости - добавление ресурсов в случае их нехватки или освобождение ресурсов, если нагрузка VNF низкая.

Возможности по управлению жизненным циклом VNF позволяют свести к минимуму перерывы в работе сервисов, связанные с запланированными процедурами обслуживания систем. Например, при необходимости обновления ПО новая версия вначале портируется в систему и тестируется, затем создаётся новый экземпляр VNF с новой версией ПО, трафик переключается на новую VNF, а старый экземпляр удаляется.



### 3. РАБОТА С СИСТЕМОЙ

Для начала работы с платформой необходимо удостовериться в наличии и доступности оркестратора. B4N Orchestrator является единой точкой управления B4N Service Platform, жизненным циклом сетевых сервисов и сквозным управлением виртуальных сетевых функций. Компонент предоставляет администратору универсальный графический Web интерфейс. Для соединения с оркестратором необходимо ввести в строке web браузера адрес оркестратора вида <http://192.168.1.11/> с указанием истинного IP адреса вашей платформы, который определяется на моменте дизайна и инсталляции платформы.

При работе с интерфейсом необходимо изучить следующие термины и основные определения:

- **Домен** – логическая группа ресурсов под единым административным управлением.
- **Дата-центр** – логическая сущность, представляющая агрегацию сетевых и вычислительных ресурсов (центры обработки данных, SDN-контроллеры, менеджеры управления инфраструктурой виртуальных сетевых функций (VNF));
- **Оркестратор** – центральный управляющий компонент, осуществляющий оркестрацию жизненного цикла услуг, предоставляемых через B4N платформу;
- **Openflow контроллер** – центральный управляющий компонент, управляющий набором связанных между собой коммутаторов (сетевым доменом) по протоколу Openflow;
- **Кластер контроллеров** – кластер из нескольких контроллеров;
- **Инфраструктура** – иерархическая структура из сетевых, вычислительных и системных ресурсов необходимых для предоставления сервисов;
- **Каталог** – список виртуальных сервисов и машин доступных через платформу;
- **Арендаторы (Тенанты)** – логическая сущность, объединяющая доступные для управления зарегистрированным пользователем ресурсы (часть SDN сети и набор виртуальных сетевых функций оператора связи);
- **Коммутатор OpenFlow** – сетевой узел коммутации Ethernet фреймов поддерживающий протокол OpenFlow;
- **Порт** – физический порт коммутатора;
- **Интерфейс** – логический порт коммутатора, подключенный к другому коммутатору или к внешнему устройству. Контроллер при построении топологии оперирует интерфейсами;
- **Логический Openflow порт** – порт коммутатора, который организуется Openflow-агентом. Openflow-контроллер оперирует OF-портами коммутатора, для него Interface = OF-port;
- **Внутренний интерфейс (NNI)** – интерфейс, подключённый к другому коммутатору внутри сетевого домена, контролируемого одним контроллером (кластером контроллеров);
- **Внешний интерфейс (UNI)** – интерфейс, подключённый к оборудованию вне сетевого домена. Интерфейс подключения пользователя;
- **Внешний интерфейс (eNNI)** – внешний интерфейс, подключённый к внешней сети. Обычно используется совместно с динамическим протоколом маршрутизации с внешней сетью;
- **Точка подключения (Connection Point)** – точка подключения пользователя к сетевому домену на каком-либо внешнем интерфейсе без привязки к какому-либо сервису. Имеет разные типы инкапсуляции (Access, VLAN, QinQ и т.д.);
- **Сервисный интерфейс (SI)** – сервисный интерфейс пользователя, подключённый к конкретному сервису на каком-либо внешнем интерфейсе;
- **Статический сервисный интерфейс (Static SI)** – SI, инкапсуляция которого определена вручную;
- **Динамический сервисный интерфейс (Dynamic SI)** – SI, инкапсуляция которого определяется контроллером в ходе работы;
- **L2- сервисный интерфейс (L2 SI)** – SI, включённый в состав L2-сервисов (P2P, P2M, M2M);
- **L3- Сервисный интерфейс (L3 SI)** – SI, включённый в состав L3-сервисов (L3 Service);
- **P2P сервис (E-line)** – сервис точка -точка (Point-to-point), соединяющий на уровне L2 два SI;



- **M2M сервис (E-LAN)** – сервис мультиточка-мультиточка (Multipoint-to-multipoint), логический мост, соединяющий на уровне L2 два или более SI;
- **P2M сервис (E-Tree)** – сервис точка-мультиточка (Point-to-multipoint), логический мост, соединяющий на уровне L2 один или более SI с ролью Root с одним или несколькими SI с ролью Leaf SI с ролями Leaf не могут обмениваться фреймами между собой;
- **IP Multicast сервис** – сервис состоит из одного или более SI, к которым подключены источники IP multicast, вещающие в режиме статической подписки и SI которым подключены подписчики;
- **L3 Service** – сервис контроллера, представляющий собой логический маршрутизатор между двумя или более L3-SI;
- **VNF** – виртуальная сетевая функция;
- **VNFM** – менеджер виртуальных сетевых функций;
- **VDU** – виртуальная машина/ы в составе VNF;
- **VIM (Virtual Infrastructure Manager)** – система управления виртуализацией (Openstack, vCenter и т.д.);
- **NS** – сетевой сервис;
- **NST** – шаблон сетевого сервиса;
- **VNFD** – дескриптор, описывающий VNF;
- **VNF Package** – пакет, содержащий образы VDU, VNFD, конфигурационные файлы.

### 3.1. Установка платформы

Установка платформы производится в среду виртуализации контейнеров Docker, работающую поверх операционной системы Linux. Работоспособность гарантируется при использовании только 64 bit операционных систем:

- Debian не ниже 8.1;
- Ubuntu не ниже 14.04 LTS;
- CentOS не ниже 7.1;
- RedHat не ниже 7.1;
- SUSE не ниже 12.4.

Версия платформы контейнерной виртуализации Docker должна быть не ниже 1.5.

Минимальные требования для установки сервисной платформы B4N Service Platform:

Тип установки\Количество	Автономная	Отказоустойчивая (HA)
Серверов\BM	1	3
Ядер на платформу	6	18
ОЗУ (RAM) на платформу, Gb	10	30
Дисковое пространство (HDD) на платформу, Gb	96	288

Процедура установки компонентов платформы подробно описана в руководстве "B4N.Controller.InstallationGuide".

## 3.2. Референсная архитектура платформы

Типовой вариант установки решения в отказоустойчивом режиме. В качестве платформы для установки программного обеспечения возможно использование аппаратных серверов или виртуальных машин с соблюдением требований для установки.

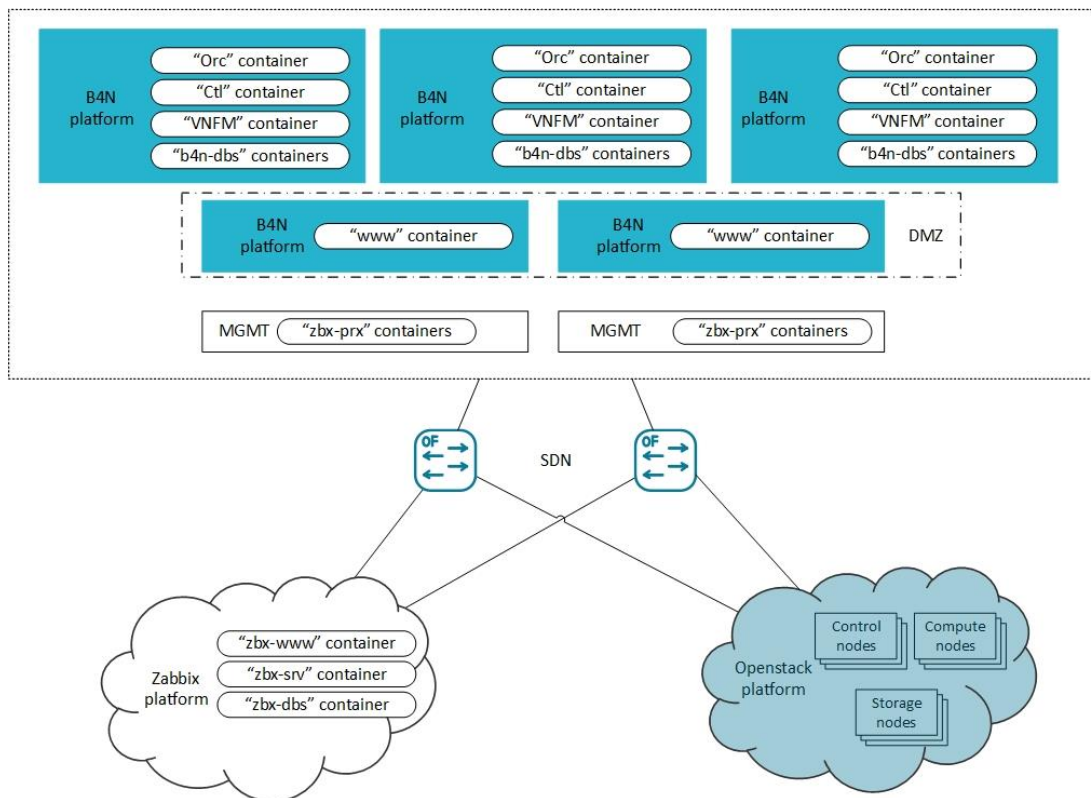


Рис 2.4 B4N Архитектура платформы

Рекомендуемый вариант установки содержит следующий набор компонентов:

- 3 сервера с компонентами платформы;
- 2 сервера с модулем Web интерфейса;
- Платформа внешнего мониторинга Zabbix;
- Платформа OpenStack.

## 3.3. Архитектура системы

В архитектуре платформы заложена отказоустойчивость ключевых элементов, таких как оркестратор, контроллер, база данных по схеме 2N+1. При начальном дизайне системы рекомендуется использовать рекомендации, изложенные в "Руководстве по планированию и внедрению платформы B4N". Для случая критически важных внедрений, для обеспечения масштабируемости и увеличения отказоустойчивости рекомендуется применить схему георезервирования с разнесением ресурсов по разным доменам и дата-центрам.

### 3.4. Настройка и мониторинг отказоустойчивости

Для выполнения минимальных требований отказоустойчивости, мы рекомендуем устанавливать оркестратор и его базу данных минимум на 3 различных хоста. Архитектура платформы обеспечит создание кластера оркестратора, балансировку нагрузки и отказоустойчивость элемента. Используемая схема резервирования контроллеров настраивается при настройке платформы на этапе создания инфраструктуры.

На этапе создания/модификации сетевых ресурсов задаётся количество хостов, формирующих кластер контроллеров. Возможны варианты один, три, пять. Экран настройки сетевой инфраструктуры позволяет отслеживать статус контроллеров и просматривать статистику их использования. Экран Мониторинг, вызываемый из меню Администрирование главного экрана платформы позволяет отслеживать состояние кластеров и отслеживать статистику работы платформы. Настройка файлов журналирования (лог файлов) осуществляется в соответствии с разделом [Создание резервной копии и восстановление \(Backup & Restoring\)](#), позволяя проконтролировать работу кластеров в течении необходимого промежутка времени.

## 4. БАЗОВЫЕ НАСТРОЙКИ ПЛАТФОРМЫ

Управление платформой производится из приложения Оркестратор. Интерфейс приложения основан на применении стандартных (используемых в большинстве приложений) схем для подобного рода приложений. Платформа использует интерфейс единого окна для управления всеми компонентами платформы. Интерактивные элементы интерфейса и строка меню позволяют переключиться между разными уровнями. Дополнительно, ряд элементов интерфейса имеют контекстное меню, активируемое при обращении к ним.

Для начала работы с платформой необходимо пройти процедуру аутентификации. Необходимо ввести Имя пользователя и Пароль, определённые на этапе дизайна и установки платформы. Платформа имеет аккаунт по умолчанию (Имя пользователя и Пароль = admin) который используется на этапе установки системы и должен быть обязательно заменён для безопасности системы. На стартовой странице размещено меню содержащее следующие опции:

- Учётная запись;
- Язык интерфейса;

При вводе корректных параметров учётной записи пользователь попадает на страницу по умолчанию. По умолчанию, данной страницей является страница с базовыми настройками платформы.

Стартовая страница интерфейса с базовыми настройками платформы содержит меню настройки основных параметров системы. Меню имеет следующие опции:

- Учётная запись;
- Администрирование;
- Язык интерфейса;
- Feedback;
- Настройка страницы по умолчанию.

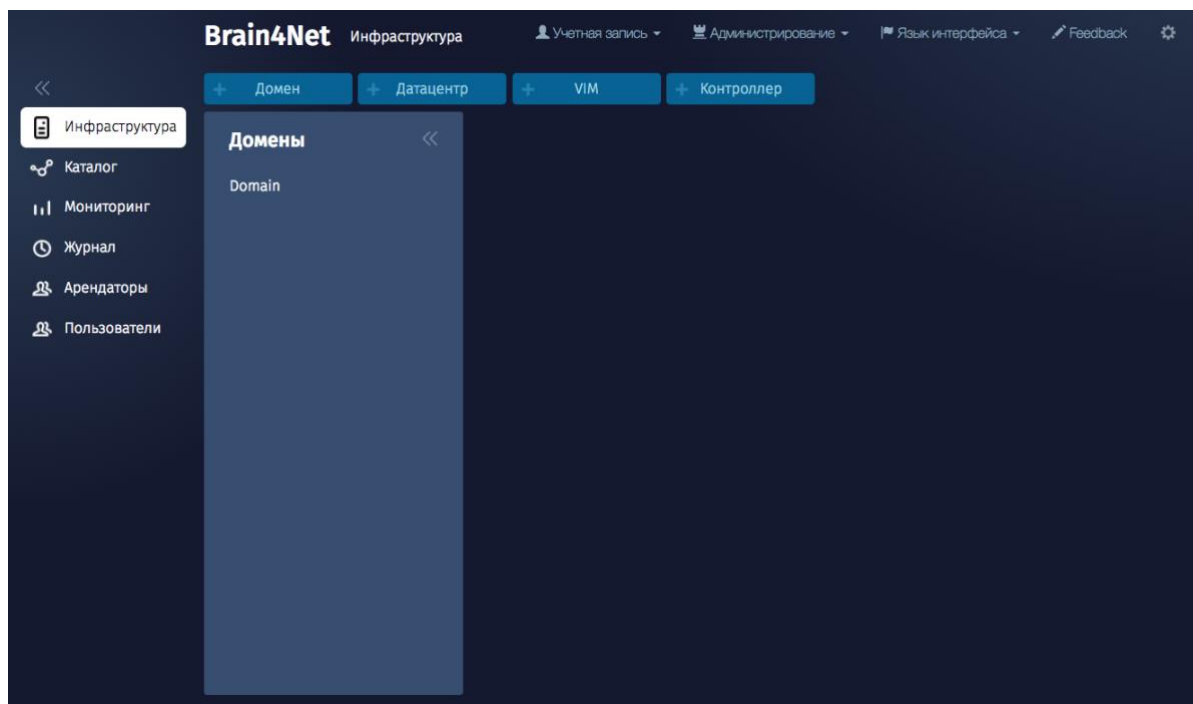


Рис. 4.1 Экран базовых настроек

## 4.1. Учетная запись

Пункт меню предназначен для настройки параметров учётных записей пользователей системы и содержит следующие пункты:

- Настройки – позволяет сменить параметры текущей учётной записи;
- Пароль – позволяет поменять пароль для текущего пользователя;
- Сессии – список активных сессий;
- Выйти – выход из системы.

Меню «Настройки» позволяет сменить следующие параметры пользователя:

- Имя – имя пользователя;
- Фамилия – фамилия пользователя;
- E-mail – электронный почтовый ящик пользователя;

Список активных сессий отображает:

- IP адрес хоста с которого осуществлено подключение;
- Название и тип web браузера, название и тип операционной системы;
- Дату создания сессии;
- Программную кнопку для принудительного завершения.

Смену настроек пользователя может произвести только пользователь с правами администратора. Изменение и просмотр остальных параметров ограничивается производится только для учётной записи под которой осуществлён вход в платформу.

## 4.2. Администрирование

Пункт меню предназначен для настройки параметров платформы. Требуется у текущего пользователя прав администратора и содержит следующие пункты:

- Метрики;
- Состояние;
- Конфигурация;
- Управление аккаунтами;
- Аудит;
- Логи;
- API.

Администратор платформы может использовать отображаемые параметры из пунктов меню для контроля производительности платформы, тонкой настройки платформы и поиска неисправностей и отправки в службу технической поддержки Brain4Net по адресу <mailto:support@brain4net.com>.

### 4.2.1. Метрики

Пункт Метрики в меню Администрирование отображают состояние производительности платформы за текущий день.

На экране метрик можно проверить показатели производительности платформы в виде:

- Используемого объёма памяти в процентах;
- Текущие показатели работы платформы;
- Количество HTTP запросов в секунду;
- Статистика работы сервисов.

Доступен просмотр метрик следующих компонентов инфраструктуры:

- Контроллеров – элемент Статистика элемента Узлы экрана Сетевые элементы;
- Коммутаторов – элементы Flow, Статистика по интерфейсам, меню Управление;
- Сервисов (P2M, P2P, M2M и т.д.) – элемент Статистика меню Управление каждого из сервисов.

#### **4.2.2. Состояние (Health)**

Пункт Состояние в меню Администрирование отображают текущий статус сервисов платформы.

