**Пользовательские Политики Балансировки нагрузки**

Объясняет, как пользовательские политики балансировки нагрузки могут помочь оптимизировать балансировку нагрузки в уникальных условиях.

Обзор

Одной из ключевых особенностей gRPC является балансировка нагрузки, которая позволяет распределять запросы от клиентов между несколькими серверами. Это помогает предотвратить перегрузку одного сервера и позволяет масштабировать систему путём добавления новых серверов.

Политика балансировки нагрузки gRPC получает список IP-адресов серверов от преобразователя имён. Политика отвечает за поддержание соединений (подканалов) с серверами и выбор соединения для использования при отправке RPC.

Реализация Вашей Собственной Политики

По умолчанию будет использоваться политика pick\_first . Эта политика фактически не выполняет балансировку нагрузки, а просто пробует каждый адрес, который получает от преобразователя имён, и использует первый адрес, к которому удаётся подключиться. Обновив конфигурацию службы gRPC, вы также можете переключиться на использование round\_robin , которая подключается к каждому полученному адресу и переключается между подключёнными бэкендами для каждого RPC. Доступны и другие политики балансировки нагрузки, но их точный набор зависит от языка. Если встроенные политики не соответствуют вашим потребностям, вы можете реализовать собственную политику.

Это предполагает реализацию интерфейса балансировщика нагрузки на используемом вами языке. На высоком уровне вам потребуется:

* Зарегистрируйте свою реализацию в реестре балансировщика нагрузки, чтобы на неё можно было ссылаться в конфигурации сервиса
* Проанализируйте объект конфигурации JSON вашей реализации. Это позволит вам настроить балансировщик нагрузки в конфигурации сервиса с помощью любого произвольного JSON, который вы решите поддерживать
* Управляйте тем, с какими бэкендами поддерживать связь
* Реализуйте picker функцию, которая будет выбирать, к какому серверу подключаться при выполнении RPC. Обратите внимание, что это должна быть быстрая операция, так как она выполняется на пути вызова RPC
* Чтобы включить ваш балансировщик нагрузки, настройте его в конфигурации вашего сервиса

Конкретные действия зависят от языка. Примеры на вашем языке см. в разделе поддержки языков.

Provides addresses &  
LB configProvides a pickerRequests  
a subchannelManages subchannels  
to backendsCreatesPicks oneName ResolverLoad BalancerChannelPickerSubchannel 1..n

Внутренние показатели

Что, если вашей политике балансировки нагрузки требуется информация о серверной части в режиме реального времени? Для этого вы можете использовать серверные метрики. Вы можете получать серверные метрики либо в составе ответов серверных RPC, либо отдельно в виде отдельных RPC-запросов от серверной части. Предоставляются стандартные метрики, такие как загрузка ЦП и памяти, но вы также можете реализовать собственные метрики.

Для получения дополнительной информации об этом, пожалуйста, ознакомьтесь с руководством по пользовательским внутренним метрикам

Сервисная сетка

Если у вас настроена сервисная сетка, в которой центральная панель управления координирует конфигурацию ваших микросервисов, вы не можете настроить собственный балансировщик нагрузки напрямую через конфигурацию сервиса. Но для этого предусмотрена поддержка протокола xDS, который используется вашей панелью управления для связи с клиентами gRPC. Пожалуйста, обратитесь к документации вашей панели управления, чтобы узнать, как поддерживается настройка собственной балансировки нагрузки.

Для получения более подробной информации, пожалуйста, ознакомьтесь с предложением gRPC A52.

Языковая Поддержка

ЯзыкПримерПримечанияJavaПример JavaВпередПерейти к примеруC++Пока не поддерживается

Последнее изменение: 12 ноября 2024 г.: встраивание видео с YouTube на разные веб-страницы (#1380) (196f408)