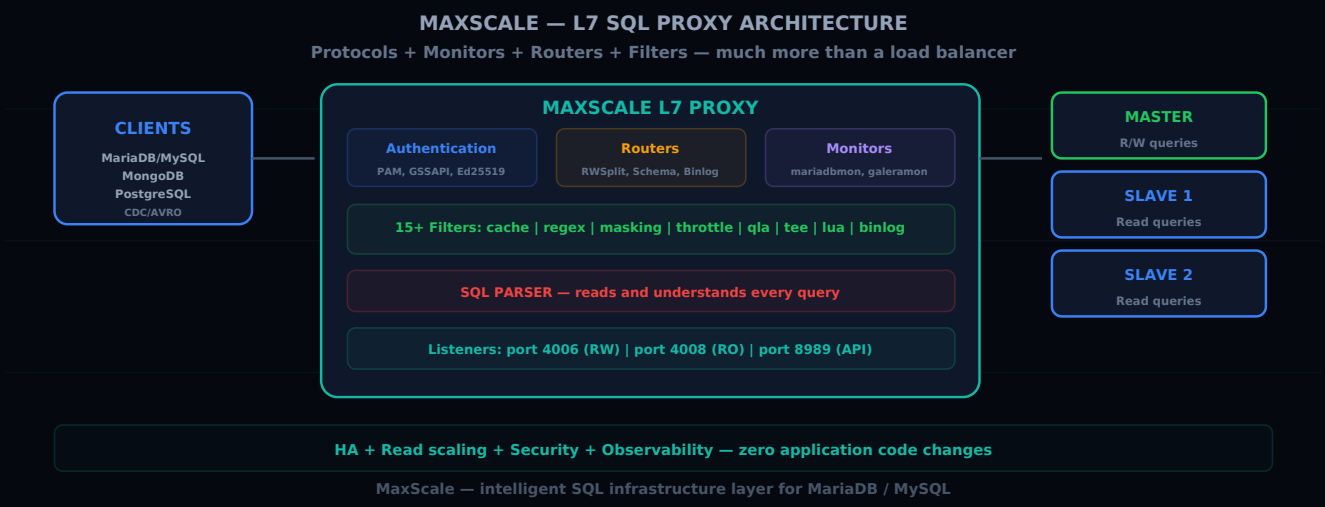
MaxScale: znacznie więcej niż reverse proxy SQL (część 1)

Sylvain ARBAUDIE · July 31, 2025

MAXSCALE MARIADB PROXY ARCHITECTURE



Nie prosty proxy

Gdy mowa o MaxScale, odruch nakazuje porównywać go z HAProxy lub klasycznym load balancerem. To fundamentalny błąd. MaxScale nie jest proxy warstwy 4 (TCP). To **proxy warstwy 7**, który rozumie protokół SQL. Parsuje zapytania, analizuje ich typ i podejmuje inteligentne decyzje routingowe.

Różnica jest porównywalna do tej między listonoszem dostarczającym pocztę według adresu (L4) a asystentem, który czyta Twoje e-maile, sortuje je według priorytetu, filtruje spam i pokazuje Ci tylko to, co istotne (L7). MaxScale czyta SQL.

Uwierzytelnianie: poza hasłem

MaxScale obsługuje kompletny zakres mechanizmów uwierzytelniania:

- **MariaDBAuth**: natywne uwierzytelnianie MariaDB / MySQL z lokalnym cache danych uwierzytelniających

- **PAM**: integracja z katalogami korporacyjnymi (LDAP, Active Directory) przez Pluggable Authentication Modules
- **GSSAPI/Kerberos**: uwierzytelnianie SSO dla środowisk Windows / Active Directory
- **Ed25519**: uwierzytelnianie kluczem publicznym MariaDB, bezpieczniejsze niż klasyczny hash SHA

Uwierzytelnianie jest zarządzane na poziomie listenera. Można więc mieć różne metody uwierzytelniania na różnych portach: PAM dla aplikacji wewnętrznych, Ed25519 dla połączeń administracyjnych, MariaDBAuth dla kompatybilności legacy.

Protokoły: nie tylko SQL

MaxScale jest poliglotą. Obsługuje wiele protokołów wejściowych i wyjściowych:

MariaDBClient / MariaDBBackend

Natywny protokół MariaDB / MySQL. To główny przypadek użycia: aplikacje łączą się z MaxScale jak ze standardowym serwerem bazy danych.

CDC / AVRO

Protokół Change Data Capture pozwala strumieniować modyfikacje z binlogu w formacie AVRO. To idealne narzędzie do budowania potoków danych w czasie rzeczywistym, zasilania data lake'ów lub synchronizacji systemów zewnętrznych.

NoSQL / MongoDB

MaxScale może wystawić interfejs kompatybilny z MongoDB. Aplikacje komunikujące się protokołem MongoDB mogą interagować z bazą MariaDB przez MaxScale. To niszowa, ale potężna funkcjonalność do migracji.

