Программное обеспечение

«Servicepipe Application Delivery Controller»

Описание функциональных характеристик программного обеспечения

Москва - 2026 г.

Содержание

1. О документе

Настоящий документ содержит описание функциональных характеристик программного обеспечения «Servicepipe Application Delivery Controller» (далее - ПО «Servicepipe ADC», ADC, ПО).

В документе используются следующие термины и определения:

Термин / Сокращение	Определение
ADC	Application Delivery Controller - программный контроллер доставки приложений, который принимает L4/L7-трафик, применяет политики маршрутизации, TLS, балансировки, доступности и обработки запросов, затем передает трафик в backend.
Application delivery	Верхнеуровневый раздел панели управления Servicepipe для настройки функций ADC: статистики, трафика, идентификации, безопасности и конфигурации.
Backend-узел	Сервер приложения, API или внутреннего сервиса, на который ADC направляет трафик после выбора маршрута и Upstream.
Consumer / Потребитель	Идентификатор клиента API или приложения. Используется для применения политик аутентификации, авторизации, лимитов, логирования и других плагинов.
Consumer Group / Группа потребителей	Группа потребителей с общими настройками плагинов и политик обработки трафика.
Global Rule / Глобальное правило	Конфигурация плагинов, которая применяется глобально к трафику ADC независимо от конкретного маршрута или сервиса.
Health Checks / Проверки здоровья	Механизм проверки доступности backend-узлов. Активные проверки выполняются служебными запросами ADC, пассивные анализируют реальные ответы backend.
HTTP(S)	Протокол прикладного уровня для передачи запросов и ответов. HTTPS использует TLS для защиты соединения.
Keepalive Pool	Пул повторно используемых соединений ADC с backend-узлами, который снижает накладные расходы на установку соединений.
L4	Транспортный уровень обработки трафика. ADC работает с TCP, UDP и TLS-соединениями без разбора HTTP-запроса.
L7	Прикладной уровень обработки трафика. ADC анализирует HTTP-запрос и может выбирать маршрут по URI, Host, HTTP-методу, IP-адресу, заголовкам, cookie, query-параметрам и другим условиям.
Node / Нода	Конкретный backend-адрес внутри Upstream: host, port, weight и priority.
Plugin / Плагин	Модуль расширения обработки трафика: аутентификация, переписывание запросов, лимиты, логирование, наблюдаемость, интеграции и другие функции.
Plugin Config / Конфигурация плагинов	Переиспользуемый набор плагинов, который можно подключать к маршрутам и потоковым маршрутам по идентификатору.
Plugin Metadata / Метаданные плагинов	Глобальные настройки конкретного плагина, используемые при его работе.
Proto / Protos	Описание protobuf/gRPC-схемы в формате .proto, которое хранится в ADC и используется в сценариях обработки gRPC/protobuf-трафика.
Route / Маршрут	L7-правило сопоставления HTTP-запросов. Маршрут определяет условия выбора, приоритет, сервис или Upstream, таймауты, плагины и связанные настройки.
Stream Route / Потоковый маршрут	L4-правило сопоставления TCP, UDP или TLS-трафика по адресу сервера, порту, remote address, SNI и связанным настройкам.

Термин / Сокращение	Определение
Service / Сервис	Логическая сущность, объединяющая общие настройки обработки трафика: hosts, WebSocket, script, Upstream, health checks и плагины. Несколько маршрутов могут использовать один сервис.
SNI	Server Name Indication - имя домена, передаваемое клиентом при установлении TLS-соединения. ADC использует SNI для выбора SSL-сертификата и stream-маршрута.
SSL-сертификат	Серверный или клиентский сертификат с приватным ключом и набором допустимых TLS-протоколов.
Secret / Секрет	Ссылка на внешний менеджер секретов или конфигурацию доступа к нему. Используется для хранения чувствительных значений вне обычной конфигурации.
Upstream	Пул backend-узлов и параметры соединения с ними: схема, алгоритм балансировки, Pass Host, повторные попытки, таймауты, keepalive, TLS и health checks.
BGP	Протокол динамической маршрутизации, используемый для обмена маршрутной информацией и анонса адресов сервисов в сетевой инфраструктуре.
RHI	Route Health Injection - механизм управления анонсом VIP-адреса в зависимости от состояния backend-узлов и доступности сервиса.
SNMP	Протокол мониторинга сетевых устройств и серверов. В системном контуре ADC используется для наблюдаемости инфраструктуры.
Пользователь	Представитель Заказчика или администратора, который получает доступ к функциям ADC через панель управления, API и разрешенные административные интерфейсы.