На экране информация отображается в виде:

- Названия сервиса, назначенного платформой;
- Статус сервиса:
  - Ошибка;
  - ОК.

Подробное описание проблемы и сервиса (Stacktrace). StackTrace доступен в случае вызова всплывающего меню.

#### **4.2.3. Конфигурация**

Пункт Конфигурация в меню Администрирование отображают текущие настройки конфигурации платформы.

На экране информация отображается в виде:

- Префикс – название параметра;
- Параметры, определённые на платформе (Свойства);
- Значения параметров.

В верхней части экрана присутствует строка для фильтрации вывода.

#### **4.2.4. Управление аккаунтами**

Доступ к пункту меню имеет учётная запись с правами администратора платформы. Отображаемый список пользователей, зарегистрированных на платформе содержит следующие поля:

- Имя – наименование пользователя;
- Арендатор – наименования арендатора которому принадлежит пользователь;
- Роль – определяет права пользователя в системе;
- Группы – название группы к которой принадлежит пользователь;
- Наличие пользователя в сети.

Для каждой учётной записи (аккаунта) необходимо заполнить следующие поля:

- Логин – наименование пользователя;
- Пароль – пароль аккаунта;
- Роль – определяет права пользователя в платформе;
- Имя – имя пользователя;
- Фамилия – фамилия пользователя;
- E-mail – электронный почтовый ящик пользователя;
- Описание – описание аккаунта.

Администратор платформы имеет следующие права на работу с учётными записями:

- Блокировка учётной записи;
- Разблокировка учётной записи;
- Смена пароля учётной записи;
- Смена персональных данных учётной записи;
- Удаление учётной записи.

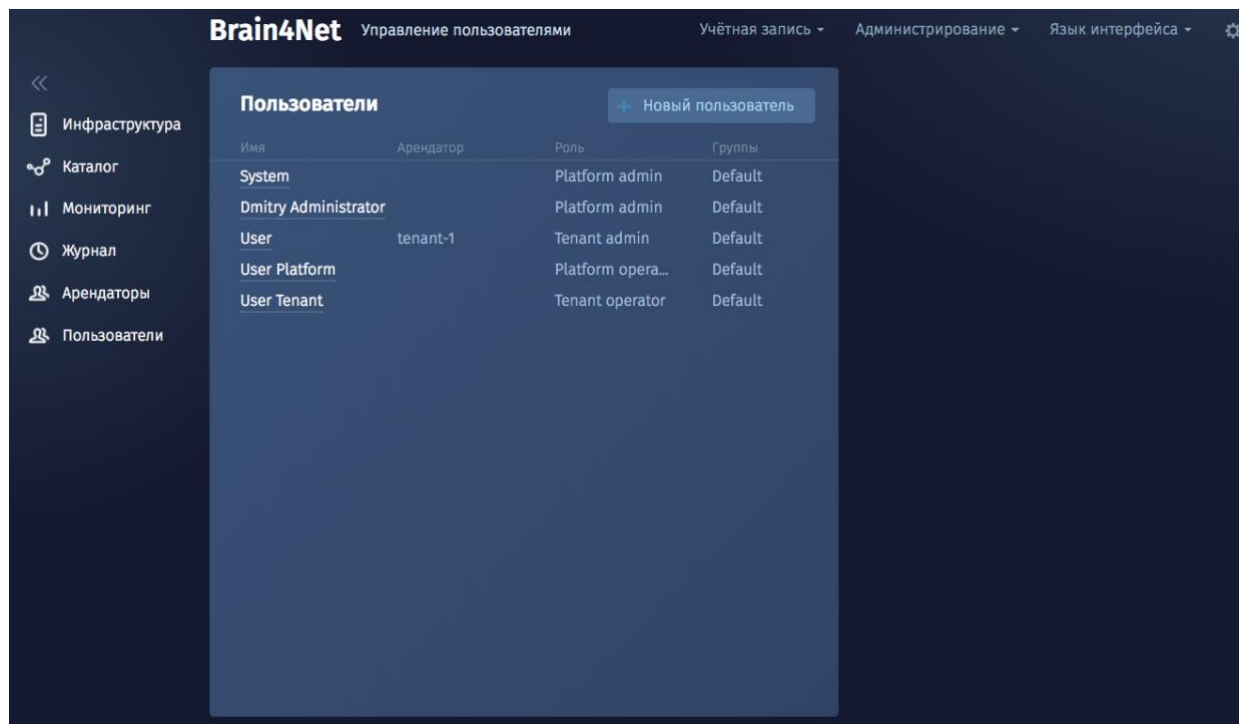


Рис 4.2 Управление аккаунтами

Доступ к управлению аккаунтами также доступен через элемент «Пользователи» основного меню.

#### 4.2.5. Описание ролей

Пользователи платформы (учётные записи) делятся на две группы: сотрудники компании – поставщика услуг (администратор платформы, оператор платформы) и сотрудники компании-арендатора (администратор тенанта, оператор тенанта), таким образом, что для каждой категории определяются свои роли и функции доступа.

В системе, на настоящий момент, определены следующие роли аккаунта с правами:

**Администратор платформы (Platform admin)** – сотрудник, должностные обязанности которого подразумевают обеспечение штатной работы платформы. В обязанности администраторов входит обеспечение оптимальной работоспособности сетевых и вычислительных ресурсов, конфигурация (создание шаблонов) сетевых функций и сервисов. Администратор платформы имеет доступ ко всей функциональности;

**Оператор платформы (Platform operator)** – сотрудник, в обязанности которого входит мониторинг и работа с арендаторами (создание, удаление, выделение ресурсов и прочие взаимодействия с заказчиками);

**Администратор арендатора (Tenant admin)** – администратор компании-клиента или провайдера, работающий с выделенными ему ресурсами. Его возможности ограничены доступными ему сетевыми и вычислительными ресурсами (его инфраструктурой), а также возможностями конфигурации сетевых сервисов. Администратор арендатора конфигурирует сервисы по созданным администратором платформы или собственным шаблонам, но при этом низкоуровневые сервисы ему не доступны.



Администратор тенанта не работает с физическими ресурсами (дата-центром) провайдера напрямую, он видит только точку входа и собственные ресурсы. Типичные задачи: создание, редактирование, мониторинг виртуальной сетевой инфраструктуры;

**Оператор арендатора (Tenant operator)** – сотрудник, который может выполнять операции для поддержания работоспособности инфраструктуры в рамках доступной ему области (мониторинг, статусы систем). Также он может подключаться к виртуальным машинам консолью для анализа неисправностей.

#### 4.2.6. Аудит

Пункт Аудит в меню Администрирование отображает действия пользователя, зарегистрированного на платформе, его удачные / неудачные попытки подключения, дата, время и IP адрес.

На экране информация отображается в виде:

- Дата события;
- Имя пользователя;
- Текущий статус пользователя;

Дополнительные параметры – отображает IP адрес станции активного пользователя или статус последней сессии пользователя.

#### 4.2.7. API

Пункт API в меню Администрирование содержит страницу с набором готовых процедур и функции, предоставляемых B4N Orchestrator для интеграции с внешними платформами. Список функций разбит на модули по функциональному назначению. Для каждого элемента списка доступна функция отображения/скрытия объектов, а также просмотр списка операции по выбранному объекту.

### 4.3. Язык интерфейса

Платформа в настоящий момент поддерживает 2 языка интерфейса: Русский и Английский. Пользователь может выбрать необходимый язык в меню стартовой страницы.

### 4.4. Feedback

Пункт меню Feedback позволяет составить комментарии, сообщения и отправить сообщение на электронный адрес пользователя с ролью «Администратор платформы».

### 4.5. Страница по умолчанию

На первоначальном этапе, для целей настройки и конфигурирования платформы стартовый экран платформы показывает меню базовой настройки платформы и список компонентов, составляющие решение. Компоненты имеют следующее разделение по типам:

- Инфраструктура:
  - Домен;
  - Дата-центр;
  - VIM;
  - Контроллер;
- Каталоги:

- Шаблоны (Templates);
  - VNF;
  - Виртуальные машины.
- Мониторинг;
- Журнал:
  - Дата-центры;
  - Ресурсы;
  - Журналы.
- Арендаторы:
  - Арендаторы;
  - VIM;
  - UNI;
  - Пользователи;
  - Каталоги;
  - Service requests.
- Пользователи платформы.

Администратор платформы может назначить любую произвольную страницу интерфейса как страницу по умолчанию.

## 5. НАСТРОЙКА ИНФРАСТРУКТУРЫ

Инфраструктура – иерархическая структура из сетевых, вычислительных и системных ресурсов необходимых для предоставления сервисов.

### 5.1. Создание доменов

Домен – логическая группа ресурсов под единым административным управлением.

Для начала работы администратору необходимо создать новый домен или выбрать один из существующих, в котором затем создаётся дата-центр. Создание домена возможно через элемент Домен верхнего меню экрана. При создании домена указывается:

- Название;
- Комментарий (описание, например, название региона).

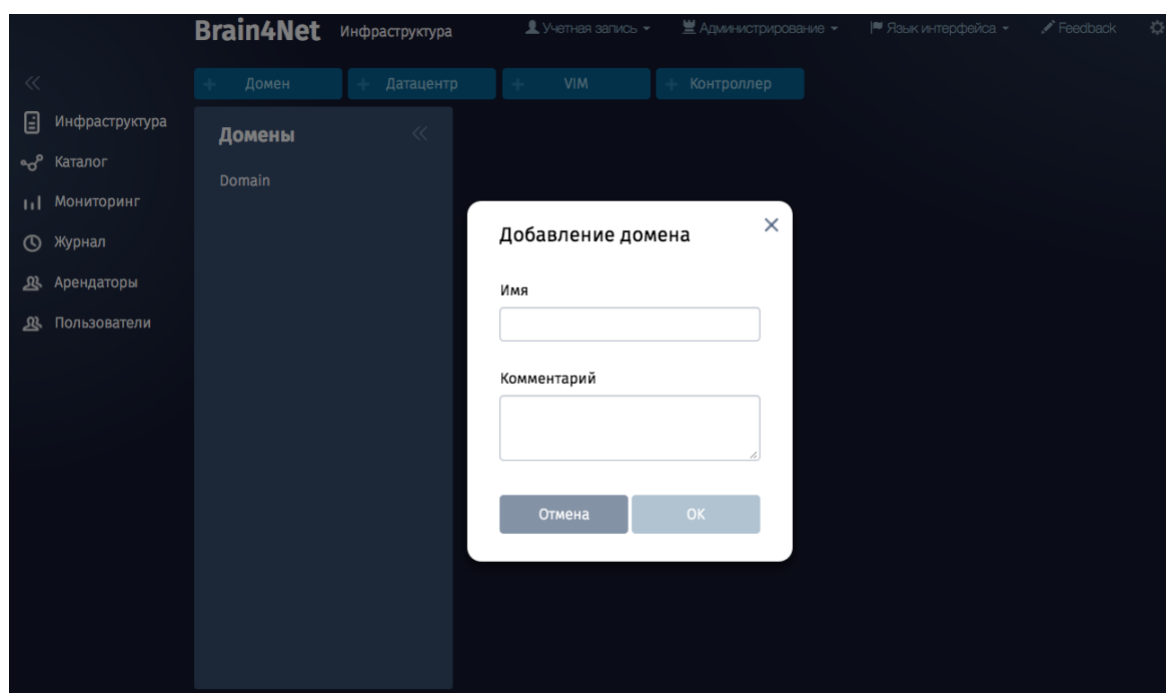


Рис 5.1 Создание доменов

### 5.2. Создание дата-центров

Дата-центр – второй необходимый элемент - логическая сущность, представляющая агрегацию сетевых и вычислительных ресурсов.

Все дата-центры обязательно должны принадлежать какому-либо домену. Возможна миграция дата-центра из одного домена в другой. Администратору необходимо создать новый домен или выбрать один из существующих, в котором затем создаётся дата-центр. При создании Дата-центра указывается:

- Название;
- Описание (например, название региона, подразделения и т.д.);
- Домен, к которому принадлежит дата-центр;

- VNFM URL – ссылку на VNF Manager – систему управления виртуальными сетевыми функциями;
- Местоположение. Адрес может быть выбран на карте или введён вручную; если адреса не существует, необходимо указать точку на карте и ввести его вручную.

При создании дата-центра администратор может сразу создать или добавить следующие ресурсы:

- вычислительные ресурсы;
- сетевые ресурсы;
- системные ресурсы - отдельные серверы под управлением платформы.

Ресурсы также могут быть добавлены позднее в рамках настройки дата-центра. Создание сетевых ресурсов подробно описано в разделе [Сетевые ресурсы](#). Создание и добавление вычислительных ресурсов подробно описано в разделе [Создание VIM](#).

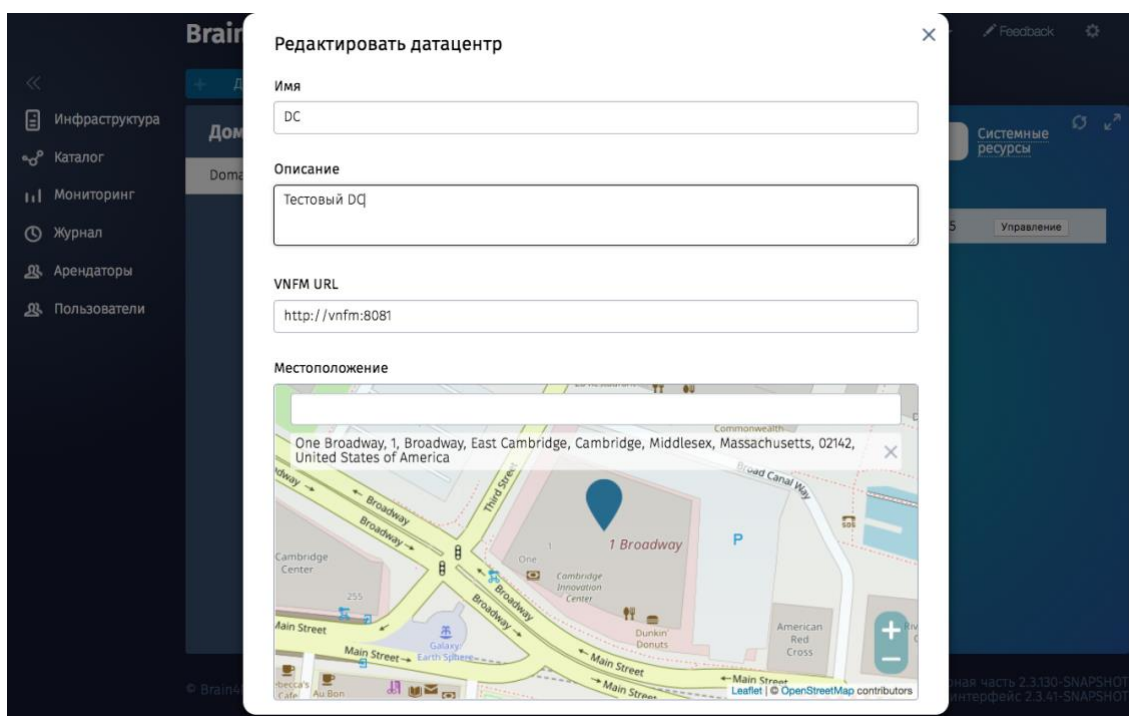


Рис 5.2 Создание дата центров

### 5.3. Вычислительные ресурсы

Вычислительные ресурсы – это предварительно настроенные физические серверы, используемые для обеспечения работы виртуальных сервисов и виртуальных машин.

Вычислительные ресурсы могут быть добавлены в систему через VIMs (Virtualized Infrastructure Manager).

VIM (Virtualized Infrastructure Manager) – менеджер инфраструктуры и ПО стороннего производства, отвечающее за конфигурацию инфраструктуры вычислительных ресурсов.

В качестве VIM в текущей версии поддерживается система виртуализации OpenStack.

