# Описание функциональных характеристик

# Компонента Сервисный прокси (SVPX)

# Продукта Platform V Synapse Service Mesh (SSM)

# ОГЛАВЛЕНИЕ

Описание функциональных характеристик ..... 3
Термины и определения ..... 3
Назначение ..... 4
Цель создания ..... 4
Концептуальная модель предметной области ..... 4
Основные функции (сущностные) ..... 6
Варианты использования ..... 7
Сценарии использования ..... 7

# Описание функциональных характеристик

## Термины и определения

| Термин/аббревиатура | Определение |
|---|---|
| API | Application Programming Interface, программный интерфейс приложения |
| gRPC | Высокопроизводительный фреймворк, разработанный компанией Google для вызова удаленных процедур (RPC) |
| mTLS | Mutual TLS, протокол взаимной TLS-аутентификации |
| TLS | Transport Layer Security, протокол защиты транспортного уровня |
| Pod/под | Набор контейнеров внутри узла кластера Kubernetes |
| Istio SE | Настраиваемая сервисная сетка с открытым исходным кодом, служащая для взаимодействия, мониторинга и обеспечения безопасности контейнеров в кластере Kubernetes |
| Контрольная панель | Проект, где запущены управляющие приложения Synapse Service Mesh (компонент POLM) |
| Управление политиками / POLM | Компонент Управление политиками из состава продукта Platform V Synapse Service Mesh |
| Platform V Synapse Service Mesh / SSM | Программный продукт на базе Istio SE, обеспечивающий возможность создания сервисной сети поверх Платформенной в |

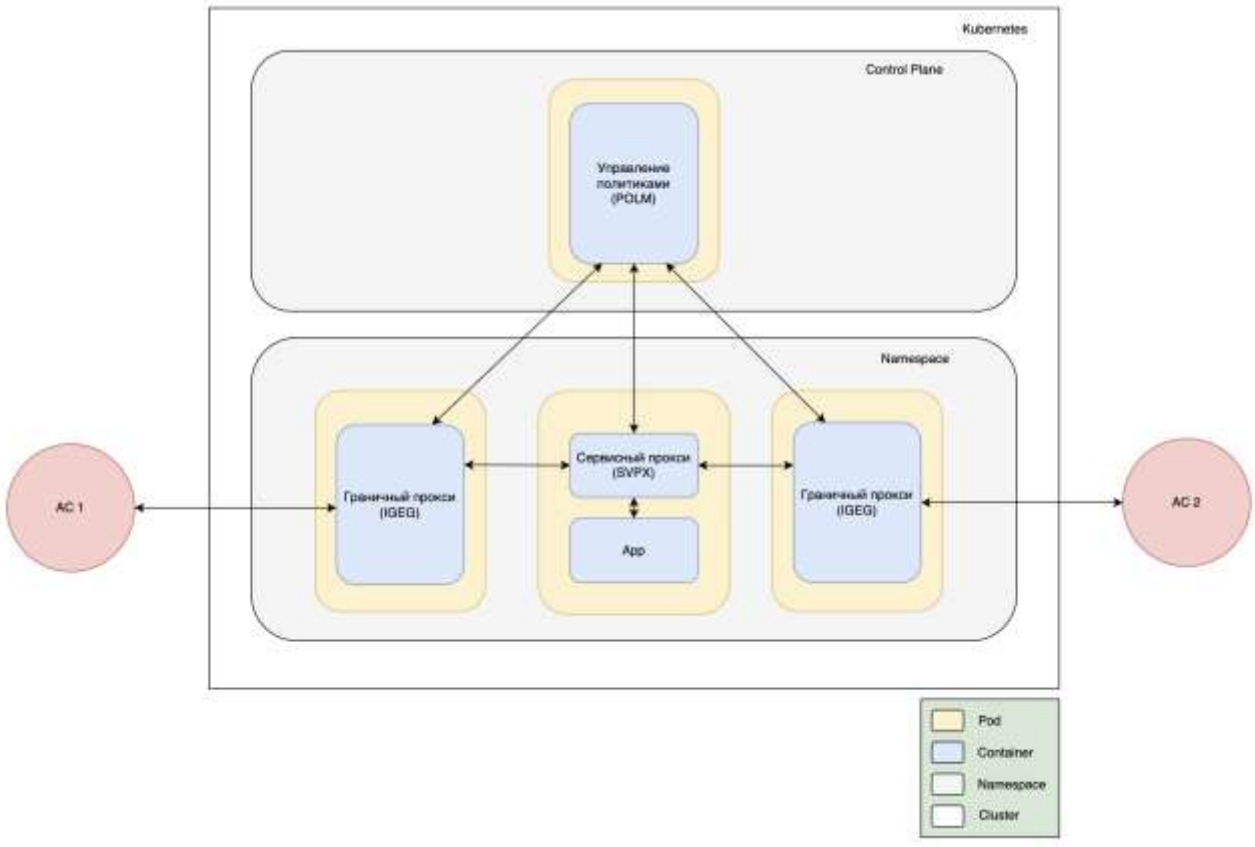
| Термин/аббревиатура | Определение |
|---|---|
| | Kubernetes |
| Граничный прокси / IGEG | Компонент Граничный прокси Platform V Synapse Service Mesh |
| Сервисный прокси / SVPX | Компонент Сервисный прокси Platform V Synapse Service Mesh |

# Назначение

## Цель создания

Компонент Сервисный прокси из состава продукта Platform V Synapse Service Mesh (далее Synapse) предназначен для предоставления базовых интеграционных операций прикладной части интеграционного сервиса. Сервисный прокси используется для маршрутизации и обеспечения безопасности трафика между приложениями.