2. О программном обеспечении

Программное обеспечение «Servicepipe Application Delivery Controller» предназначено для доставки приложений и API в on-premises-контуре Заказчика. ADC размещается перед backend-сервисами, принимает входящий трафик, сопоставляет его с правилами обработки, применяет политики TLS, маршрутизации, балансировки, доступности, логирования и обработки HTTP-трафика, а затем передает запрос или соединение в подходящий Upstream.

ПО используется как единая точка управления публикацией HTTP(S), gRPC, TCP, TLS и UDP-сервисов. В прикладных сценариях ADC позволяет отделить внешний трафик от backend-инфраструктуры, централизовать TLS-терминацию, управлять маршрутизацией запросов, выполнять health checks, ограничивать нагрузку на backend и подключать плагины обработки трафика без изменения кода приложений.

Управление функциями ADC выполняется через панель управления в разделе Application delivery. Раздел включает статистику, управление трафиком, идентификацией, безопасностью и конфигурацией gateway-уровня. Сущности интерфейса соответствуют модели gateway-уровня: Service, Route, Stream Route, Upstream, Consumer, Consumer Group, SSL, Secret, Global Rule, Plugin Config, Plugin Metadata и Proto.

ПО поставляется для развертывания в инфраструктуре Заказчика. При необходимости ADC может использоваться совместно с другими решениями Servicepipe, расширяющими контур защиты веб-приложений и сетевой инфраструктуры.

- балансировка L4/L7-трафика и публикация приложений через единый входной контур;
- маршрутизация HTTP-запросов по URI, Host, HTTP-методу, IP-адресу клиента и дополнительным условиям;
- управление Upstream, backend-нодами, алгоритмами балансировки, повторными попытками и таймаутами;
- активные и пассивные проверки здоровья backend-узлов;
- управление TLS, SNI, серверными и клиентскими сертификатами;
- подключение плагинов для аутентификации, логирования, переписывания запросов, ограничений и интеграций;
- сбор статистики, логирование, диагностика и интеграция с системами мониторинга.

3. Информация о правообладателе

Параметр	Значение
Полное наименование	Общество с ограниченной ответственностью «Сервиспайп»
Сокращенное наименование	ООО «Сервиспайп»
ИНН	7708257951
КПП	772201001
ОГРН	1157746483784
Юридический адрес	Россия, 111024, г. Москва, ул. Кабельная 2-я, д. 2, стр. 3, эт. 2, помещение XVIII, ком. 8
Телефон	+7 (495) 374-62-65
Адрес электронной почты	welcome@servicepipe.ru
Веб-сайт	servicepipe.ru

4. Функциональные возможности

Состав доступных функций ADC определяется конфигурацией поставки, ролью Пользователя и правами доступа, назначенными администратором. Основные функции доступны через раздел Application delivery панели управления и через программные интерфейсы, используемые системой управления.

4.1 Общие сведения

В панели управления Application delivery реализована единая модель работы с сущностями ADC. Пользователь может просматривать списки объектов, создавать новые объекты, редактировать существующие, удалять неиспользуемые объекты, заполнять формы настройки и применять изменения к gateway-конфигурации.

- табличный просмотр сущностей с колонками ID, имя, описание, схема, дата обновления, состояние и действия;
- создание сущностей через кнопку «+ Добавить» и боковую форму настройки;
- редактирование и удаление объектов из колонки «Действия»;
- использование закрепленных кнопок «Сохранить» и «Отменить» в формах;
- поддержка меток в формате key:value для классификации и поиска конфигураций;
- валидация пользовательского ввода, включая JSON-поля, HTTP-статусы, заголовки, идентификаторы и обязательные поля;
- переиспользование общих сущностей: Upstream, Service, Plugin Config, Consumer Group, SSL и Secret.