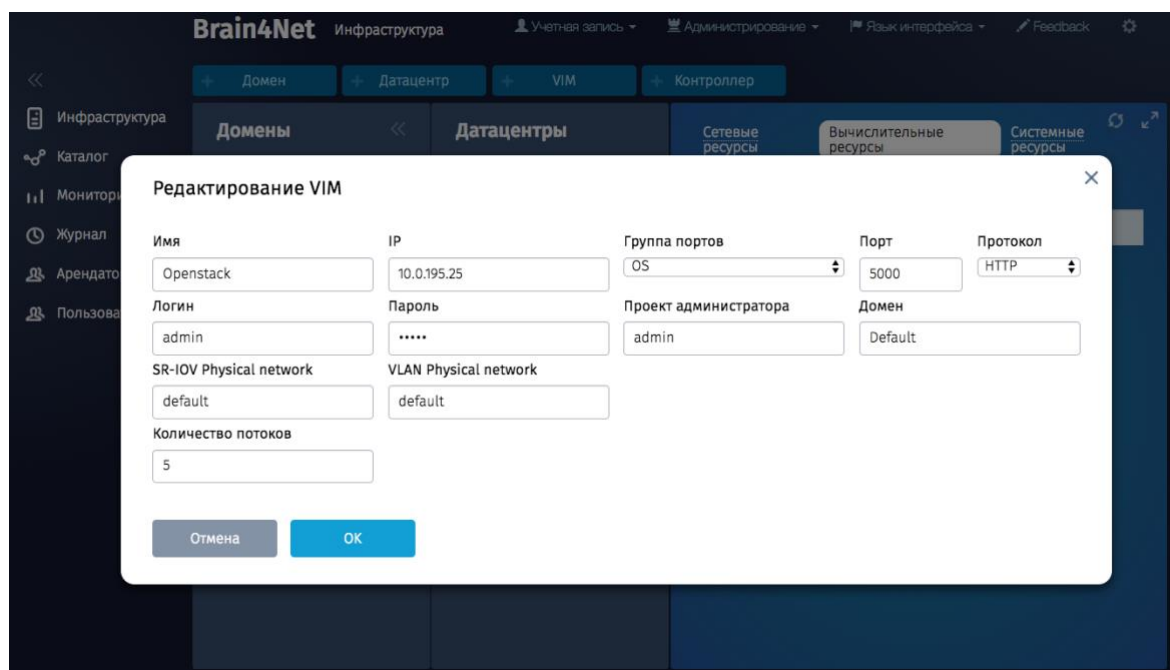
### 5.3.1. Создание VIM

Доступ к меню создания доступен с закладки Инфраструктура / Дата центр. При конфигурировании указываются следующие параметры:

- Имя VIM внутри платформы;
- IP адрес OpenStack контроллера (controller node);
- Группа портов OpenStack коммутатора, через которую подключён OpenStack контроллер. Настройка портов описана в разделе Сетевые ресурсы / Настройка группы портов;
- Порт;
- Протокол: HTTP/HTTPS;
- Учётные данные согласно настройкам, OpenStack:
  - логин;
  - пароль;
- Проект администратора согласно настройкам, OpenStack;
- Домен согласно настройкам, OpenStack;
- Адрес SR-IOV physical network согласно настройкам сети, OpenStack;
- Адрес VLAN physical network согласно настройкам сети, OpenStack;
- Количество потоков - параметр определяющий количество параллельных запросов от Оркестратора в сторону VIM. Количество запросов зависит от производительности OpenStack и влияет на скорость работы решения. Рекомендованное начальное значение – 5.

Для созданного VIM, пользователю с правами Администратора возможны следующие действия:

- Редактирование параметров;
- Удаление.



– Рис. 5.3 Создание VIM

Перед подключением VIM к платформе, в Сетевых ресурсах должны быть созданы следующие сущности и параметры:

- Подключен Контроллер;

- Активный коммутатор, через который подключён VIM;
- Созданы правила QoS на OpenFlow сети;
- Созданы сервисные интерфейсы (SI) для вычислительной инфраструктуры.
- Созданы группы портов для вычислительной инфраструктуры.

Настройка подробно описана в соответствующих разделах Сетевых настроек.

## 5.4. Системные ресурсы

В разделе «Системные ресурсы» настраиваются параметры сопряжения с системой внешнего мониторинга и параметры менеджера виртуальной инфраструктуры – VNFМ.

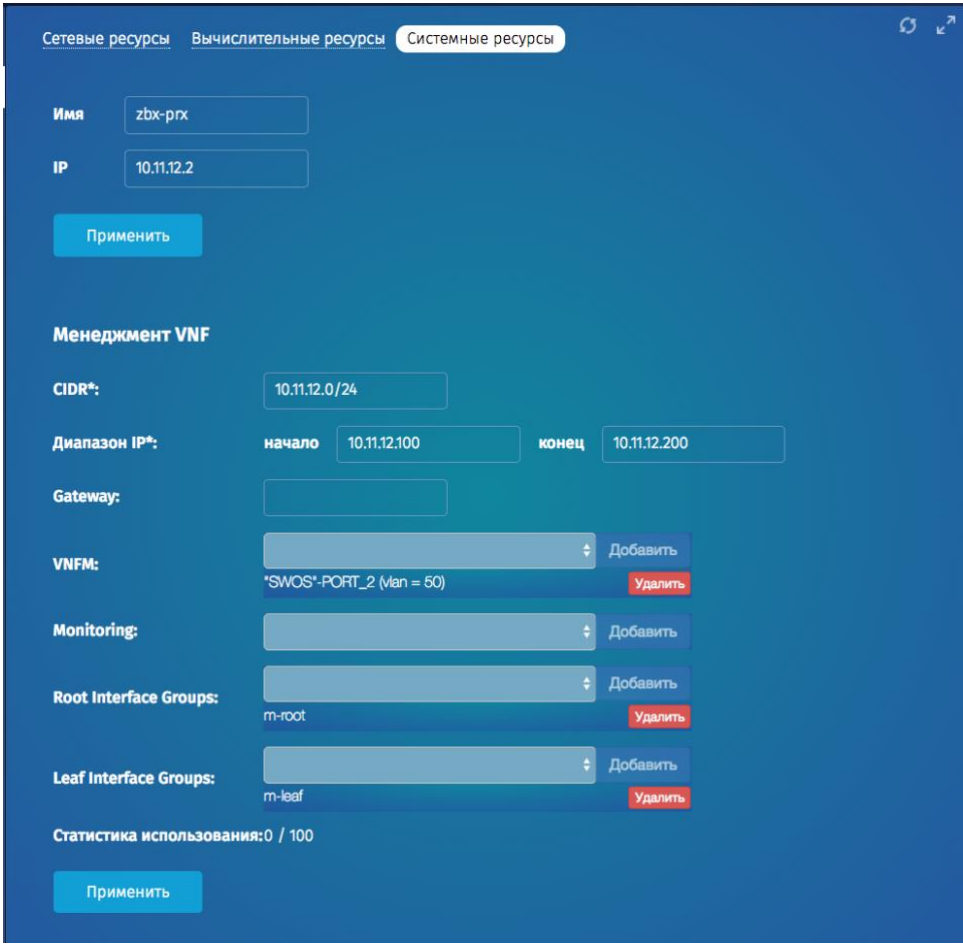


Рис. 5.4 Системные ресурсы

### 5.4.1. Настройка интерфейса к системе мониторинга

Для масштабирования, в архитектуре решения предусмотрено размещение проху сервера рекомендуемой внешней системы мониторинга Zabbix в каждом ЦОД.

При настройке необходимо указать:

- Имя проху сервера;
- Его IP адрес.

Подробная информация по организации мониторинга виртуальных функций см. Раздел «Мониторинг VNF» документа «Руководство по дизайну»

#### **5.4.2. Настройка интерфейсов к VNFM**

Для настройки работы менеджера виртуальных функций (VNF) – VNFM необходимо предварительно создать интерфейсы и группы портов к которым подключены элементы вычислительной инфраструктуры. Настройка подробно описана в разделе «Сетевые сервисы» текущего документа.

При настройке необходимо определить следующие параметры:

- IP сеть (CIDR) – сеть управления VNF;
- Диапазон IP адресов – IP адреса, назначаемые VNF;
- Gateway – IP адрес шлюза по умолчанию;
- VNFM – преднастроенный сервисный интерфейс OF коммутатора к которому подключен VNFM;
- Monitoring - преднастроенный сервисный интерфейс OF коммутатора к которому подключен сервер Zabbix проху;
- Root interface groups - преднастроенная группа портов OF коммутатора в которую входят порты с подключенными VNFM, система мониторинга (Zabbix проху), OpenStack Controller Node и VNFM Orchestrator (B4N Orchestrator);
- Leaf interface groups - преднастроенная группа портов OF коммутатора в которую входят порты с подключёнными ресурсами VIM, на которых разворачиваются VNF (OpenStack Compute Node).

После выбора каждого параметра необходимо добавить его в платформу. По завершению настройки необходимо применить параметры.

В случае успешного применения надпись: «Настройки сохранены» появляется на экране.



## 6. СЕТЕВЫЕ РЕСУРСЫ

Сетевые ресурсы – сетевая инфраструктура OpenFlow (OF) устройств под управлением отдельного контроллера или пула контроллеров.

### 6.1. Регистрация контроллеров

Создание топологии SDN под управлением платформы начинается с регистрации контроллеров.

Контроллеры – это компоненты платформы, отвечает за управление «жизненным циклом» сетевых услуг. Функция оркестрации имеет также подфункции, относящиеся к различным аспектам оркестрации SDN сети.

На платформе, в зависимости от схемы отказоустойчивости, создаваемой топологии возможно зарегистрировать один, три или пять контроллеров. Для создания отказоустойчивого решения мы рекомендуем использовать не менее 3 или 5 контроллеров разворачиваемых на базе серверов. Регистрация (подключение) контроллеров возможно через меню экрана настройки инфраструктуры (элемент Контроллер).

Для добавления контроллера необходимо указать:

- Имя – наименование контроллера / группы контроллеров в платформе;
- Описание (назначение контроллера, его местоположение);
- Количество контроллеров;
- Транспортная / сервисная стратегия – VLAN(SR) / MPLS – OFDPA (SWAP)
- Тип подключения - multicast / unicast;
- Данные контроллеров (для каждого контроллера):
  - Адрес (IP или имя хоста);
  - GRPC порт;
  - JGroups порт;
- Указать основной (Primary) контроллер.

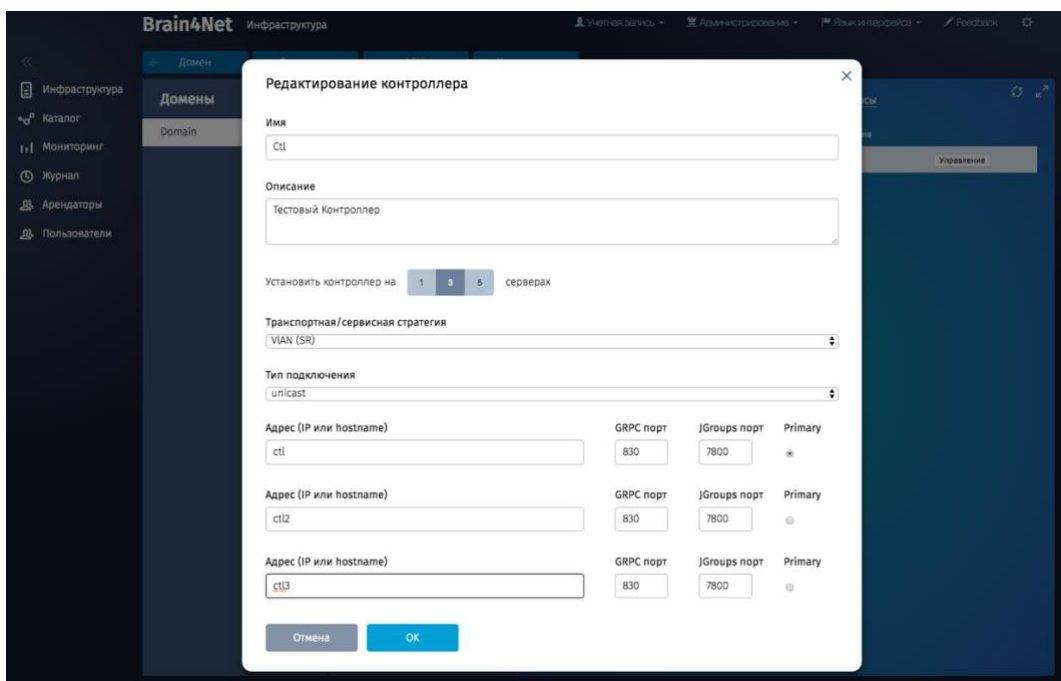


Рис 6.1 Регистрация контроллеров

Транспортная стратегия зависит от используемых OpenFlow коммутаторов на сети SDN. Платформа поддерживает работу с программными OpenFlow коммутаторами на базе x86 платформы и bare-metal коммутаторами на базе ASIC Broadcom Trident II, включая собственный продукт B4N SwitchOS.

- В случае использования программных коммутаторов на базе x86 платформы необходимо выбрать стратегию VLAN (SR);
- В случае использования bare-metal коммутаторов или присутствия обоих типов коммутаторов на сети SDN необходимо выбрать стратегию MPLS – OFDPA (SWAP).

Тип подключения определяет тип маршрутизации на сети управления кластера контроллеров. По умолчанию используется тип «unicast». В случае типа «multicast» сеть должна поддерживать передачу multicast трафика. В качестве Multicast IP адреса рекомендуется выбрать произвольное значение из пула 239.0.0.0 - 239.255.255.255, за исключением диапазона 239.0.0.x - 239.128.0.x.

После создания (регистрации) контроллеров, на экране появляется поле со списком сетевых ресурсов, на котором отображаются контроллеры, зарегистрированные в системе. Слева от списка присутствует программная кнопка «Управление» с меню состоящего из следующих элементов:

- Изменить – изменение параметров контроллеров;
- Конфигурировать – настройка сетевой инфраструктуры и сетевых сервисов;
- Резервировать – создание резервной копии конфигурации контроллеров;
- Восстановить – восстановление конфигурации из резервной копии;
- Удалить – удалить зарегистрированные на платформе контроллеры.

## 6.2. Создание сервисов

Вторым шагом при создании сетевых ресурсов будет создание логической (сервисной) топологии на базе физической топологии, созданной из OpenFlow (OF) коммутаторов, имеющих соединение контроллером. Предварительно, на этапе инсталляции сети на OF коммутаторе должен быть настроен доступ к контроллеру. Подробности описаны в "B4N.Controller.InstallationGuide" и руководстве по настройке оборудования, предоставляемой производителем.

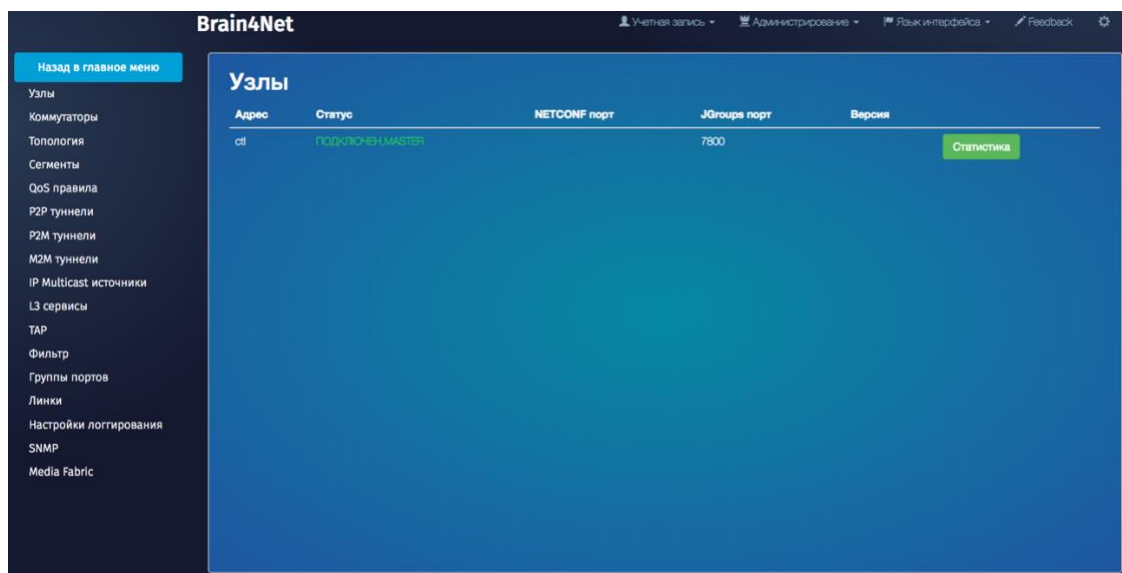


Рис. 6.2 Создание сетевых сервисов

Для подключения к контроллеру на коммутаторе указываются IP-адрес контроллера и порт (в случае наличия нескольких контроллеров указывается диапазон адресов);

Доступ на страницу настройки сетевых сервисов и осуществляется через элемент «Конфигурировать» меню «Управление» списка сетевых ресурсов.

Меню содержит следующие элементы:

- Узлы;
- Коммутаторы;
- Топология;
- Сегменты;
- QoS правила;
- P2P туннели;
- P2M туннели;
- M2M туннели;
- IP Multicast источники;
- L3 сервисы;
- TAP;
- Фильтр;
- Группы портов;
- Линки;
- Настройки логгирования;
- SNMP;
- Media Fabric.

### **6.2.1. Узлы**

Узлы – элементы (вычислительные ресурсы) сети на которых расположены контроллеры. По каждому из узлов возможно просмотреть:

- IP адрес;
- статус;
- порты подключения;
- версию программного обеспечения
- статистику.

Статистика отображает использование памяти, жизненный цикл потоков и статистику по использованию файловой системы.

### **6.2.2. Коммутаторы**

Коммутаторы – OF коммутаторы, имеющие связность с контроллером и формирующие топологию передачи данных сети SDN. Каждый коммутатор перед подключением, по необходимости, должен быть переведён в режим OpenFlow Switch (OFS), и на нём должен обязательно должен быть настроен IP адрес контроллера. В процессе регистрации коммутатору присваивается идентификатор (SID) и TCP порт для общения контроллер-коммутатор.

