

Велиева С.А.

студент магистратуры

2 курс, факультет «Кибербезопасности и управления»

Поволжский Государственный университет телекоммуникаций и информатики

Россия, г. Самара

Научный руководитель: Гавлиевский С.Л., д.т.н.,

Поволжский Государственный университет телекоммуникаций и информатики

Россия, г. Самара

**РАСПРЕДЕЛЕНИЕ И МАСШТАБИРОВАНИЕ НАГРУЗКИ С
ИСПОЛЬЗОВАНИЕМ ПРОТОКОЛА AMQP**

Velieva S.A.

Master's Degree student

2nd year, Faculty of Cybersecurity and Management

Volga State University of Telecommunications and Informatics

Russia, Samara

SHARING AND SCALING LOADER WITH AMQP PROTOCOL

Аннотация: В данной публикации рассматриваются распределение и масштабирование нагрузки с помощью протокола AMQP. Даны определения компонентам брокера сообщений RabbitMQ.

Abstract: The article presents load balancing and scaling using the AMQP protocol. The components of the RabbitMQ message broker are defined.

Ключевые слова: протокол AMQP, брокер сообщений RabbitMQ, распределение и масштабирование нагрузки.

Keywords: protocol AMQP, message broker RabbitMQ, distribution and load scaling.

Протокол AMQP – протокол прикладного уровня, который используется для передачи сообщений между компонентами. Основная идея данного

протокола заключается в том, что есть несколько компонентов одного приложения или несколько независимых приложений, которые обмениваются информацией с помощью сообщений. Для передачи сообщений используется AMQP-брокер или брокер сообщений. С его помощью осуществляется маршрутизация сообщений и гарантия их доставки; подписка на нужные типы сообщений и распределение потоков данных.

Рассмотрим реализацию протокола AMQP на основе брокера сообщений

Сообщение (message) – единица передаваемых данных, которая содержит какую-либо полезную нагрузку;

Обменник или точка обмена (exchange) – точка, в которую происходит отправка сообщения. Точка имеет привязку к одной или нескольким очередям. После получения сообщения обменником, данное сообщение распределяется по очередям, которые имеют привязку к обменнику;

Очередь (queue) – структура данных, которая хранит ссылки на сообщения и распределяет копии этих сообщений по потребителям. При этом каждое сообщение будет назначено определенному потребителю, а не будет продублировано всем сразу;

Привязка (binding) – правило, которое связывает между собой обменники и очереди. Отношение между обменниками и очередями можно описать как многие-ко-многим. То есть у одного обменника может быть несколько очередей, так и одна очередь может быть связана с несколькими обменниками.

Потребитель (consumer) – приложение, которое получает сообщение из очереди и обрабатывает его. Для одной очереди может быть запущено несколько экземпляров одного приложения. Таким образом все сообщения будут распределяться между потребителями [1].

Работу протокола AMQP можно описать следующим образом: некое приложение генерирует сообщение и отправляет его в обменник. Обменник

пересылает сообщения в очереди, которые имеют привязки к этому обменнику. Затем сообщения из очереди распределяются по потребителям, которые занимаются обработкой этих сообщений.

Такой подход позволяет получить ряд преимуществ, одним из которых является возможность гибкой настройки. С помощью протокола AMQP можно легко настроить не только распределение сообщений из обменника в очереди, но и обработку сообщений в этих очередях. Например, можно запустить несколько экземпляров приложений-потребителей на различных устройствах, установить соединение с необходимой очередью и брокер сообщений будет самостоятельно распределять нагрузку между приложениями-потребителями

Таким образом, если один обработчик сообщений не справляется с нагрузкой, мы можем запустить еще несколько экземпляров без какой-либо дополнительной настройки. Данный подход позволяет нам легко масштабировать нагрузку.

Использованные источники

abbitMQ. Documentation [Электронный ресурс] – Режим доступа:

оптимизация обработки сообщений RabbitMQ [Электронный ресурс] – Режим доступа: <https://habr.com/ru/articles/153431/>. Дата обращения: 31.03.2023.