4.2 Доступ к статистике и мониторингу Application delivery

ПО поддерживает отображение сводной статистики по трафику ADC. В разделе «Статистика - Обзор» пользователь выбирает период, сервис и маршрут, после чего получает агрегированные метрики и графики по выбранному срезу.

- отображение интенсивности запросов Requests/sec;
- отображение доли ошибочных ответов Error rate;
- отображение 95-го процентиля задержки P95 latency;
- отображение полосы пропускания Bandwidth;
- отображение количества активных соединений Active connections;
- отображение состояния Upstream и признаков проблем в backend-пулах;
- график Request Rate с распределением по классам HTTP-статусов 2xx, 3xx, 4xx, 5xx;
- график Error Rate для анализа клиентских и серверных ошибок;
- графики Latency Percentiles и Latency Breakdown для анализа задержек на уровне request, upstream и ADC.

4.3 Управление сервисами

ПО поддерживает управление сущностями «Сервис». Сервис выступает логическим контейнером для общих настроек обработки трафика и может использоваться несколькими маршрутами. Это позволяет не дублировать однотипные настройки Upstream, health checks, WebSocket, script и плагинов.

- отображение списка сервисов с ID, именем, описанием, временем обновления и доступными действиями;
- создание, редактирование и удаление сервиса;
- задание имени, описания, меток и статуса сервиса;
- включение или отключение поддержки WebSocket;
- задание скрипта обработки на уровне сервиса;

- задание списка Host, для которых применяется сервис;
- привязка к существующему Upstream по ID или создание inline-конфигурации Upstream внутри сервиса;
- настройка health checks для backend-узлов, связанных с сервисом;
- подключение плагинов, применяемых на уровне сервиса.

4.4 Управление HTTP/L7-маршрутами

ПО поддерживает управление сущностями «Маршрут». Маршрут определяет, какие HTTP-запросы должны быть обработаны конкретной конфигурацией и куда они должны быть направлены. Для сопоставления запроса используются URI, Host, HTTP-методы, IP-адрес клиента и дополнительные условия.

- отображение списка маршрутов с ID, именем, URI, описанием, временем обновления и действиями;
- создание, просмотр, редактирование и удаление маршрута;
- задание имени, описания, меток, приоритета и статуса маршрута;
- выбор одного или нескольких HTTP-методов: GET, POST, PUT, DELETE, PATCH, HEAD, OPTIONS, CONNECT, TRACE, PURGE;
- настройка URI и множественных URI для сопоставления запросов;
- настройка Host и множественных Host;
- настройка удаленного адреса и списка удаленных адресов клиента;
- задание Vars (JSON) и функции фильтрации для дополнительных условий маршрутизации;
- включение WebSocket для маршрута;
- привязка маршрута к существующему сервису или существующему Upstream;
- настройка таймаутов Connect, Send и Read;
- подключение ID конфигурации плагинов и локальных плагинов маршрута;
- использование health checks и inline-настроек Upstream, если маршрут не привязан к общей сущности.

4.5 Управление потоковыми маршрутами

ПО поддерживает управление сущностями «Потоковый маршрут» для L4/stream-сценариев. Потоковые маршруты применяются к TCP, TLS, UDP и другим потоковым протоколам, когда ADC должен принимать соединение на заданном адресе и порту и передавать его в нужный Upstream без разбора HTTP-запроса.

- отображение списка потоковых маршрутов с ID, адресом сервера, портом сервера, описанием, временем обновления и действиями;
- создание, редактирование и удаление потокового маршрута;
- задание описания и меток потокового маршрута;
- настройка адреса сервера, порта сервера, удаленного адреса и SNI;
- привязка к сервису по ID или к Upstream по ID;
- создание inline-конфигурации Upstream для потокового маршрута;
- подключение ID конфигурации плагинов и stream-плагинов;
- настройка блока «Протокол»: имя протокола, Superior ID, конфигурация JSON и логгер JSON.