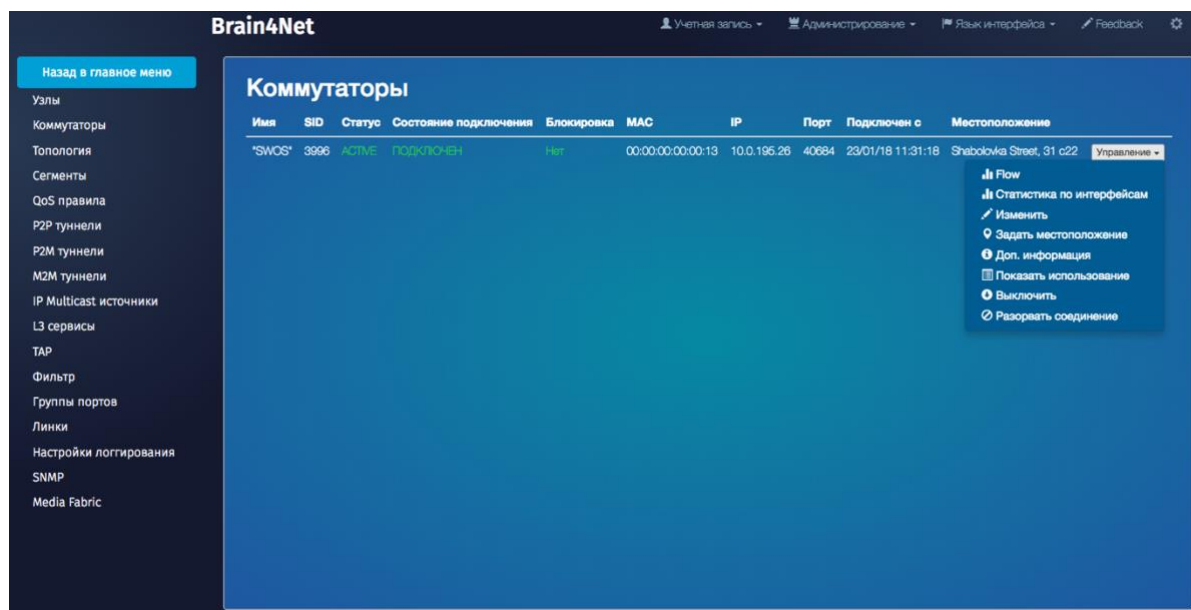


Рис. 6.3 Список коммутаторов

Элемент отображает список связанных с контроллером коммутаторов, находящихся в активном состоянии. Для каждого коммутатора отображается следующий список параметров:

- Имя – произвольный набор символов в соответствии с дизайном сети. По умолчанию принимается равным MAC адресу устройства;
- SID – назначается контроллером;
- Статус – обозначает участие коммутатора в построении SDN топологии. Определяется пользователем. Имеет 2 состояния (Active/Inactive);
- Состояние подключения – отображает наличие связности с контроллером;
- Статус блокировки – определяемое контроллером состояние коммутатора. Может быть активировано в случае нештатного поведения устройства;
- MAC адрес;
- IP адрес;
- Порт – TCP порт для связки с контроллером;
- Время нахождения в активном состоянии;
- Местоположение – задаваемое географическое местоположение устройства. Необходимо для построения топологии.

Управление коммутатором возможно через контекстное меню Управление. Оно позволяет просмотреть статистику и настроить параметры коммутатора.

Меню содержит следующие элементы:

- Flow;
- Статистика по интерфейсам;
- Изменить;
- Задать местоположение;
- Доп информация;
- Показать использование;
- Включить/Выключить;
- Разорвать соединение.

### 6.2.3. Настройка коммутатора

Для настройки коммутатора необходимо произвести 3 обязательных операции:

**Первая операция** при создании сетевого окружения – "Включить". Тем самым пользователь меняет статус коммутатора, переводя его в активное состояние. Контроллер использует коммутатор со статусом "Active" в построении SDN топологии;

**Вторая обязательная операция** – "Задать местоположение". Это необходимо для отображения топологии сети;

**Третья операция** при настройке коммутатора – это создание сервисных интерфейсов (SI) на коммутаторе. При создании SI задаются следующие параметры:

- Физический порт коммутатора;
- Инкапсуляция:
  - Access – трафик без VLAN ID;
  - VLAN ID;
  - QinQ.
- VLAN ID – становится активным при выборе типов инкапсуляции VLAN ID и QinQ;
- Inner VLAN ID – становится активным при выборе типа инкапсуляции QinQ.

Создание SI возможно через использование элемента меню Изменить.

Управление коммутатором позволяет принудительно разорвать связность с контроллером для повторного обнаружения коммутатора. Элемент меню «Разорвать соединение».

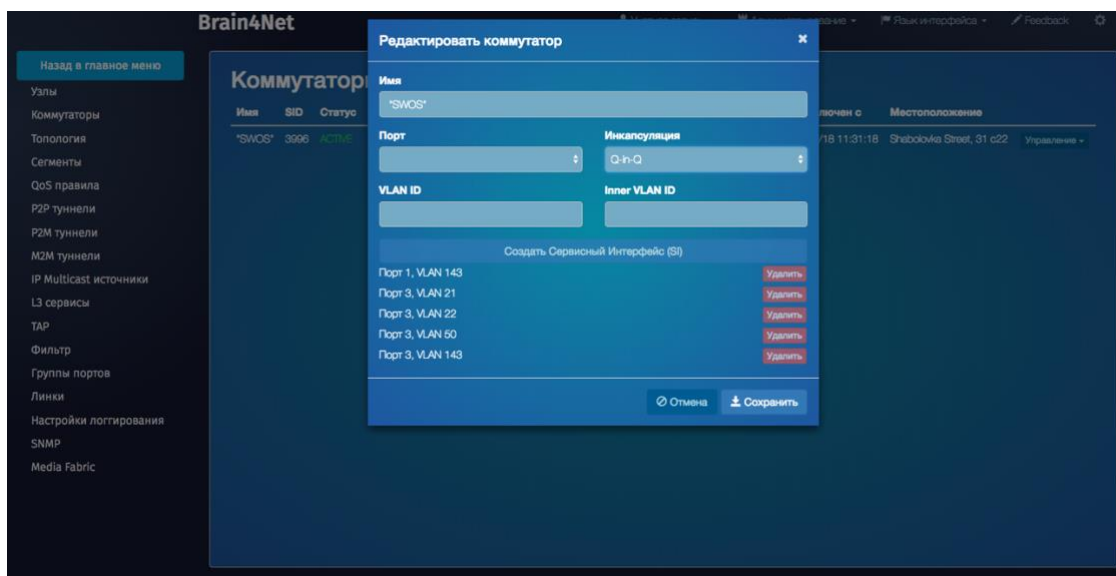


Рис. 6.4 Создание сервисных интерфейсов.

### 6.2.4. Сбор информации о коммутаторе

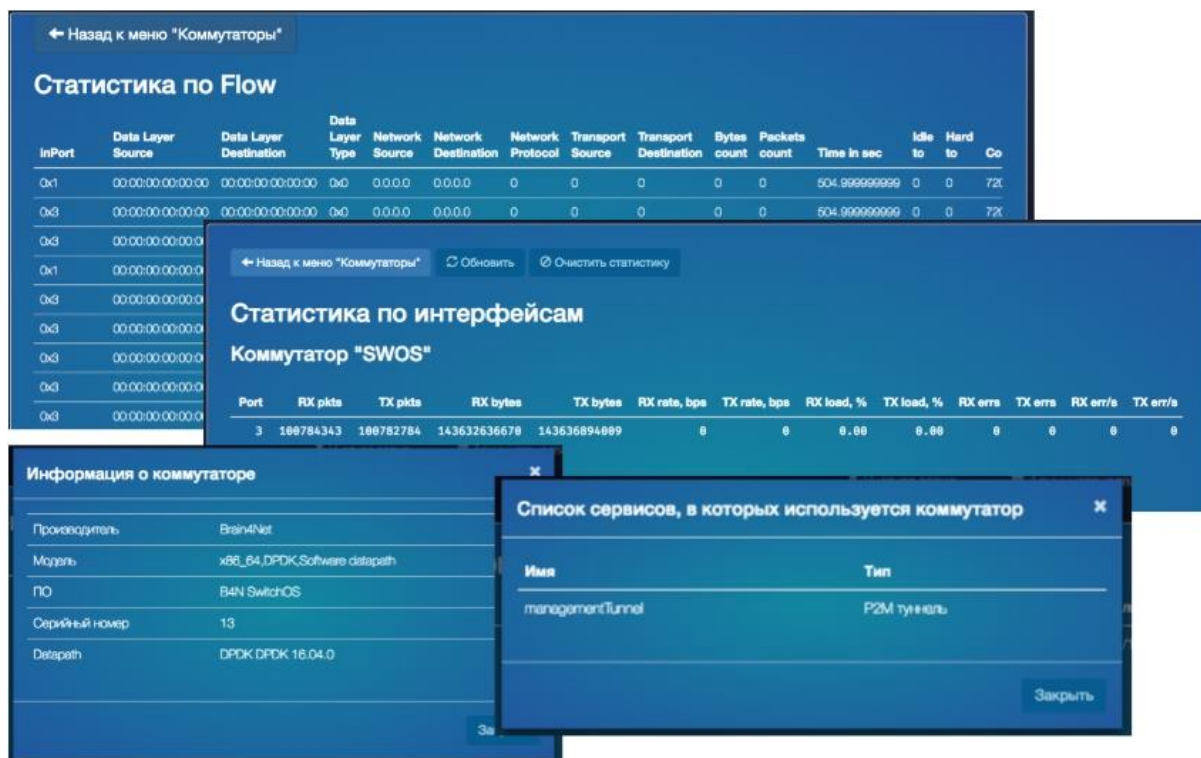
Платформа позволяет показать статистику по потокам (Flow), по интерфейсам и сервисам, в обработке/передаче которых участвует коммутатор. Flow показывает статистику по существующим

потокам. Под Flow понимается поток данных, соответствующий заданным критериям (VLAN tag, VLAN PCP и т.д.). В статистике отображается:

- Порт с настроенным сервисным портом in Port;
- MAC адрес источника и получателя;
- Ethertype;
- IP адрес источника и получателя;
- Протокол, номер порта источника и получателя;
- Счётчик переданных byte и пакетов.

Статистика по интерфейсам показывает статистику по переданным/полученным данным в пакетах, byte, скорость потоков, процент загрузки порта, количество ошибок для обоих направлений;

- Показать использование отображает список сервисов;
- Элемент меню Доп информация отображает информацию для инвентаризации устройств:
- Производитель;
- Модель;
- Версия установленного ПО;
- Серийный номер;
- Datapath.



The screenshot displays the 'Статистика по Flow' (Flow Statistics) section, which includes a table with columns: InPort, Data Layer Source, Data Layer Destination, Data Layer Type, Network Source, Network Destination, Network Protocol, Transport Source, Transport Destination, Bytes count, Packets count, Time in sec, Idle to, Hard to, and Co. Below this, the 'Статистика по интерфейсам' (Interface Statistics) section for 'Коммутатор "SWOS"' is shown, with a table listing ports and their respective RX/TX statistics. A modal window titled 'Информация о коммутаторе' (Switch Information) is open, showing details like Manufacturer (Brain4Net), Model (x86\_64.DPDK.Software datapath), OS (B4N SwitchOS), Serial number (13), and Datapath (DPDK DPDK 16.04.0). Another modal window titled 'Список сервисов, в которых используется коммутатор' (List of services using the switch) is also visible, showing a table with columns 'Имя' (Name) and 'Тип' (Type), listing 'managementTunnel' as a 'P2M туннель' (P2M tunnel).

Рис. 6.5 Сбор информации о коммутаторе

### 6.2.5. Топология

Топология – представляет собой интерактивную схему компонентов, привязанных к географическим точкам на карте с возможностью масштабирования. При масштабировании изменяется детализация схемы.

При необходимости пользователь переходит к описанию любого компонента и его настройке. На схеме можно отобразить: загруженность линков и сегментов; имя и IP адрес устройства. Настройка

устройства возможна по контекстному меню и позволяет отобразить информацию о устройстве, использование его в настроенных сервисах, изменить параметры. Проблемные компоненты подсвечиваются с указанием ошибок. Пример ошибки: нет связи между коммутаторами.

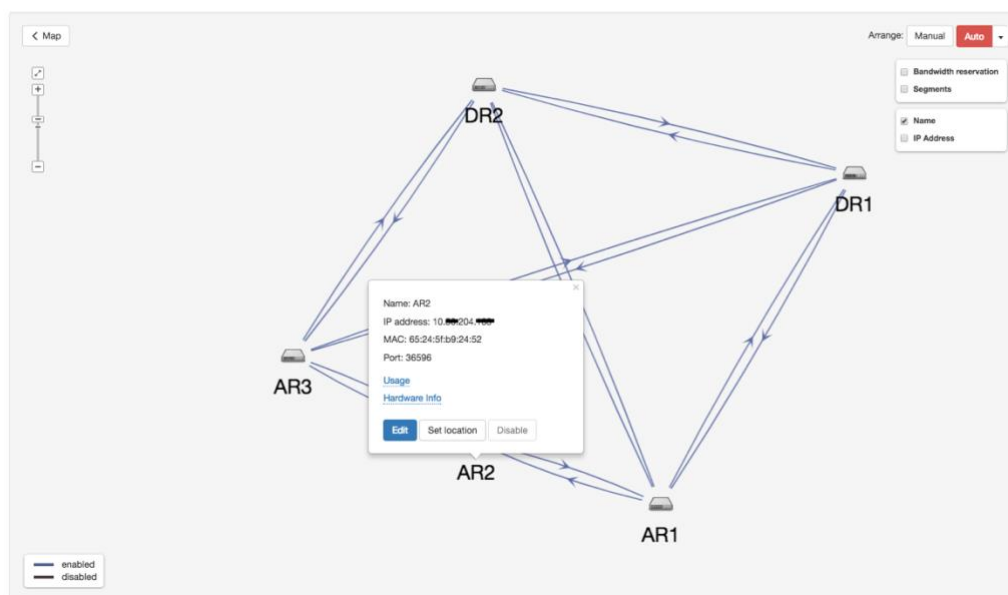


Рис. 6.6 Топология сети

### 6.2.6. Сегменты

Сегменты – виртуальные однонаправленные каналы передачи данных между OF-коммутаторами. Список содержит информацию о конечных точках сегмента и описание пути – транзитных коммутаторов. Описание каждой точки содержит имя коммутатора и номер порта.

### 6.2.7. QoS правила

Платформа поддерживает 8 классовую модель QoS. По умолчанию, в системе определены три класса:

- RealTime – трафик реального времени (голос, видео, стриминг);
- Business Critical – трафик критичных приложений;
- BestEffort – обычный трафик.

Элемент меню QoS позволяет определить модель правила классификации и правила QoS. Параметры настройки QoS определяются на этапе проектирования/дизайна сети в соответствии с "Руководством по дизайну".

Для инициализации/создании в SDN сети модели QoS необходимо в закладке QoS Settings/Настройки QoS задать следующие глобальные доменные настройки:

- Создать классы QoS или редактировать созданные по умолчанию;
- Определить внутренние тэги (метки трафика), соответствующие определённому классу;
- Настроить маппирование трафика определённого класса в аппаратную очередь коммутатора;
- Определить, по необходимости, коэффициент переподписки (KOver) для класса трафика;



- Указать значения тэга трафика какого класса не учитываются при построении сегментов;
- В элементе Access port Queue Name указать, в какой класс трафика помещать трафик с SI с инкапсуляцией "Access";
- Указать максимальную зарезервированную полосу для трафика в процентах от полной полосы пропускания порта коммутатора.

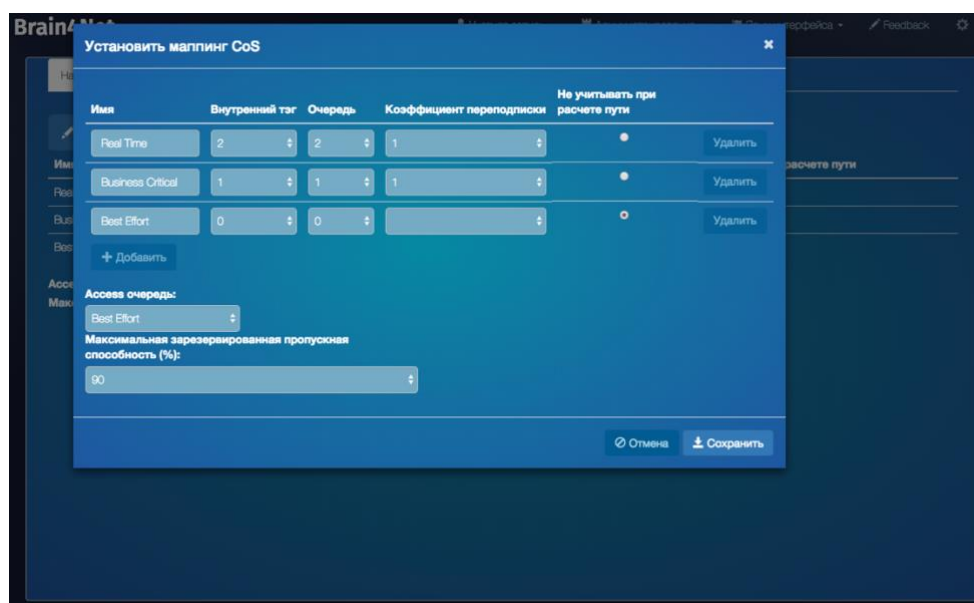


Рис.6.7 Настройки QoS

Существующие настройки QoS можно редактировать. Для настройки/создания классификации входящего трафика необходимо в закладке Classifier/Классификатор задать следующие глобальные доменные настройки:

- Имя классификатора;
- Назначить модель применения классификатора/Тип трафика:
  - PCP trust diffserv. Для входящего трафика назначается определённый класс на основе значения полей IEEE 802.1p PCP;
  - Untrust. Для всего входящего трафика назначается определённый класс.
- Создать матрицу классификации для модели PCP trust diffserv:
  - Указать поле, на основании которого принимается решение. По умолчанию Outer\_VLAN\_PCP;
  - Указать значение PCP входящего трафика;
  - Указать класс трафика.