## Концептуальная модель предметной области



Компонент сервисный прокси (SVPX) перехватывает весь сетевой трафик и применяет к нему набор правил. В каждый Pod добавляется «умный прокси» в виде sidecar-контейнера.

Service mesh (сервисная сетка) — это слой инфраструктуры, который позволяет управлять связью между микросервисами приложения. Service mesh состоит из двух зон: Control Plane и Data Plane.

Компоненты Граничный и сервисный прокси образуют собой Data Plane и могут динамически настраиваться с помощью Control Plane.

Data Plane реализуется с помощью контейнеров разворачиваемых в режиме sidecar контейнеров-прокси. Прокси контролируют и выступают посредниками в сетевых взаимодействиях между микросервисами.

Функции Data Plane:

- применение политик маршрутизации и балансировки трафика;
- управление повторами и тайм-аутами, определение «мертвых» узлов («circuit breaking»), управление сбойными ситуациями и обеспечение других механизмов отказоустойчивости сервисов;
- аутентификация и авторизация вызовов;
- формирование метрик.

Control Plane позволяет управлять слоем выделенных sidecar-прокси.

Функции Control Plane:

- назначение и распространение политик маршрутизации и балансировки трафика;
- распространение ключей, сертификатов, токенов;
- сбор телеметрии, формирование метрик мониторинга;
- интеграция с инфраструктурой безопасности и мониторинга;
- безопасное взаимодействие между службами в кластере с помощью проверки подлинности и авторизации на основе удостоверений.

Control Plane выносится в отдельный проект Kubernetes и содержит компонент POLM который:

- рассылает политики для сервисных и граничных прокси.
- отвечает за добавление сервисного прокси к Поду с приложением.
- обеспечивает надежную аутентификацию между сервисами со встроенной идентификацией и управлением учетными данными. Выдает TLS-сертификаты, следит за сроком их истечения и автоматически обновляет.
- является компонентом валидации конфигурации компонента POLM. Отвечает за изоляцию остальных компонентов Istio SE от конфигурации пользователя базовой платформы.

Сервисный прокси позволяет использовать базовые интеграционные операции на своем уровне. В этом случае не потребуется доработка прикладного кода для внесения изменений в интеграционные сценарии, например, для изменения тайм-аута ожидания ответа.

Правила и политики маршрутизации трафика доставляются в каждый сервисный прокси из модуля управления политиками POLM, который располагается в проекте Kubernetes Control Plane.

Процесс отправки метрик вызова основан на API/Framework OpenTracing. Каждый сервисный прокси при отправке запросов прикладного контейнера отправляет метрики вызова в централизованный сервис сбора и анализа метрик.

## Основные функции (сущностные)

| Название функции | Потребитель функции | Аргументы функции | Результат |
|---|---|---|---|
| Определение адресата service-discovery | Приложение в контейнере | - | Управление правилами маршрутизации и унификация процесса определения адресата |
| Динамическая маршрутизация трафика | Приложение в контейнере | - | Маршрутизация и балансировка запросов приложений |
| Балансировки и политики балансировки нагрузки | Приложение в контейнере | - | Балансировка запросов согласно политикам |
| Проверка доступности адресатов | Приложение в контейнере | - | Управление процессом повторной отправки запросов. Маршрутизация и балансировка запросов |
| Управление повторами отправки запросов | Приложение в контейнере | - | Маршрутизация и балансировка запросов. Управление временем повторной отправки запроса |
| Управление временем ожидания ответов | Приложение в контейнере | - | Маршрутизация и балансировка запросов. Управление тайм-аутами |
| «Circuit breaker» и другие механизмы обеспечения отказоустойчивости | Приложение в контейнере | - | Минимизирует влияние сбоя внешнего сервиса на производительность вызывающего сервиса |

| Название функции | Потребитель функции | Аргументы функции | Результат |
|---|---|---|---|
| Возможности обеспечения аутентификации/авторизации вызовов | Приложение в контейнере | - | Аутентификация и авторизация |
| Сбор и отправка метрик телеметрии запросов (open tracing) | Сервис трассировки | - | Формирование отчетов на основании значений метрик |

# Варианты использования

Запуск компонента происходит в среде Kubernetes совместно с бизнес-приложением.

# Сценарии использования

Запуск компонента происходит в среде Kubernetes совместно с бизнес-приложением.