4.6 Управление Upstream и backend-нодами

ПО поддерживает управление сущностями Upstream. Upstream определяет пул backend-нод и правила соединения с ними. Upstream может использоваться самостоятельно, подключаться к сервису, маршруту или потоковому маршруту.

- отображение списка Upstream с ID, именем, схемой, временем обновления и действиями;

- создание, редактирование и удаление Upstream;
- задание имени, описания и меток;
- выбор источника Upstream: статические ноды или Service Discovery;
- добавление backend-нод с параметрами Хост, Порт, Вес и Приоритет;
- задание имени сервиса, типа обнаружения и аргументов JSON для Service Discovery;
- выбор схемы соединения с backend: HTTP, HTTPS и другие поддерживаемые схемы;
- выбор алгоритма балансировки нагрузки: Round Robin, CHash, Least Conn, EWMA, Weighted Source Hashing;
- настройка Hash On и ключа для алгоритмов с хэшированием;
- настройка Pass Host и Host Upstream;
- включение повторных попыток, количества повторов и таймаута повторов;
- настройка таймаутов Connect, Send и Read;
- настройка Keepalive Pool: размер пула, idle timeout и число запросов;
- настройка TLS-проверки до upstream и клиентского сертификата.

4.7 Управление проверками здоровья и отказоустойчивостью

ПО поддерживает активные и пассивные проверки здоровья backend-узлов. Проверки позволяют исключать недоступные ноды из балансировки и возвращать их в работу после восстановления.

- включение общего блока Health Checks;
- включение активных проверок, при которых ADC самостоятельно выполняет служебные проверки backend;
- выбор типа активной проверки: HTTP, HTTPS или TCP;
- настройка таймаута, параллельности, хоста, порта и HTTP-пути активной проверки;
- включение проверки HTTPS-сертификата для активной проверки;
- задание дополнительных заголовков запроса в формате Header-Name: Header Value;
- настройка критериев Healthy: интервал, число успешных проверок и HTTP-статусы;
- настройка критериев Unhealthy: интервал, HTTP-ошибки, TCP-ошибки, таймауты и HTTP-статусы;
- включение пассивных проверок, при которых ADC анализирует реальные ответы backend на пользовательский трафик;
- использование пассивных и активных проверок совместно для быстрого обнаружения отказа и автоматического возврата ноды в работу.

4.8 Управление идентификацией клиентов API

ПО поддерживает управление потребителями и группами потребителей. Эти сущности используются для идентификации клиентов API, применения аутентификации, авторизации, лимитов и других политик через плагины.

- отображение списка потребителей с именем пользователя, описанием, ID группы, временем обновления и действиями;
- создание, редактирование и удаление потребителя;
- задание уникального имени пользователя, описания, меток и ID группы потребителей;
- подключение плагинов на уровне потребителя;
- отображение списка групп потребителей с ID, описанием, временем обновления и действиями;
- создание, редактирование и удаление группы потребителей;
- задание ID, описания и меток группы;
- подключение общих плагинов на уровне группы потребителей.

4.9 Управление TLS, SSL-сертификатами и секретами

ПО поддерживает централизованное управление TLS-параметрами. Пользователь может добавлять SSL-сертификаты, связывать их с SNI, выбирать допустимые TLS-протоколы и хранить чувствительные параметры через менеджеры секретов.

- отображение списка SSL-сертификатов с ID, SNI, типом, статусом, временем обновления и действиями;
- создание, редактирование и удаление сертификата;
- задание описания, меток, типа сертификата и статуса;
- выбор типа сертификата: серверный или клиентский;
- задание TLS-протоколов, например TLSv1.2 и TLSv1.3;
- задание одиночного SNI и списка SNI;
- ввод сертификата PEM и приватного ключа PEM;
- использование сертификата для HTTPS-сценариев, TLS-терминации, TLS bridging, TLS до upstream и mTLS-сценариев;
- управление секретами и конфигурациями менеджеров секретов;
- поддержка HashiCorp Vault, AWS Secrets Manager и Google Cloud Secret Manager;
- задание URI, префикса, токена, namespace, access key, secret access key, region, endpoint URL, параметров SSL-проверки и сервисных учетных данных в зависимости от выбранного менеджера.