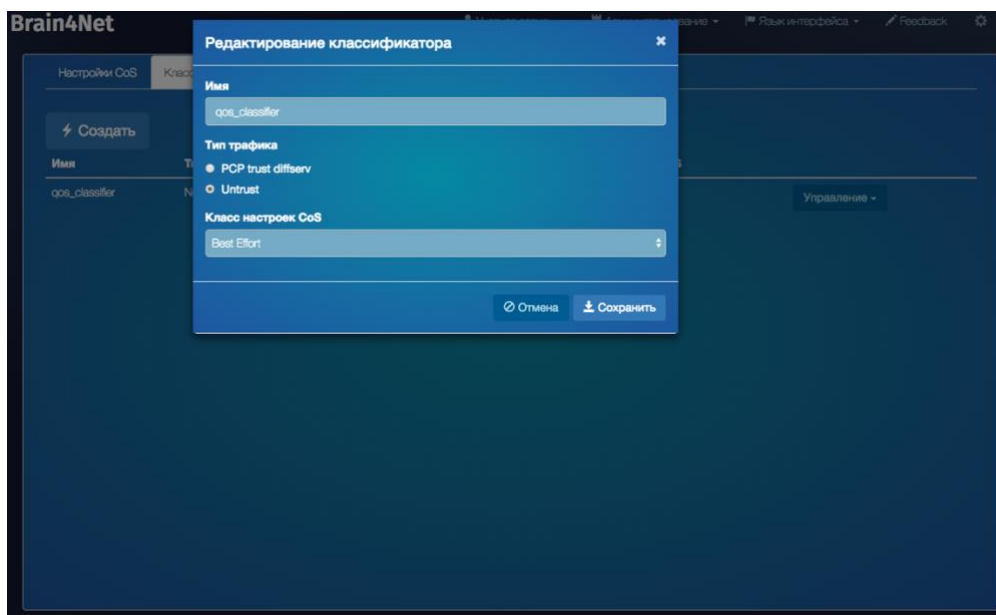


Рис.6.8 Настройки Классификаторов

Существующие классификаторы можно редактировать. Для настройки/создания правил QoS обработки трафика необходимо в закладке QoS Rules/QoS правила задать следующие глобальные доменные настройки:

- Назначить имя правила;
- Определить MBR – maximum bit rate/максимальная скорость передачи. Необходимая скорость передачи трафика на канале в Mbps;
- Указать имя используемого классификатора;
- В случае использования классификатора с моделью PCP trust diffserv:
  - Настроить распределение полосы пропускания между классами.

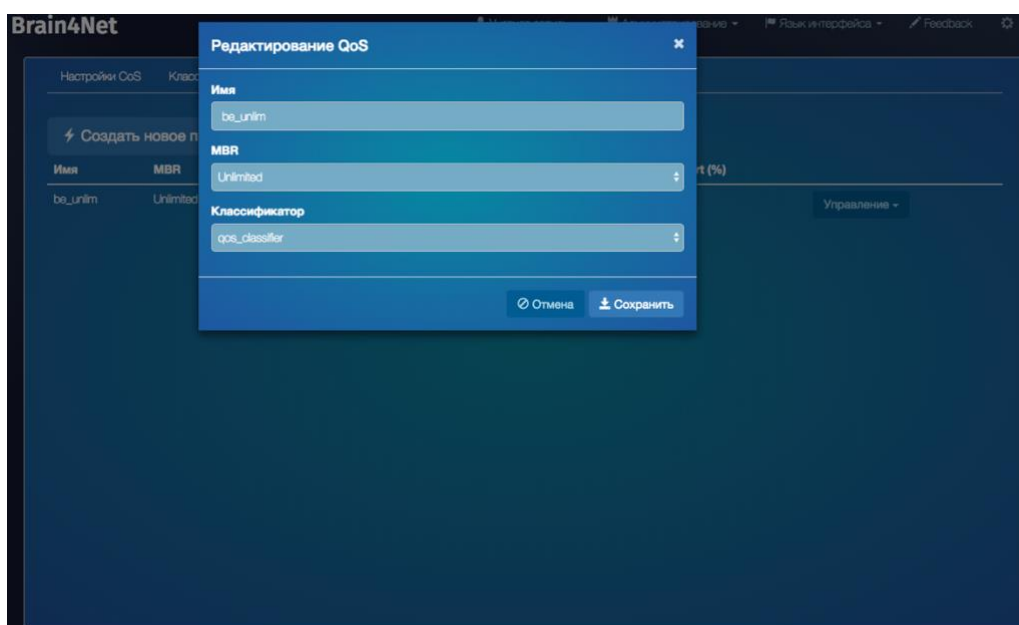


Рис.6.9 Настройки QoS правил

Коэффициент переподписки означает – предопределённое увеличение полосы пропускания трафика в число раз от заданной полосы. Маппирование означает – правило по которому трафик помещается куда-либо на основании значения указанных полей заголовка. Существующие правила можно редактировать и удалять.

### 6.2.8. P2P туннели

Point-to-Point (P2P) туннель или E-line в классификации MEF. Сервис организует транспортный туннель, соединяющий сервисные порты (SI) 2-х коммутаторов. SI разделяются как SI источник (source) и приёмник (destination). Трафик данного сервиса поддерживает фреймы 802.1q или 802.1ad.

Перед созданием P2P туннеля убедитесь в том, что:

- Коммутаторы находятся в активном состоянии;
- Коммутаторы присутствуют в топологии;
- На коммутаторах созданы сервисные интерфейсы SI.

Интерфейс платформы предусматривает создание двунаправленного туннеля за один шаг создания туннеля. В архитектуре P2P туннеля возможно создание резервного (backup) сервисного интерфейса приёмника (destination). Резервный destination SI может быть определён на том же самом коммутаторе или на любом другом, определённом на этапе дизайна сети. P2P туннель может быть двух типов:

- Статический – существует независимо от наличия пакетов данных на туннеле;
- Динамический – туннель создаётся контроллером при появлении потока данных и ликвидируется при отсутствии пакетов по истечении timeout равного 60с.

P2P Туннели									
⚡ Создать новый P2P туннель									
Имя	Тип	Откуда	Куда	QoS	Нахождение пути	Обратный туннель	Резервный получатель	Статус	
B2B_L3	Статический	AR2, Порт 1, VLAN 100	DR1, Порт 5, VLAN 3	QR_Untrust	Кратчайший путь	Симметричный QoS: QR_Untrust		РАБОТАЕТ	Управление ▼
IPTV_Unicast	Статический	AR1, Порт 16, VLAN 2137	DR1, Порт 27, VLAN 2030	QR_Trust	Кратчайший путь	Симметричный QoS: QR_Trust		РАБОТАЕТ	Управление ▼
IPTV_Unicast_ORNB-ATS56-DSW5	Статический	AR2, Порт 16, VLAN 2131	DR1, Порт 27, VLAN 2031	QR_Trust	Кратчайший путь	Симметричный QoS: QR_Trust		РАБОТАЕТ	Управление ▼
RoSL_SPD_VPN	Статический	DR1, Порт 5, VLAN 20	AR2, Порт 16, VLAN 78	QR_Untrust	Кратчайший путь	Симметричный QoS: QR_Untrust		РАБОТАЕТ	Управление ▼

Рис. 6.10 P2P туннели

При создании туннеля определяются следующие параметры:

- Имя туннеля;
- Тип туннеля (статический; динамический);
- Указание коммутатора и сервисного интерфейса для обеих сторон туннеля;
- Указание трансляции статуса порта в случае состояния «Administrative down» на одном из концов туннеля;
- Указание предопределённого QoS правила для пути прохождения трафика;
- Комментарий;
- Принцип установления пути прохождения трафика:
  - Кратчайший путь, определяемый контроллером;
  - Указание пользователем транзитных коммутаторов;
- Реверсный путь – создание обратного пути прохождения трафика.
- Симметричность обратного пути:
  - Симметрично прямому пути;
  - Указание пользователем транзитных коммутаторов;
- Выбор предопределённого QoS правила для резервного пути;
- Наличие резервирования destination SI;

- Указание коммутатора и сервисного интерфейса для destination SI;
- Принцип установления пути прохождения трафика для destination SI:
  - Кратчайший путь, определяемый контроллером;
  - Указание пользователем транзитных коммутаторов.

Для созданного P2P туннеля пользователем с правами администратора возможны следующие действия:

- Редактирование параметров;
- Отображение топологии сервиса (туннеля);
- Отображение статистики;
- Включение режима отладки с выводом сообщений;
- Удаление.

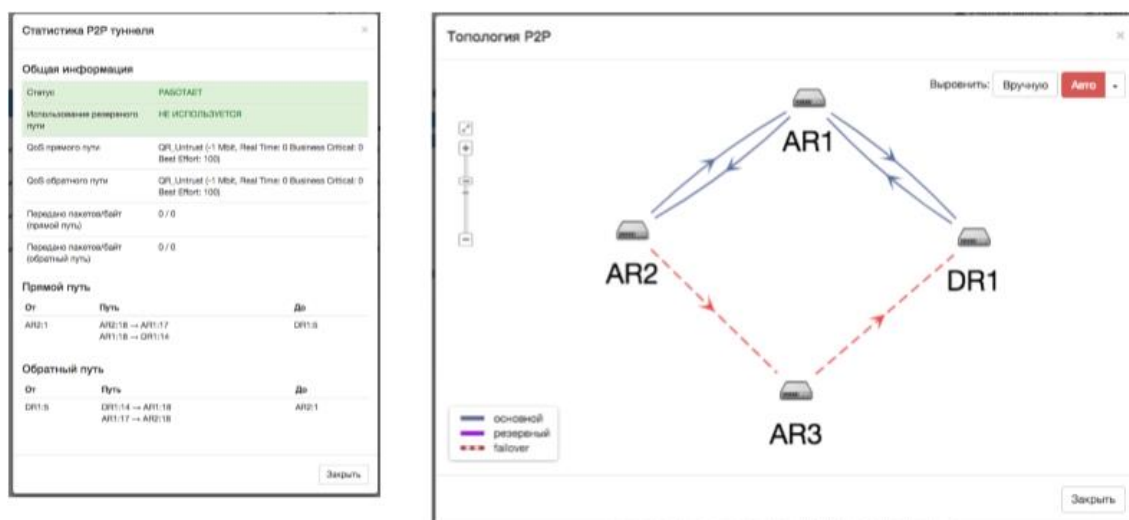


Рис. 6.11 P2P туннели Статистика.

### 6.2.9. P2M туннели

Point to Multipoint (P2M) или E-TREE в классификации MEF. Сервис организует туннель между двумя и более сервисными интерфейсами (SI).

Каждый SI имеет роль:

- Root;
- Leaf.

SI с ролью Root может передавать трафик на любой другой SI внутри сервиса. SI, имеющий роль Leaf, имеет право отправлять трафик только на SI с ролью Root. В топологии сервиса минимум один SI должны иметь роль Root. Трафик данного сервиса поддерживает фреймы 802.1q или 802.1ad.

Перед созданием P2M туннеля убедитесь в том, что:

- Коммутаторы находятся в активном состоянии и присутствуют в топологии;
- На коммутаторах созданы сервисные интерфейсы SI;
- На этапе дизайна определена сервисная модель и разделение SI на роли.

P2M туннели					
<a href="#">⚡ Создать P2M туннель</a>					
Имя	Время жизни сессии / туннеля / MAC (с)	Размер таблицы MAC	MAC learning	Выходные точки	
AR1_B2C_IPOE	10 / 60 / 300	100	DFI with FIB on leaf	AR1, Порт 16, VLAN 2107, QoS QR_Untrust, Role ROOT (DFI) DR1, Порт 6, VLAN 165, QoS QR_Untrust, Role LEAF, Резервный: DR2, Порт 8, VLAN 165 DR1, Порт 8, VLAN 165, QoS QR_Untrust, Role LEAF, Резервный: DR2, Порт 6, VLAN 165	<a href="#">Управление ▾</a>
AR1_B2C_PPOE	10 / 60 / 300	100	DFI with FIB on leaf	AR1, Порт 16, VLAN 2117, QoS QR_Untrust, Role ROOT (DFI) DR1, Порт 6, VLAN 1440, QoS QR_Untrust, Role LEAF, Резервный: DR2, Порт 8, VLAN 1440 DR1, Порт 8, VLAN 1440, QoS QR_Untrust, Role LEAF, Резервный: DR2, Порт 6, VLAN 1440	<a href="#">Управление ▾</a>
AR2_B2C_IPOE_ORNB-ATSS6-DSW5_DHCP	10 / 60 / 300	2000	DFI with FIB on leaf	DR1, Порт 6, VLAN 166, QoS QR_Untrust, Role LEAF, Резервный: DR2, Порт 6, VLAN 166 DR1, Порт 8, VLAN 166, QoS QR_Untrust, Role LEAF, Резервный: DR2, Порт 8, VLAN 166 AR2, Порт 16, VLAN 2003, QoS QR_Untrust, Role ROOT (DFI)	<a href="#">Управление ▾</a>

Рис. 6.12 P2M туннели.

При создании туннеля определяются следующие параметры:

- Имя;
- Параметры, определяющие время жизни сервиса:
  - Время неактивности сессии, после которого считается что сессия не активна, определяется в секундах. По истечении этого времени сессия удаляется. Значение по умолчанию – 900с;
  - Время неактивности туннеля, после которого считается что туннель не активен. Определяется в секундах. По истечении этого времени туннель удаляется. Значение по умолчанию – 900с;
- Параметры, определяющие размер таблицы MAC адресов:
  - Время, в течении которого MAC адрес считается активным, определяемое в секундах. Значение по умолчанию – 3600с;
  - Размер таблицы MAC адресов на контроллере. Значение по умолчанию 100шт.;
- Способ изучения (learning) MAC адресов:
  - Classic
  - DFI с FIB на root и leaf;
  - DFI с FIB на leaf;
  - DFI без изучения MAC;
- Комментарий/описание к туннелю;
- Создание конечных точек туннеля (минимум 2шт):
  - Коммутатор;
  - Сервисный порт;
  - Указание предопределённого QoS правила для пути прохождения трафика;
  - Указание роли точки (Root/Leaf);
  - Использование резервного сервисного интерфейса:
    - Коммутатор;
    - Сервисный интерфейс;
- Назначение конечной точки как Default Forwarding Interface (DFI) или как точки куда направляется трафик с неизвестным MAC адресом;
- Указание группы портов:
  - Указание предопределённой группы портов;
  - Указание предопределённого QoS правила;
  - Определение VLAN.

Для созданного P2M туннеля возможны следующие действия:

- Редактирование параметров;
- Отображение топологии сервиса (туннеля);
- Отображение статистики;
- Включение режима отладки с выводом сообщений;

- Удаление.

### 6.2.10. M2M туннели

Multipoint to Multipoint (M2M) или E-LAN в классификации MEF. Сервис организует туннель между двумя и более сервисными интерфейсами (SI).

Сервис представляет собой распределённый bridge-domain, использующий механизм mac-learning через контроллер для заполнения MAC-таблицы. При этом на каждом узле организуется отдельный bridge-domain, связанный с другими через транспортные туннели. Наряду с общей таблицей MAC адресов сервиса на каждом из узлов, где присутствуют сервисные интерфейсы образующие локальные bridge-domain присутствуют частичные таблицы MAC-адресов.

Перед созданием M2M туннеля убедитесь в том, что:

- Коммутаторы находятся в активном состоянии и присутствуют в топологии;
- На коммутаторах созданы сервисные интерфейсы SI;
- На этапе дизайна определена сервисная модель и разделение SI на роли.

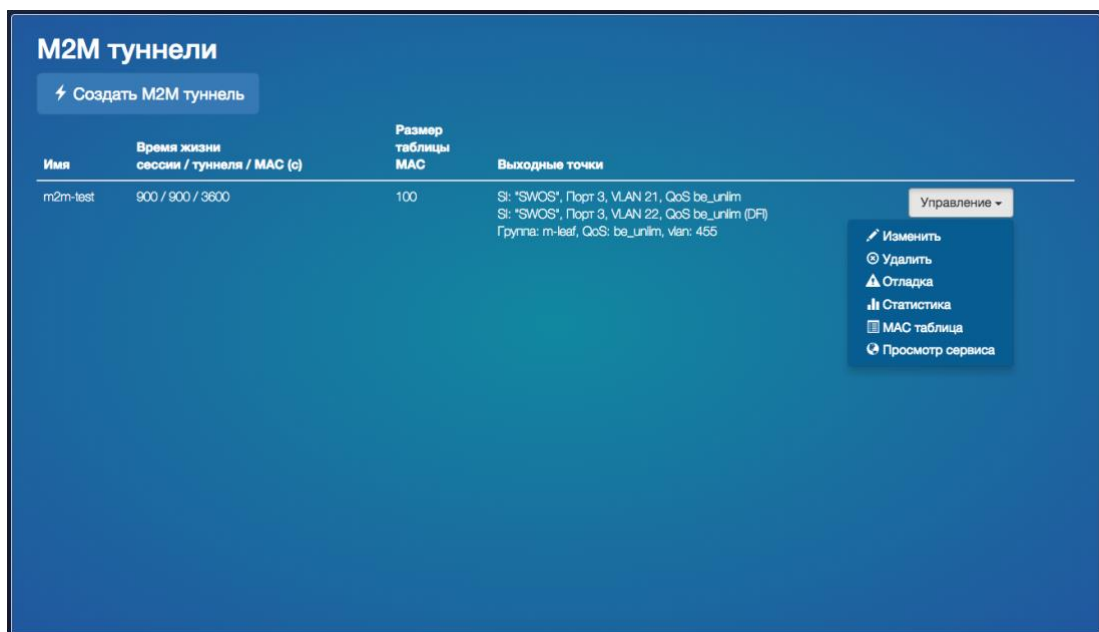


Рис. 6.13 M2M туннели.