PostgreSQL (eksperymentalny)

Obsługa protokołu PostgreSQL na wejściu jest w trakcie rozwoju. Celem jest umożliwienie aplikacjom PostgreSQL łączenia się z backendami MariaDB.

Monitory: inteligencja topologii

Monitory to oczy MaxScale. Regularnie odpytują serwery backendowe, by automatycznie wykrywać topologię i stan zdrowia klastra.

mariadbmon

Główny monitor dla topologii replikacji MariaDB / MySQL. Wykrywa mastera, slave'y, relay'e i stan replikacji. Zarządza automatycznym failoverem: jeśli master padnie, slave jest promowany, a pozostałe slave'y są rekonfigurowane do replikacji z nowego mastera.

galeramon

Monitor dedykowany klastrom Galera. Wykrywa stan klastra (Primary, Non-Primary, Disconnected), liczbę węzłów, state UUID i zarządza routingiem odpowiednio.

Routery: inteligencja routingu

Routery to mózg MaxScale. Decydują, gdzie wysłać każde zapytanie w zależności od jego typu i wykrytej topologii.

readwritesplit

Najczęściej używany router. Zapytania zapisu (`INSERT` , `UPDATE` , `DELETE` , `CREATE` itp.) są wysyłane do mastera. Zapytania odczytu (`SELECT`) są rozdzielane na slave'y. Transakcje są wysyłane w całości do mastera.

readconnroute

Prostszy router rozdzielający połączenia (nie zapytania) na dostępne serwery. Przydatny dla intensywnych odczytów niewymagających separacji odczytu/zapisu na poziomie zapytania.

schemarouter

Kieruje zapytania do serwera hostującego żądany schemat. Idealny do shardingu po bazie danych: `client_europe` na serwerze A, `client_asia` na serwerze B.

binlogrouter

Przekształca MaxScale w relay replikacji. MaxScale zachowuje się jak slave mastera, a rzeczywiste slave'y łączą się z MaxScale zamiast z masterem. Redukuje to obciążenie mastera i centralizuje dystrybucję binlogu.

kafkarouter (CDC)

Wysyła zdarzenia z binlogu do Apache Kafka. Każda modyfikacja bazy jest publikowana jako wiadomość Kafka, pozwalając konsumentom reagować w czasie rzeczywistym.

Filtry: ponad 15 modułów

Filtry przechwytują i transformują strumień SQL między klientem a serwerem. MaxScale oferuje ponad 15:

- **qlafilter**: kompletne logowanie zapytań (audyt)
- **regexfilter**: przepisywanie zapytań wyrażeniami regularnymi
- **cache**: cache zapytań z automatyczną inwalidacją
- **throttlefilter**: ograniczanie przepustowości zapytań na użytkownika
- **masking**: dynamiczne maskowanie wrażliwych danych (e-maile, numery kart)
- **topfilter**: zbieranie najwolniejszych zapytań
- **commentfilter**: wstrzykiwanie komentarzy SQL do śledzenia
- **tee**: duplikacja strumienia do drugiego backendu (testowanie w cieniu)
- **namedserverfilter**: routing do konkretnego serwera na podstawie reguł
- **hintfilter**: interpretacja wskazówek SQL do wymuszenia routingu
- **luafilter**: wykonywanie skryptów Lua dla niestandardowej logiki
- **binlogfilter**: filtrowanie zdarzeń binlogu według schematu lub tabeli

Wdrożenie: gotowy na Docker i Kubernetes

MaxScale jest dostępny jako oficjalny obraz Docker:

```
docker run -d --name maxscale \
  -v /path/to/maxscale.cnf:/etc/maxscale.cnf \
  -p 4006:4006 -p 8989:8989 \
  mariadb/maxscale:latest
```

Dla Kubernetes MaxScale wdraża się jako StatefulSet lub Deployment, zależnie od przypadku użycia. REST API ułatwia integrację z health checks i probes readiness.

Wartość biznesowa

Poza funkcjonalnościami technicznymi MaxScale wnosi konkretną wartość biznesową:

- **Wysoka dostępność**: automatyczny failover w kilka sekund, transparentny dla aplikacji
- **Skalowalność odczytów**: dodawanie slave'ów bez modyfikacji kodu aplikacji
- **Bezpieczeństwo**: filtrowanie SQL, maskowanie danych, scentralizowane uwierzytelnianie
- **Obserwowalność**: logowanie, metryki, REST API do monitoringu
- **Ułatwiona migracja**: obsługa wielu protokołów dla przejść technologicznych

MaxScale nie jest prostym reverse proxy. To inteligentna warstwa infrastrukturalna umieszczona między aplikacjami a bazami MariaDB / MySQL, wnosząca odporność, bezpieczeństwo i elastyczność bez modyfikowania ani jednej linii kodu aplikacji.

Ten artykuł został pierwotnie opublikowany na Medium.