4.10 Управление плагинами, глобальными правилами и метаданными

ПО поддерживает расширение обработки трафика с помощью плагинов. Плагины могут подключаться к сервисам, маршрутам, потоковым маршрутам, потребителям, группам потребителей, глобальным правилам и переиспользуемым конфигурациям плагинов.

- добавление плагина через окно выбора плагинов и заполнение его конфигурации в JSON;
- настройка локальных плагинов на уровне Service, Route, Stream Route, Consumer и Consumer Group;
- создание глобальных правил, выполняемых для всего трафика ADC;
- создание конфигураций плагинов с ID, именем, описанием, метками и набором плагинов;
- подключение Plugin Config к маршрутам и потоковым маршрутам по ID;
- создание и редактирование метаданных плагина по имени плагина;
- настройка конфигурации метаданных в JSON;
- поддержка сценариев аутентификации, авторизации, rate limiting, переписывания запросов, редиректов, модификации ответов, логирования, observability и интеграций;
- переиспользование типовых наборов плагинов для снижения дублирования конфигурации.

4.11 Управление proto-файлами, gRPC и протоколами

ПО поддерживает хранение proto-описаний и настройку протокольных параметров в потоковых маршрутах. Это используется в сценариях gRPC, protobuf и специализированной stream-обработки.

- отображение списка Protos с ID, временем обновления и действиями;
- создание, редактирование и удаление proto-файла;
- задание ID proto-файла и содержимого .proto;
- использование proto-описаний в сценариях, где требуется знание protobuf/gRPC-схемы;
- настройка имени протокола, Superior ID, конфигурации JSON и логгера JSON для потокового маршрута;
- поддержка передачи gRPC-трафика через маршруты и Upstream при соответствующей конфигурации схемы и плагинов.

4.12 Обработка HTTP-трафика, лимиты и политики

ПО поддерживает программируемую обработку HTTP-запросов и ответов на уровне маршрутов, сервисов и плагинов. ADC может выполнять преобразования без изменения backend-приложений.

- переписывание URI, HTTP-метода, Host и заголовков запроса перед передачей в upstream;
- добавление, удаление и перезапись HTTP-заголовков запроса и ответа;
- возврат редиректов с заданными HTTP-статусами и Location;
- изменение тела ответа или запроса, в том числе по шаблону или регулярному выражению;
- описание условной логики обработки через набор условий по методу, URI, заголовкам, cookie, query-параметрам и другим признакам;
- включение HSTS-политики для HTTPS-сервисов;
- ограничение нагрузки на backend по частоте запросов, числу запросов за период и числу одновременных соединений;
- ограничение количества HTTP-заголовков, размера request line и размера отдельных заголовков;
- использование Consumer, Route, IP-адреса и других признаков в качестве ключа лимитирования.

4.13 Логирование, диагностика и наблюдаемость

ПО поддерживает отправку событий во внешние системы логирования и предоставляет средства диагностики трафика и инфраструктуры.

- настройка логирования в Syslog через соответствующий плагин;
- настройка логирования в Kafka через соответствующий плагин;
- выбор и расширение состава полей логов: метод, URI, статус ответа, IP клиента, upstream, route ID, задержки и другие параметры;
- включение логирования тела запроса или ответа с учетом ограничений по размеру и безопасности;
- использование SNMP для мониторинга системного состояния узлов;
- использование tcpdump для захвата пакетов и анализа сетевых проблем;
- использование инструментов анализа TLS-трафика и PCAP-файлов для диагностики HTTPS/HTTP2-сценариев;
- использование метрик статистики Application delivery для оперативного контроля ошибок, задержек, полосы пропускания и состояния Upstream.

4.14 Сетевые режимы, маршрутизация и RHI

ПО поддерживает эксплуатационные сетевые функции, необходимые для размещения ADC в инфраструктуре Заказчика и обеспечения отказоустойчивого приема трафика.