При создании туннеля определяются следующие параметры:

- Имя;
- Параметры, определяющие время жизни сервиса:
  - Время неактивности сессии после которого считается что сессия не активна, определяется в секундах. По истечении, которого сессия удаляется. Значение по умолчанию – 900с;
  - Время неактивности туннеля после которого считается что туннель не активен. Определяется в секундах. По истечении которого туннель удаляется. Значение по умолчанию – 900с;
- Параметры, определяющие размер таблицы MAC адресов:

- Время, в течении которого MAC адрес считается активным, определяемое в секундах. Значение по умолчанию – 3600с;
  - Размер таблицы MAC адресов на контроллере. Значение по умолчанию 100шт.;
- Комментарий/описание к туннелю;
- Создание конечных точек туннеля (минимум 2шт). Для каждой точки определяются:
  - Коммутатор;
  - Сервисный порт;
  - Указание предопределённого QoS правила для пути прохождения трафика;
- Использование резервного сервисного интерфейса:
  - Коммутатор;
  - Сервисный интерфейс.
- Назначение конечной точки как Default Forwarding Interface (DFI) или как точки куда направляется трафик с неизвестным MAC адресом;
- Указание группы портов:
  - Указание предопределённой группы портов;
  - Указание предопределённого QoS правила;
  - Определение VLAN.

Для созданного M2M туннеля возможны следующие действия:

- Редактирование параметров;
- Отображение топологии сервиса (туннеля);
- Отображение статистики;
- Включение режима отладки с выводом сообщений;
- Удаление.

### **6.2.11. IP multicast источники**

IP Multicast source tree. Данный сервис состоит из:

- одного или более сервисных интерфейсов, к которым подключены источники IP Multicast, вещающие в режиме статической подписки;
- одного или более сервисных интерфейсов, к которым подключены абоненты, подписывающиеся на группы IP Multicast по протоколу IGMPv2/v3.

Сервис организует дерево распространения IP Multicast внутри домена. Корнями деревьев являются определённые источники. Подписчики присоединяются к ближайшему узлу дерева, если не заданы ограничения (указание конкретного источника, запрет на подключение к источнику данному абоненту и т.д.). Трафик данного сервиса внутри домена идёт как Ethernet-фрейм с IP-payload, без дополнительной инкапсуляции. Трафик сервиса полностью мапируется в класс QoS RealTime.



Источники IP Multicast						
⚡ Создать новый источник IP Multicast						
Имя	Коммутатор/Порт	Backup коммутатор/порт	Querier Address	Разрешенные порты	Группы	
IPTV	DR1, Порт 27, VLAN 2137	DR2, Порт 27, VLAN 2137	10.0.0.1	AR1, Порт 16, VLAN 2137 AR2, Порт 16, VLAN 2131 AR1, Порт 1, VLAN 500 AR2, Порт 1, VLAN 400	225.3.1.0 / 24, 10 MBit 225.50.50.0 / 24, 10 MBit 225.52.3.0 / 24, 10 MBit 225.52.13.0 / 24, 10 MBit 225.52.20.0 / 24, 10 MBit 225.52.23.0 / 24, 10 MBit 225.56.221.0 / 24, 10 MBit 225.56.223.0 / 24, 10 MBit 225.56.225.0 / 24, 10 MBit 233.97.176.0 / 24, 10 MBit 235.56.12.0 / 24, 10 MBit 233.99.61.0 / 24, 20 MBit 233.171.199.0 / 24, 20 MBit 225.52.53.0 / 24, 20 MBit 225.52.43.0 / 24, 20 MBit	Управление ▾

Рис. 6.14 Сервис IP Multicast источники.

При создании сервиса определяются следующие параметры:

- Имя;
- Источник:
  - Коммутатор, к которому подключён источник;
  - Порт коммутатора, к которому подключён источник.
- Querier адрес – IP адрес источника multicast трафика;
- Резервный источник:
  - Коммутатор, к которому подключён источник;
  - Порт коммутатора, к которому подключён источник.
- Список разрешённых внешних физических портов потребителей с указанием:
  - Коммутатора;
  - Порт.
- Указание IP адреса multicast группы;
- Маски адреса;
- GBR – Guaranteed Bit Rate – для группы в Мб/с;

Для созданного сервиса возможны следующие действия:

- Редактирование параметров;
- Отображение статистики;
- Удаление.

Использование IP Multicast источника позволяет сохранить полосу пропускания, которая в свою очередь позволит снизить объёмы трафика за счёт одновременной доставки одного и того же потока информации нескольким адресатам при групповой передаче.

### 6.2.12. L3 сервис.

L3 предоставляет клиентам услугу L3-маршрутизации между непосредственно подключёнными сетями клиента с возможностью определения статических маршрутов.

Сервис поддерживает топологию Full-Mesh

В Full-Mesh топологии разрешается взаимодействие между любыми сетями клиента.

Перед созданием L3 VPN Static сервиса убедитесь в том, что:

- Коммутаторы находятся в активном состоянии;
- Коммутаторы присутствуют в топологии;
- На коммутаторах созданы сервисные интерфейсы SI;

- Определён сервис M2M используемый в сервисе;
- На этапе дизайна определена сервисная модель и разделение SI на роли.
- 

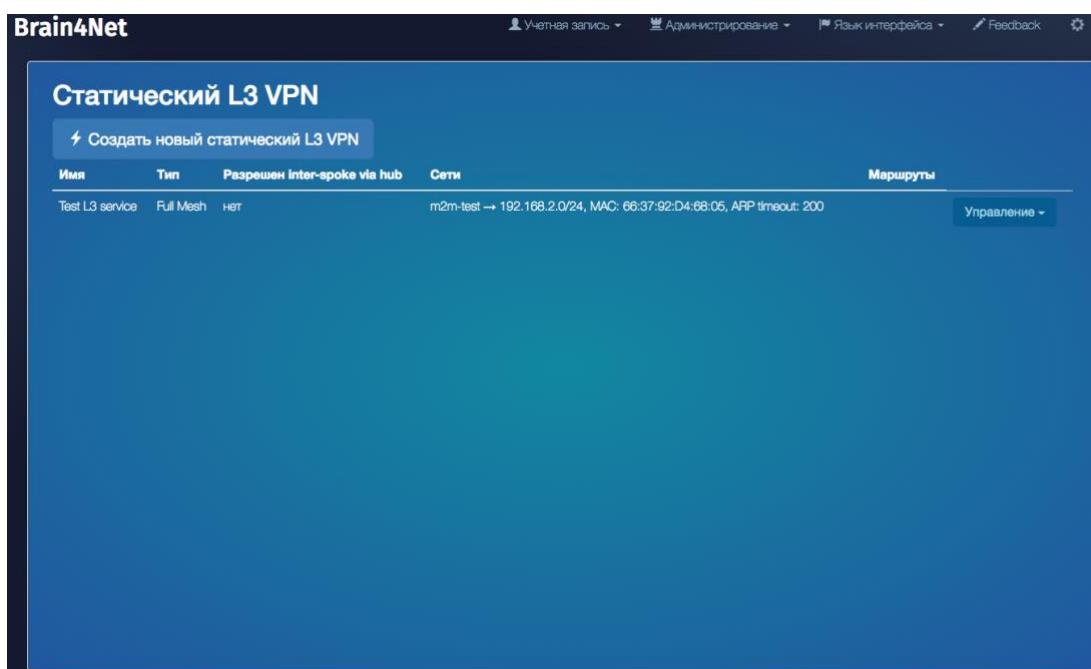


Рис. 6.15 Сервис Статический L3 VPN.

При создании туннеля настраиваются следующие параметры:

- Имя сервиса;
- Сервисные интерфейсы SI участвующие в топологии сервиса. После определения параметров интерфейса нужно выбрать кнопку Add для добавления в сервис. В качестве SI могут использоваться:
  - M2M сервис, для которого указываются:
    - Имя предопределенного сервиса;
    - IP адрес, назначаемый на L3 интерфейс (SI) сервиса M2M;
    - Маска IP адреса;
    - MAC адрес интерфейса. MAC может быть сгенерирован платформой (выбором опции Generate заголовке параметра или задан вручную в формате XX:XX:XX:XX:XX:XX;
  - Порт коммутатора, для которого указываются:
    - Имя коммутатора;
    - Сервис интерфейс;
    - Правило QoS;
    - IP адрес, назначаемый на интерфейс коммутатора (SI);
    - Маска IP адреса;
    - MAC адрес интерфейса. MAC может быть сгенерирован платформой (выбором опции Generate заголовке параметра или задан вручную в формате XX:XX:XX:XX:XX:XX;
- Статические маршруты для подключённых сетей. Для каждого из которых задаётся:
  - IP адрес сети на которую задаётся статический маршрут;
  - Маска IP адреса сети;
  - Имя L3 SI (SVI) интерфейса, определённого на предыдущем шаге;
  - IP адрес L3 SI (SVI) интерфейса, определённого на предыдущем шаге. Поле Gateway.

Для созданного сервиса возможны следующие действия:

- Редактирование параметров;
- Удаление.

### 6.2.13. Фильтры

Элемент меню, позволяющий настроить правила фильтрации трафика, поступающего на сервисный интерфейс SI.

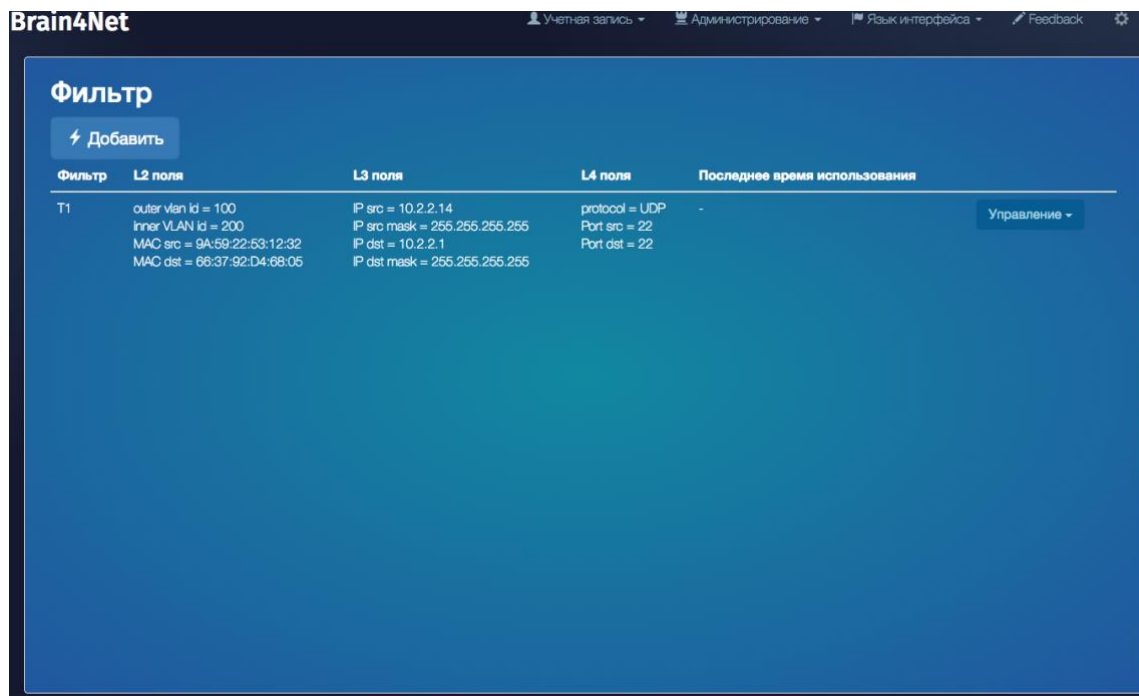


Рис. 6.16 Фильтр.

При создании фильтра определяются следующие параметры:

- Название;
- Уровень модели OSI (транспортный L2, сетевой L3, сеансовый L4) на основе полей которого будет фильтроваться трафик;
- Для L2 уровня:
  - Внешний/внутренний VLAN тэг IEEE 802.1ad;
  - Поле PCP IEEE 802.3p, Внешнего/внутреннего VLAN тэг;
  - MAC адрес источника/MAC адрес назначения;
- Для L3 уровня:
  - IP адрес источника/маска IP адреса источника;
  - IP адреса назначения/Маска IP адреса назначения;
- Для L4 уровня:
  - Протокол (TCP / UDP / ICMP);
  - Номер порта источника;
  - Номер порта назначения.

Созданные фильтры используются при создании сервиса TAP. Фильтры из списка возможно редактировать и удалять.

### 6.2.14. Сервис TAP (Test Access Point)

Сервис B4N TAP (Test Access Point) позволяет получить и обеспечить доставку копии трафика любого сервиса, существующего на SDN сети или от устройств, осуществляющих зеркалирование трафика к получателям. Сервис позволяет построить интеллектуальную распределённую сеть, способную агрегировать, фильтровать и реплицировать трафик от различных участков сети и передавать его на централизованные системы управления, мониторинга и безопасности.

Перед созданием TAP сервиса убедитесь в том, что:

- Коммутаторы находятся в активном состоянии;
- Коммутаторы присутствуют в топологии;
- На этапе дизайна определена сервисная модель и разделение SI на роли;
- На коммутаторах созданы сервисные интерфейсы SI;
- Созданы фильтры трафика;
- Создан carrier Ethernet сервис который является источником трафика.

Решение B4N TAP позволяет с необходимой точностью выбрать трафик, соответствующий заданным критериям: принадлежности определённым виртуальным сетям, по IP адресу или номеру порта.

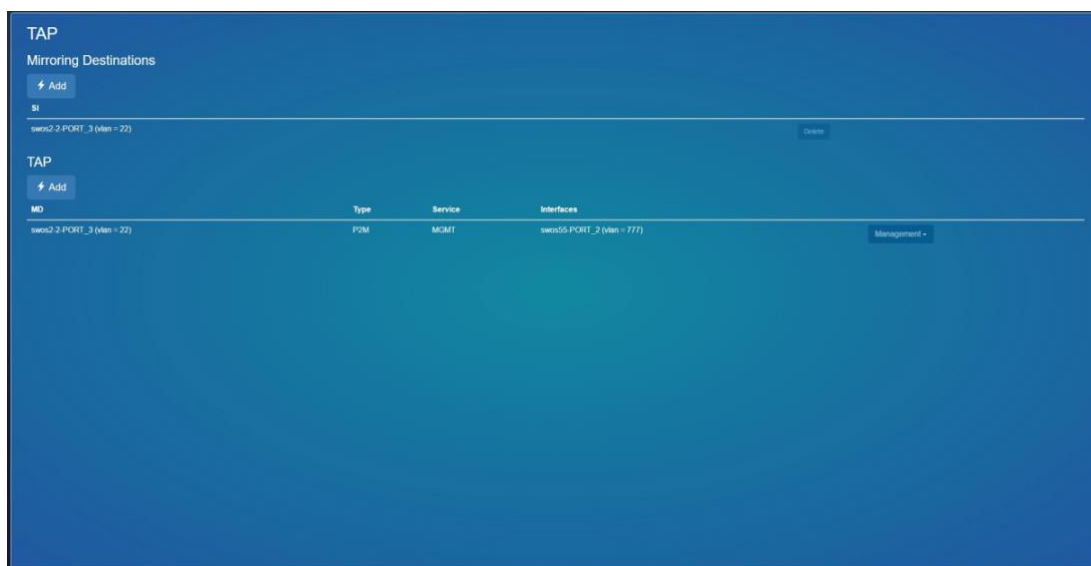


Рис. 6.17 TAP сервис.

При создании сервиса определяются следующие параметры:

Сервисный интерфейс SI, к которому подключён получатель трафика:

- Имя коммутатора;
- Порт коммутатора;

Источник трафика:

- Определение действия над трафиком. Будет трафик копироваться или перенаправляться на SI получателя;
- Тип источника трафика;
  - Сервисный интерфейс;

- Сервисный интерфейс SI, к которому подключён источник трафика;
- Сервис:
  - Тип сервиса
  - Наименование сервиса источника трафика;
  - Сервисный интерфейс SI, к которому подключён источник трафика;
- Предварительно созданный фильтр для выбранных интерфейсов.

Если в качестве источника трафика выбирается сервис, то необходимо определить:

- Тип сервиса (P2P / P2M / M2M / Multicast);
- Предварительно созданный на сети сервис;
- Интерфейсы.

При создании сервиса можно создать логическую топологию, при которой будет возможно:

- Распределить трафик, приходящий на один высокоскоростной порт, между несколькими портами, к которым подключены системы безопасности;
- Агрегировать трафика от нескольких портов 1GbE или 10GbE в порты 10GbE и 40GbE для максимальной загрузки систем безопасности;
- Реплицировать трафик на несколько портов, что позволяет разным системам мониторинга получить доступ к одной копии трафика.

