

Publish–subscribe message queue

Politechnika Poznańska, Informatyka Semestr V
Sieci Komputerowe 2

Klaudia Kowalska 148184
Michał Zieliński 148064

1. Czym jest kolejka komunikatów? (message queue)

Kolejka komunikatów to protokół komunikacyjny o charakterze asynchronicznym. Nadawca i odbiorca wiadomości nie muszą łączyć się i dokonywać interakcji z kolejką w tym samym czasie. Kolejki wiadomości implementują schemat **publish subscribe pattern**. Jest to sposób przesyłania wiadomości, w którym **publisher** nie przesyła wiadomości bezpośrednio do **subscribiera**, a nadaje wiadomościom tagi/klasy i przesyła je bez wiedzy o istnieniu subskrybentów. W tym sposobie komunikacji wiadomości wysyłane są do tzw. **brokera wiadomości**. Jest to serwis świadczący usługi między nadawcami a odbiorcami. Wiadomości przekazywane są do kolejki, która sama sprawdza czy istnieją subskrybenci wiadomości danego tematu, a następnie przesyła je dalej do klientów końcowych.

Takie rozwiązanie zapewnia o wiele lepszą **skalowalność** systemu oraz **dynamiczną topologię infrastruktury sieciowej**. Z tego powodu bardzo często stosuje się kolejki wiadomości do komunikacji:

- w systemach czasu rzeczywistego (RTOS)
- między serwisami w architekturze mikroservisowej

Standardowym protokołem implementowanym przez kolejki wiadomości jest **AMQP - Advanced Message Queuing Protocol**, niektóre rozwiązania takie jak Apache Kafka implementują swoje własne protokoły oparte na TCP/IP.

2. Charakterystyka gotowych rozwiązań opartych na schemacie message queue.

2.1. Apache Kafka - jest to opensourcowa platforma do event streamingu. Implementuje schemat pub/sub, natomiast nie korzysta ze standardowej kolejki wiadomości. Wiadomości nadpisywane są w logu i pozostawione tam do momentu odczytania przez konsumenta. Cechuje się dużą przepustowością. Wykorzystywana w real-time processing i analizowaniu danych.

2.2. RabbitMQ - stworzony w Erlangu broker wiadomości. Cechuje się bardzo niskim czasem opóźnienia. I łatwą konfiguracją ze strony klienta. Idealny do wykorzystania w komunikacji między aplikacjami.

2.3. **ActiveMQ** - kolejne rozwiązanie wyprodukowane przez Apache. Również open source napisany w Javie. Oferuje również rozwiązania biznesowe. Jest trochę prostszy i bardziej ograniczony od Kafki.

3. Nasza kolejka wiadomości

Nasza kolejka wiadomości jest to rozwiązanie napisane w C++, zapewniający komunikację z publisherem i subskrybentami poprzez protokół TCP/IP. Klienci mogą rejestrować swoje tematy wiadomości i subskrybować się do tych już istniejących.

System oferuje możliwość konfiguracji parametrów takich jak:

- klucz identyfikacyjny - niezbędny do zarejestrowania klienta w systemie (hasło do kolejki)
- Maksymalna liczba użytkowników podłączonych do serwera
- Maksymalna liczba tematów wiadomości
- Maksymalna liczba wiadomości w kolejce, które jeszcze nie zostały przeczytane

Za pomocą tych parametrów zapewniamy bezpieczeństwo oraz możliwość dopasowania implementacji kolejki do posiadanych zasobów i infrastruktury.

Operacje możliwe do wykonania ze strony klienta:

- Połączenie - klient otrzymuje zachętę od serwera z prośbą o podanie klucza identyfikacyjnego
- Logowanie - klient wysyła do serwera wiadomość z kluczem - wiadomość jest sprawdzana, a klient zostaje zautoryzowany w kolejce lub nie.
- Dodawanie wiadomości na dany temat - klient podaje tag/temat wiadomości oraz jej treść. W przypadku, gdy dany tag nie istnieje w systemie, a nie został osiągnięty limit tematów i wiadomości - system utworzy nowy temat.
- Wyświetlenie listy dostępnych tematów/tagów
- Subskrypcja do danego tematu - klient wysyła żądanie subskrypcji danego tematu - jeśli dany temat istnieje - operacja się powiedzie i klient będzie dostawał wiadomości, które pojawią się z danym tagiem.
- Zdefiniowanie interwału czasowego keep-alive, po którym połączenie z serwerem zostanie przerwane jeśli klient nie wyśle wiadomości w tym czasie lub jeśli żadna wiadomość nie zostanie odebrana. Jeśli nie będzie wiadomości do wysłania, klient będzie okresowo wysyłał tzw. heartbeat message do brokera, aby zapewnić utrzymanie połączenia.

Operacje możliwe do wykonania ze strony serwera:

- Dodanie nowego tematu - po otrzymaniu wiadomości od klienta z prośbą o dodanie wiadomości do kolejki, serwer sprawdza czy podany temat już istnieje - jeśli nie to tworzy go

- Otrzymanie i dodanie nowych wiadomości do kolejki
- Autoryzacja klucza do subskrypcji kolejki - po otrzymaniu wiadomości od klienta z zapytaniem o subskrypcję tematu, serwer sprawdza czy podany klucz jest poprawny