- работа в режимах Active/Active и Active/Standby для отказоустойчивой обработки трафика;
- поддержка статической маршрутизации на уровне операционной системы;
- поддержка динамической маршрутизации BGP через BIRD;
- анонс сервисных адресов и управление маршрутами в сетевой инфраструктуре;
- использование RHI для снятия или возврата анонса VIP в зависимости от состояния upstream;
- поддержка настройки сетевых интерфейсов, маршрутов, системных служб и TCP-параметров;
- синхронизация времени через NTP/chrony для корректной работы TLS, журналов и корреляции событий;
- администрирование через GUI, API, SSH и, для физических установок, IPMI/BMC.

5. Технологический стек

ПО «Servicepipe ADC» включает интерфейс управления, gateway-уровень обработки трафика, API-взаимодействие с системой управления и инфраструктурные сервисы узла ADC.

5.1 Архитектура и компоненты

- Панель управления Application delivery - веб-интерфейс для настройки сущностей ADC и просмотра статистики.
- Gateway-уровень - компонент приема, маршрутизации, балансировки и обработки L4/L7-трафика.
- Система управления конфигурацией - программный интерфейс и backend-сервисы, через которые панель управления создает, изменяет и удаляет сущности ADC.
- Инфраструктурные сервисы - сетевые, временные, мониторинговые и диагностические службы узлов ADC.
- Внешние интеграции - Syslog, Kafka, менеджеры секретов, системы мониторинга и сетевое оборудование Заказчика.

5.2 Технологии интерфейса управления

Компонент	Технологии
Языки и платформа	TypeScript, JavaScript, Node.js 20.19
Фреймворк пользовательского интерфейса	React 19, React DOM, React Router
Сборка	Vite 7, Yarn 1.22
Работа с формами и валидацией	react-hook-form, zod
API и данные	Apollo Client, GraphQL Codegen, axios, TanStack React Query
Графики и визуализация	Chart.js, react-chartjs-2, ECharts, react-google-charts
Локализация	i18next, react-i18next, i18next-browser-languagedetector
Контейнеризация и публикация frontend	Docker, nginx web server
Контроль качества	ESLint, Prettier, Vitest, Cypress, Storybook

5.3 Технологии gateway-уровня и инфраструктурные сервисы

Компонент	Назначение
Apache APISIX / NGINX / OpenResty, Lua	Шлюзовая модель обработки L4/L7-трафика, маршрутов, сервисов, upstream и плагинов.
Linux	Операционная среда узлов ADC, настройка сетевых интерфейсов, маршрутов, системных служб и диагностики.
BIRD	Динамическая маршрутизация BGP и управление анонсами сервисных адресов.
chrony	Синхронизация системного времени по NTP.
Net-SNMP	Мониторинг системного состояния и интеграция с внешними системами мониторинга.
Syslog	Передача событий и журналов во внешние системы логирования.
Kafka	Потоковая доставка логов и событий в инфраструктуру обработки данных.
tcpdump, tshark, Wireshark-compatible PCAP	Диагностика сетевого трафика и анализ инцидентов.
HashiCorp Vault, AWS Secrets Manager, Google Cloud Secret Manager	Хранение и получение чувствительных значений через менеджеры секретов.

5.4 Системные требования и среда исполнения

- Среда исполнения frontend-модуля - nginx web server, обслуживающий статические артефакты веб-приложения.
- Среда сборки frontend-модуля - Node.js 20.19 и Yarn 1.22.
- Среда исполнения узлов ADC - Linux-совместимая серверная операционная система в физическом или виртуальном on-premises-контуре.
- Доступ к интерфейсу управления предоставляется через веб-браузер Пользователя.
- Взаимодействие между интерфейсом управления и сервисами управления выполняется через программные интерфейсы API.
- Объем исходного кода фронтального модуля Application delivery в предоставленном репозитории составляет 7,83 МБ без внешних зависимостей и бинарных артефактов. Общий объем исходного кода серверных компонентов определяется комплектом поставки ADC.

Программное обеспечение представляет собой инструмент управления и эксплуатации ADC, реализованный на современном технологическом стеке. ПО позволяет включать и изменять параметры доставки приложений, управлять политиками трафика, контролировать состояние backend-узлов и выполнять диагностику работы сервисов в инфраструктуре Заказчика.