### 6.2.15. Группы портов

Для целей удобства создания сервисов на большой группе интерфейсов платформа позволяет собрать несколько сервисных интерфейсов (SI) в группу сервисных интерфейсов. Группа портов может использоваться в сервисах M2M и P2M. Состав группы портов может меняться, интерфейсы могут как добавляться, так и удаляться. При создании одного сервиса возможно комбинировать группы портов и предварительно созданные SI. При создании одного сервиса нельзя использовать одну и ту же группу портов, даже с разными VLAN.

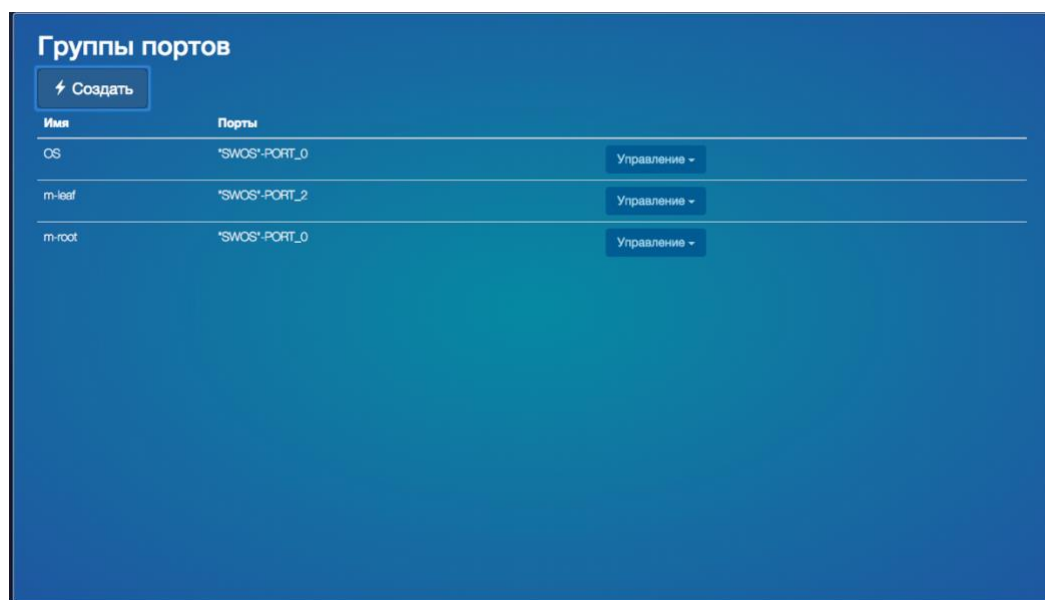
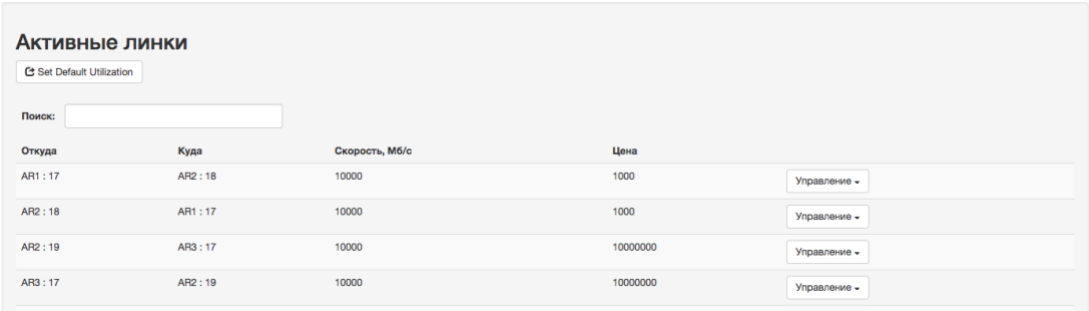


Рис. 6.18 Группы портов.

### 6.2.16. Линки

Линк – список физических каналов передачи данных между коммутаторами. Список отображает следующие параметры:

- Информацию о конечных точках сегмента, состоящую из имени коммутатора и номера порта;
- Скорость линка в Мб/с;
- Стоимость линка.



<input type="button" value="Set Default Utilization"/>			
Поиск: <input type="text"/>			
Откуда	Куда	Скорость, Мб/с	Цена
AR1 : 17	AR2 : 18	10000	1000
AR2 : 18	AR1 : 17	10000	1000
AR2 : 19	AR3 : 17	10000	10000000
AR3 : 17	AR2 : 19	10000	10000000

Рис. 6.19 Линки.

Цена линка учитывается при создании каналов передачи данных между OF-коммутаторами или сегментах. При построении сегмента выбираются линки с меньшей стоимостью. Стоимость линка рассчитывается автоматически как отношение референсной полосы пропускания к реальной в Мб/с:

**М**

Пользователь может заменить цену по умолчанию через элемент Управление.

### 6.2.17. Логгирование

Файлы отчётов (лог файлы) создаются на основе истории мониторинга устройств. Логи в системе формируются автоматически из создаваемых системой файлов. В коде системы прописана отправка сообщения в лог-файл после наступления определённых действий. Логи выводятся в едином списке с возможностью фильтрации по объекту, типу или состоянию (предупреждения или ошибки). Логи задач и событий привязаны к каждому модулю кластера. Предусмотрена возможность экспорта выбранного лога в файл.

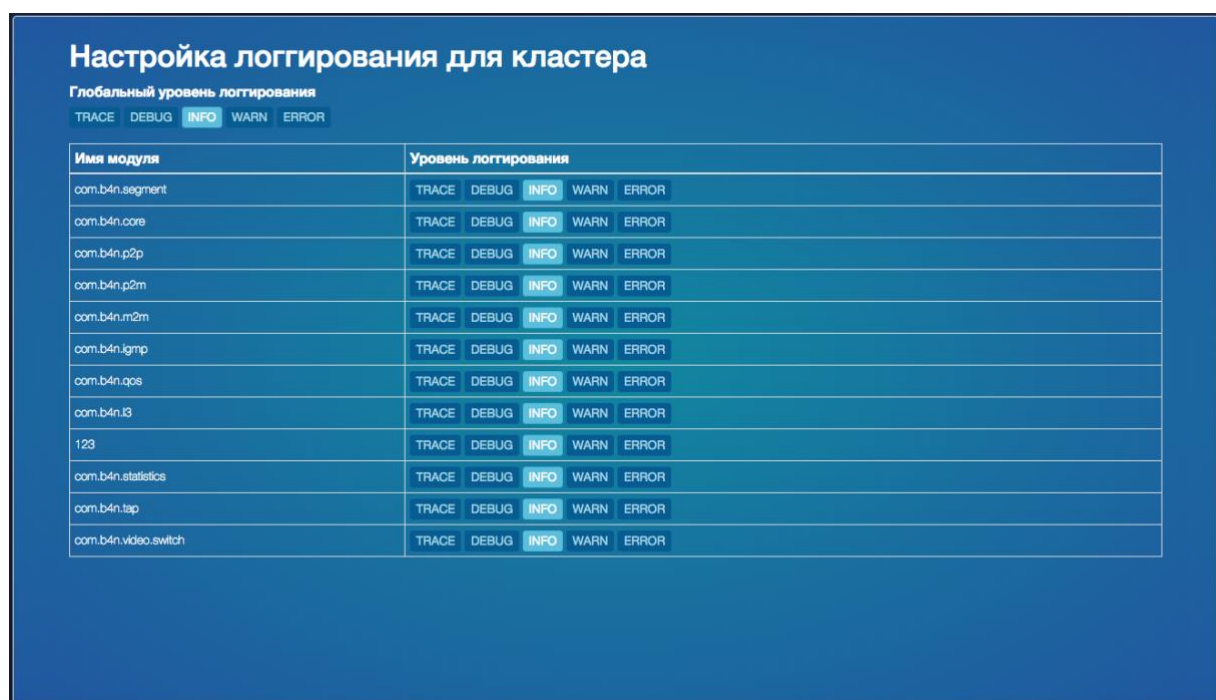


Рис. 6.20 Настройка логгирования.

При настройке логгирования для сетевых ресурсов (кластера) пользователь может определить следующие уровни логгирования для каждого из модулей (сегмент, QoS, TAP, P2P туннели и т.д.):

- **Error** – регистрация сообщений об ошибке. В лог выводятся особого класса ошибки. Такие ошибки приводят к неработоспособности системы в целом, или неработоспособности одной из подсистем. Чаще всего случаются фатальные ошибки из-за неверной конфигурации или отказов оборудования;
- **Warn** – регистрация предупреждений уровня неожиданных параметров вызова, странных форматов запроса, использование дефолтных значений взамен некорректных – все, что может свидетельствовать о нештатном использовании;
- **Info** – регистрация информационных сообщений разовых операций, которые повторяются крайне редко, но не регулярно. (загрузка конфига, плагина, запуск бэкапа);
- **Debug** – регистрация сообщений уровня отладки. Журналирование моментов вызова "крупных" операций. Старт/остановка потока, запрос пользователя и т.п.;
- **Trace** – регистрация всех событий. Отмечается на тот случай, если метод Отладка Debug не позволяет локализовать ошибку.

Уровни перечислены в порядке от минимального к максимальному количеству журналируемой информации.

### 6.2.18. SNMP

Элемент позволяет настроить SNMP протокол для сетевых ресурсов.

Настраиваются 2 блока параметров: SNMP протокола для платформы с указанием:

- IP адрес,
- UDP порт;
- Название Community.



По умолчанию, платформа настраивает IP адрес платформы и UDP порт 2020. Community присваивается значение – public.

Второй блок – получатель SNMP trap или система управления сетью. Определяются следующие параметры:

- IP адрес системы управления;
- UDP порт;
- Название Community;
- Разрешённые типы trap:
  - ifUpNotification;
  - ifDownNotification;
  - swUpNotification;
  - swDownNotification;
- Описание принимающей стороны.

### **6.2.19.      *Media Fabric***

Сервис Media Fabric организует транспорт multicast потоков от источников (камеры, медиашлюзы и пр.) к получателям (медиашлюзы, системы обработки, системы хранения).

Сеть SDN обеспечивает:

- маршрутизацию потоков от отправителя к получателям;
- организацию независимых RTPv2 (IEEE 1588-2008) доменов в режиме forward;
- организация управления устройствами.

Полная спецификация, описание дизайна решения и настройка сервиса доступно в “Руководстве по дизайну и настройке сервиса Media Fabric”.

## 7. КАТАЛОГИ

Раздел Каталоги служит для отображения доступных к использованию на платформе виртуальных функций и сервисов, работающих поверх SDN сети. Каталог также позволяет добавлять виртуальные функции, создавать и конфигурировать сервисы.

Под сервисом, в данном случае, понимается – развёрнутый на SDN-сети и платформе виртуализации (OpenStack) набор, в произвольной комбинации, виртуальных сетевых функций (VNF), сервисных интерфейсов (UNI) и соединений между ними роль которых выполняют транспортные сервисы (P2P, P2M, M2M) SDN-сети.

Каталог представляет собой список, включающий в себя:

- VNF;
- Шаблоны сервисов (преднастроенные наборы элементов).

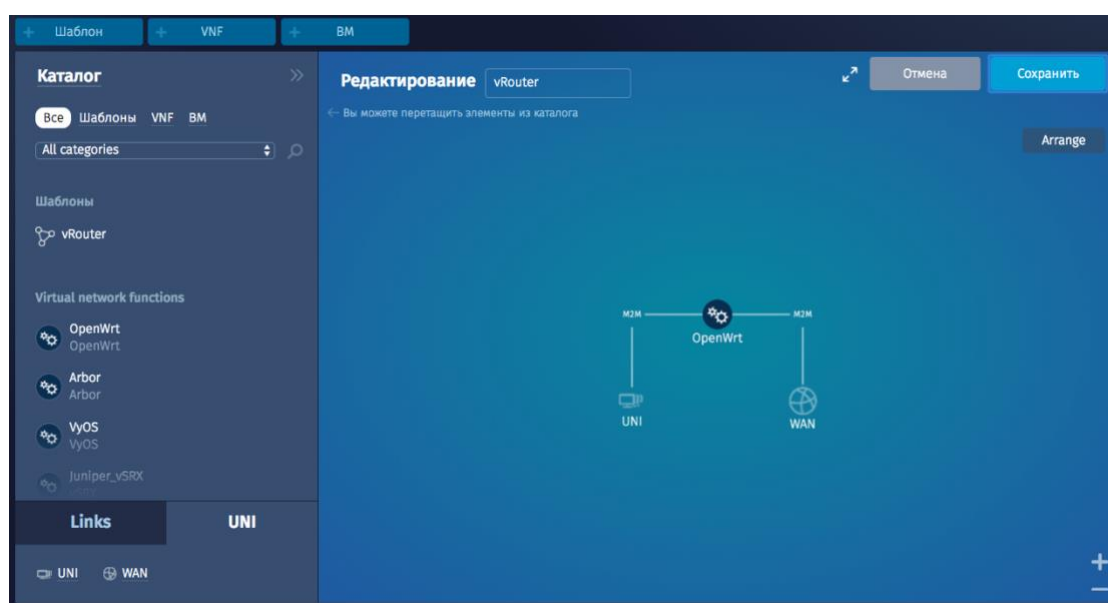


Рис. 7.1 Каталог платформы.

Для выполнения последующих действия необходим пользователь, обладающий правами администратора платформы.

Компоненты, добавленные на платформу, в дальнейшем, Арендаторы могут использовать при развёртывании сервисов.

### 7.1. Наполнение каталога VNF

Для добавления VNF в каталог платформы необходимо:

- Получить образ VNF или установочный файл (.iso) от производителя;
- Ознакомиться с документацией по развёртыванию образа от производителя;
- Подготовить образ виртуальной машины OpenStack с выделением необходимых ресурсов для работы в соответствии с документацией производителя;
- Создать конфигурационные файлы VNF;
- Подготовить файл описания VND (VNF descriptor);

Из подготовленных элементов формируется архивный файл, называемый VNF package который помещается в репозиторий VNF (доступный с платформы ресурс, содержащий VNF package готовые к развёртыванию на виртуальной платформе).

Все дальнейшее взаимодействие происходит в разделе Каталог платформы. Используя элемент меню VNF в разделе Каталог, пользователь импортирует VNF в каталог. Выберите элемент «Сохранить» для внесения VNF в каталог. При успешном импорте VNF появляется в каталоге.

При выборе VNF доступны к просмотру и скачиванию:

- Описание VNF полученное от производителя и добавленное в VNF package;
- Файл VNF descriptor.

## 7.2. Наполнение каталога шаблонами сервисов

Шаблоны сервисов необходимы для автоматизации применения сервисов на SDN сети.

Для создания шаблонов сервисов на платформе используется визуальный конструктор. Созданный администратором шаблон сервиса сохраняется в каталоге платформы с отображением его наименования.

Автоматически добавляются следующие атрибуты:

- дата создания;
- автор (кем создан).

Формирование шаблона происходит путём перетаскивания компонент в окно визуального конструктора и соединение компонент с помощью линков. Виртуальный конструктор запускается выбором элемента «Шаблон». Привязка компонентов внутри шаблона к предустановленным на платформе компонентам производится на этапе развёртывания сервиса арендатором. Элементы сервиса размещаются надлежащим образом на логической топологии по требованию пользователя (ручной режим). Компонент «Arrange» помогают разместить компоненты автоматически.

Для создания шаблона сервиса доступны следующие компоненты:

- Сетевые функции (VNF) из каталога;
- Сервисные интерфейсы (SI) разных типов:
  - Точка подключения сервиса к устройству заказчика (UNI);
  - Точка подключения к сети интернет (WAN);
- Различные типы связей (линков)
  - P2P – E-line;
  - M2M – E-LAN;
  - P2M – E-Tree.

Конфигурации ограничены так, чтобы невозможно было построить неработоспособную топологию (например, если у сервиса есть только подключение P2P, то другой тип соединения поставить невозможно). В процессе создания топологии отображаются подсказки и справки по созданию сервисов. В отдельных случаях в процессе создания шаблона сервиса, пользователю необходимо указывать дополнительные параметры для компонентов (VNF, линков). Например, комплексная VNF в конструкторе может представлять собой группу из нескольких функций. В этом случае после добавления подобной VNF в конструктор пользователю понадобится выбрать, какие именно функции из группы ему нужны. Набранные компоненты необходимо объединить линками используя элемент создать линк в окне виртуального конструктора. Для создания линка нужно последовательно отметить компоненты, между которыми нужно создать связность.

Для каждого добавленного компонента возможно два действия:

- Настройка;
- Удаление.

## 8. МОНИТОРИНГ

На платформе реализован функционал интеграции со сторонними системами управления. В качестве решения для интеграции мы рекомендуем использовать систему Zabbix.

Платформа поддерживает 2 варианта интеграции:

- С использованием агента, устанавливаемого на хосты с VNF;
- По SNMP протоколу.

Достоинства интеграции со сторонней системой:

- Наличие единой точки мониторинга VNF модулей и платформы B4N;
- Возможность накопления данных комплексного мониторинга;
- Возможность построения отчётов.

Настройка интеграции производится в двух разделах. Настройка из главного меню платформы, раздел Мониторинг. Производится настройка соединения с центральным сервером.

Указываются следующие параметры:

- Тип системы мониторинга (Zabbix);
- URL API Zabbix сервера;
- Параметры учётной записи на Zabbix сервер:
  - Логин;
  - Пароль.
- URL API графической подсистемы Zabbix сервера.

Для VNF, развёрнутых в дата-центре, для организации связности с Zabbix сервером, устанавливается Zabbix проху. Для организации связности VNF с проху, в разделе Инфраструктура, на экране Дата центр, раздел Системные ресурсы указываются параметры для подключения к Zabbix Proху развёрнутому в дата-центре. Настраиваются имя проху и IP адрес проху.

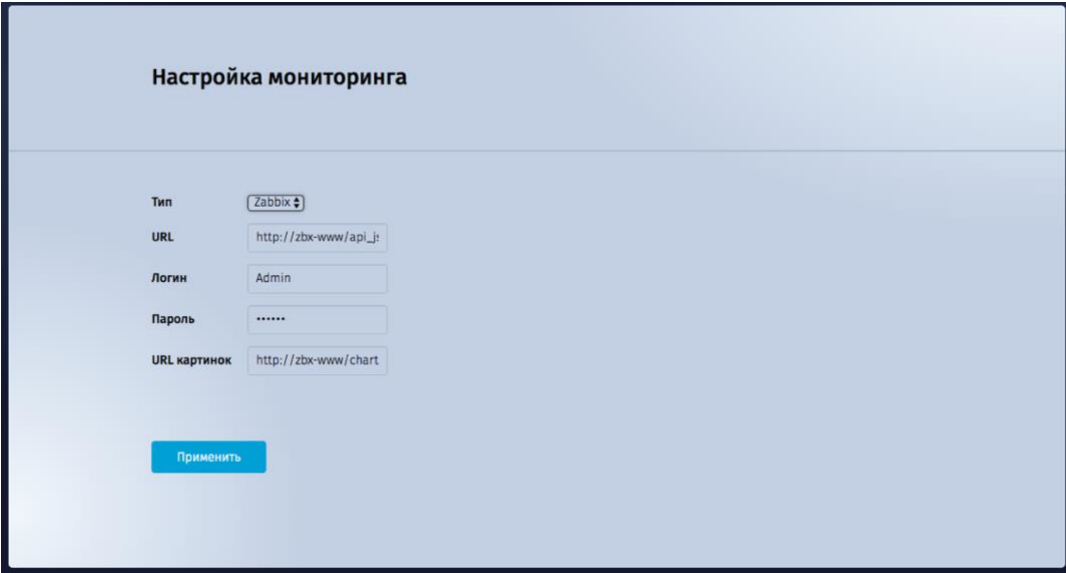


Рис. 8.1 Мониторинг.

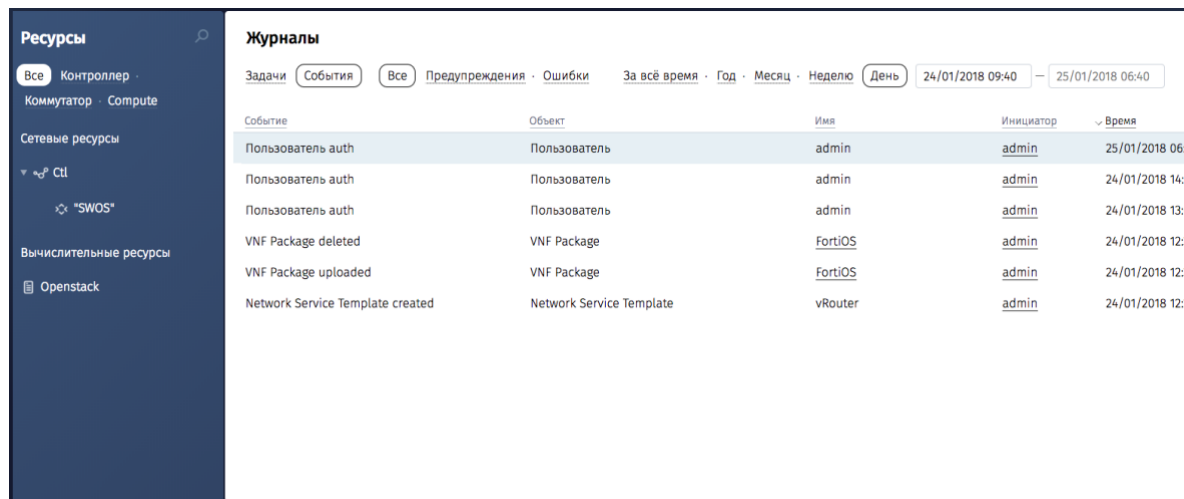
Полная спецификация и описание архитектуры доступно в разделе «Мониторинг VNF» «Руководства по дизайну».

Настройка мониторинга сетевых компонентов и сервисов посредством внешней системы мониторинга описан в разделе 6.2.18 данного руководства.



## 9. ЖУРНАЛ

Раздел «Журнал» служит для отображения log файлов работы платформы.



Событие	Объект	Имя	Инициатор	Время
Пользователь auth	Пользователь	admin	admin	25/01/2018 06:40
Пользователь auth	Пользователь	admin	admin	24/01/2018 14:40
Пользователь auth	Пользователь	admin	admin	24/01/2018 13:01
VNF Package deleted	VNF Package	FortiOS	admin	24/01/2018 12:34
VNF Package uploaded	VNF Package	FortiOS	admin	24/01/2018 12:34
Network Service Template created	Network Service Template	vRouter	admin	24/01/2018 12:23

Рис. 9.1 Журнал событий

Логирование или журналирование событий платформы требуются для фиксации событий, происходящих в модулях системы. Файлы событий формируются автоматически из создаваемых системой файлов. В коде системы прописана отправка сообщения в лог-файл после выполнения определённых действий. Администратор платформы может настроить журналирование необходимых ему действий. На платформе существует выделенная настройка журналирования состояния сетевых ресурсов, доступная из элемента Настройки логирования экрана Конфигурирование сетевых ресурсов экрана Инфраструктура.

Настройка журналирования работы остальных модулей системы проводится из элемента меню Логи меню Администрирование. При настройке журналирования событий пользователь может определить следующие уровни журналирования для каждого из модулей:

- **Error** – регистрация сообщений об ошибке. В лог выводятся особого класса ошибки. Такие ошибки приводят к неработоспособности системы в целом, или неработоспособности одной из подсистем. Чаще всего случаются фатальные ошибки из-за неверной конфигурации или отказов оборудования;
- **Warn** – регистрация предупреждений уровня неожиданных параметров вызова, странных форматов запроса, использование дефолтных значений взамен некорректных – все, что может свидетельствовать о нештатном использовании;
- **Info** – регистрация информационных сообщений разовых операции, которые повторяются крайне редко, но не регулярно. (загрузка кон-фига, плагина, запуск бэкапа);
- **Debug** – регистрация сообщений уровня отладки. Журналирование моментов вызова "крупных" операций Старт/остановка потока, запрос пользователя и т.п.;
- **Trace** – регистрация всех событий. Отмечается на тот случай, если метод Отладка Debug не позволяет локализовать ошибку.

Логи выводятся в едином списке с возможностью фильтрации по объекту, типу или состоянию (предупреждения или ошибки). Логи задач и событий привязаны к каждому объекту системы. Формируемый файл с зарегистрированными событиями можно просмотреть через элемент Посмотреть логи элемента меню Логи меню Администрирование. Пользователь просматривает список логов в системе. Файл с событиями содержит список событий, зафиксированных системой. Каждая строка состоит из следующих элементов:

- Временная отметка;
- Уровень события;
- Название модуля, сгенерировавшего событие;
- Описание события.

## 10. АРЕНДАТОРЫ, ПОЛЬЗОВАТЕЛИ РЕСУРСОВ ПЛАТФОРМЫ

Для пользователей платформы (компаний-клиентов, подразделении компании и т.д.) работающих с выделенными ресурсами необходимо, в рамках платформы, создание логической сущности «Арендатор» и привязки доступных им ресурсов для самостоятельного управления ими.

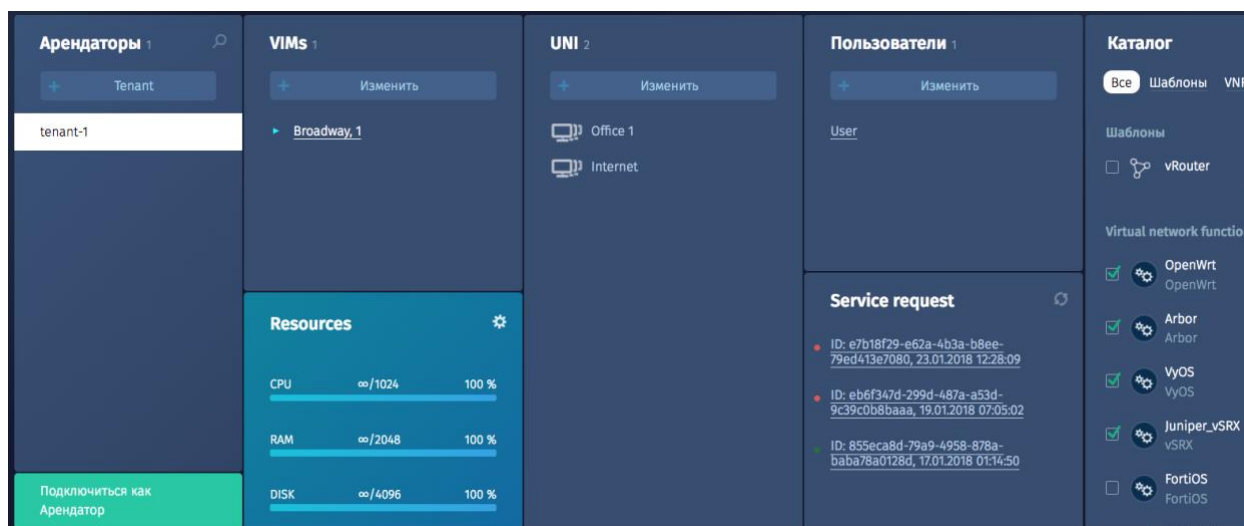


Рис. 10.1 Арендаторы

Перед созданием арендатора, должны быть созданы следующие сущности и параметры:

- Созданы сервисные интерфейсы (SI) к которым подключены пользователи и устройства сопряжения с внешними сетями;
- Предварительно созданы пользователи с соответствующими правами доступа.

Настройка подробно описана в соответствующих разделах текущего документа.

Создание сущности и назначение ресурсов происходит через элемент главного меню Арендаторы. Необходимо настроить следующие параметры:

- Арендатор:
  - Наименование в рамках платформы;
  - Выбрать и добавить VIM, предопределённые на платформе;
  - Выбрать и добавить пользователей платформы в соответствии с предопределёнными ролями.
- Ресурсы доступные арендатору:
  - CPE ядер (шт.);
  - Объем памяти (МБ);
  - Объем дискового пространства (Мб).
- UNI – точка подключения пользователя к сети;
- Определить параметры UNI:
  - Название;
  - Домен, к которому прикреплен арендатор;
  - Дата-центр, к которому прикреплен арендатор;
  - Контроллер-кластер контроллеров под управлением которого находится SDN сеть к которой подключен арендатор;
  - Коммутатор SDN сети;

- Порт коммутатора;
- Тип инкапсуляции порта (Access, VLAN, QinQ);
- Номер VLAN/внешнего VLAN tag (опционально);
- Номер внутреннего VLAN (опционально);
- Выбрать создать UNI.
- Определить из присутствующих в каталоге платформы Шаблонов и VNF доступные арендатору.

Одному арендатору может принадлежать несколько UNI. Арендатор может быть прикреплен к нескольким дата-центрам и VIM в зависимости от необходимых ресурсов и географического расположения точек подключения арендатора. Определенные на данном этапе привязки определяют возможности арендатора на самостоятельное создание сервисов на базе сети SDN и доступных ему VNF. Для подключения к платформе арендатор использует определенные ему учетные записи и привязанные к ним параметры входа (login и password).

## 10.1. Развёртывание логической топологии

Пользователь сущности «Арендатор» с правами «Tenant administrator», используя созданные шаблоны сервисов или отдельные компоненты разворачивает логическую топологию, когда все необходимые параметры уже настроены в конфигураторе.

В процессе развёртывания конфигурации пользователь может:

- Назначить имя сервиса, актуальное в рамках логической топологии арендатора;
- Откорректировать параметры существующего шаблона;
- Изменить параметры компонентов, заданные по умолчанию на требуемые;
- Добавить либо удалить компоненты.

Для запуска сервиса на сети необходимо активировать элемент Развёртывание. В процессе развёртывания на сети будут в соответствии с заданными параметрами созданы линки указанного типа, обеспечивающие соединение компонентов заказчика с VNF компонентом. Развертывание может занимать от нескольких секунд до десятков минут. Для развёрнутого на топологии арендатора сервиса возможны следующие действия:

- Редактировать сервис;
- Удалить сервис.

Пользователь с правами администратор платформы при настройке параметров окружения арендатора может эмулировать подключение под учетной записью арендатора с правами «Tenant administrator» для создания и развёртывания логической топологии от имени арендатора.

Подробное описание процесса в разделе «Каталог» текущего документа.

## 11. РЕЗЕРВНАЯ КОПИЯ. ПОИСК НЕИСПРАВНОСТЕЙ

### 11.1. Создание резервной копии и восстановление (backup & restoring)

Для двух ключевых составляющих элементов платформы, Оркестратора и Контроллера, крайне рекомендуется к регулярному использованию и заложен в платформе функционал создания резервной копии и восстановления данных. Под данными понимается данные по топологии и настройке компонентов инфраструктуры.

#### 11.1.1. *Использование Web*

Резервирование и восстановление Контроллера можно сделать из главного меню раздела Инфраструктура элемента Сетевые ресурсы меню Управление команда – «Резервировать» или «Восстановить». Файл с расширением \*.json содержащий информацию о сетевых сервисах будет создан на локальном хосте с которого осуществлён доступ на платформу.

При восстановлении конфигурации будет предложено выбрать backup файл.

#### 11.1.2. *Использование интерфейса командной строки*

Резервирование и восстановление Оркестратора можно сделать через копирование базы данных оркестратора. База данных содержит информацию и параметры настройки всех компонентов платформы (домены, дата-центры, VIM, VNF и арендаторы). Благодаря используемой архитектуре платформы на основе контейнерной среды виртуализации Docker, база данных оркестратора хранится в docker контейнере с названием ORC-DBS. Для создания резервной копии оркестратора необходимо сделать резервную копию (export) контейнера.

Для экспорта контейнера используйте следующую команду:

```
docker export orc-dbs > /home/orc-dbs.tar
```

имя файла архива должно совпадать с именем контейнера. Каталог /home дан для примера, вы можете использовать любой необходимый.

Для восстановления контейнера используйте следующую команду:

```
docker import - /home/orc_dbs.tar
```

Для восстановления контейнеров или обновление версии компонентов, после авторизации на репозитории brain4net на docker hub, используйте следующую команду:

```
docker run [OPTION] IMAGE [COMMAND] [ARG...]
```

Подробные команды см. в руководстве по установке соответствующего компонента платформы.

Подробное описание из каких компонентов состоит платформа и список контейнеров см. В руководстве по дизайну. Краткое изложение состава в разделе «Работа с системой» данного руководства.

## 11.2. Поиск неисправностей (troubleshooting)

На платформе B4N Service Platform, в текущей реализации, возможно использовать 3 механизма поиска неисправностей:

- Offline механизм;
- Использование сторонних систем управления;
- Использование интерфейса командной строки.

### 11.2.1. *Offline механизм*

Основан на анализе файлов журналирования (log) и статистики предоставляемой платформой и системой виртуализации Docker. Для каждого из ключевых элементов платформы и для каждого из сервисов, в зависимости от настроек системы, собирается статистика и список событий разного уровня. Настройка режима сбора информации и работа с информацией прописана в соответствующих разделах текущего документа.

### 11.2.2. *Использование сторонних систем управления*

Для отслеживания состояния VNF, сбора и анализа информации в платформе предусмотрена возможность использования сторонней системы мониторинга. В текущей реализации настроена интеграция с системой Zabbix v.3.2. Подробности по настройке связности см. раздел Мониторинг текущего документа. Система собирает информацию о количественных показателях работы и доступности сервиса VNF на базе популярных платформ Windows, Linux, Solaris. Собранная информация передаётся Оркестратору платформы для информирования о текущем состоянии VNF.

### 11.2.3. *Использование интерфейса командной строки*

Используемая для работы B4N Service Platform система контейнеризации Docker имеет расширенные инструменты для troubleshooting. При работе платформы организуются контейнеры для Контроллера, Оркестратора их компонентов. Также, система Docker может использоваться для развёртывания виртуального коммутатора B4N SwitchOS. Управление осуществляется через интерфейс командной строки хоста, на котором развёрнуты указанные выше компоненты.

Состав существующих в системе контейнеров можно проверить, используя команду:

```
docker ps -a
```

Каждый контейнер в составе платформы ведёт свой файл журналирования (Log). Просмотр возможен по команде:

```
docker logs [OPTION] CONTAINER
```

Например, для отображения информации log файла, введите команду:

```
docker logs CONTAINER
```

Для отображения детальной информации, введите команду:

```
docker logs --details CONTAINER
```

Для отслеживания информации on-line, введите команду:

```
docker logs --follow CONTAINER
```

Контейнер может быть остановлен, запущен или перезапущен с использованием следующих команд:

```
docker stop CONTAINER  
docker start CONTAINER  
docker restart CONTAINER
```

Список контейнеров, используемых в решении можно проверить в разделе «Референсная архитектура». Подробный список команд и опций для работы с контейнерами системы Docker можно получить на сайте <https://docs.docker.com/docker-for-mac/>. При работе с контейнерами работающей платформы следует быть уверенным в действиях, которые запускаете, во избежание нарушения работы платформы.