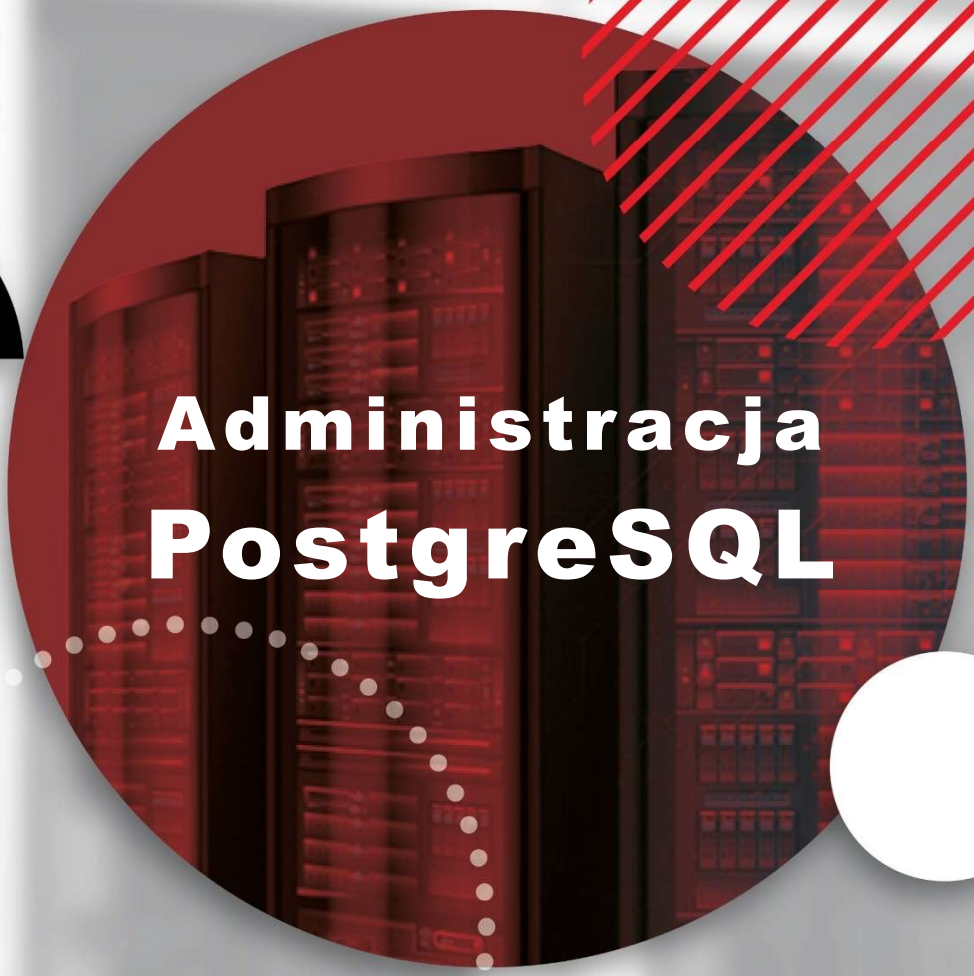
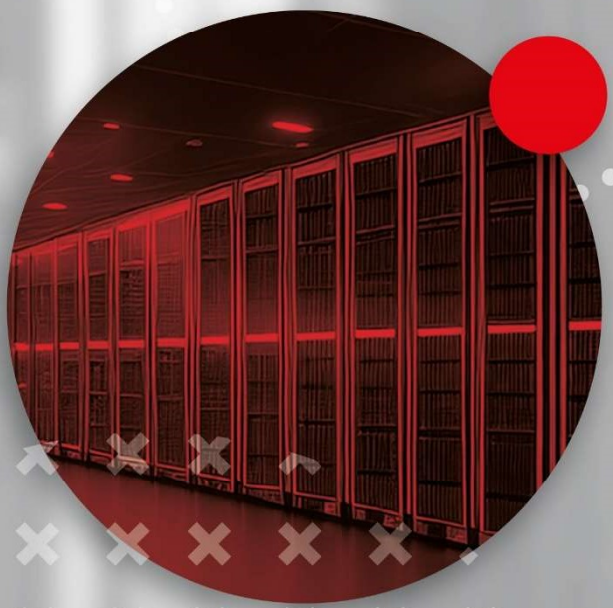
JSYSTEMS



Administracja PostgreSQL



Replikacja fizyczna, strumieniowa hot-standby

ćwiczenia

PostgreSQL

Replikacja fizyczna, strumieniowa hot-standby

1	Przygotuj 2 hosty w tej samej podsieci.
2	Na obu hostach zainstaluj binaria PostgreSQL. Pierwszy będzie serwerem primary, a drugi repliką.
3	Na obu serwerach dodaj binaria PostgreSQL do zmiennej PATH użytkownika systemowego “postgres”.
4	Na obu serwerach wyłącz i dezaktywuj uruchamianie usługi systemowej PostgreSQL.
5	Na pierwszym serwerze (nasz przyszły primary) zainicjalizuj klaster PostgreSQL w lokalizacji “/data_pg”.
6	Na pierwszym serwerze (primary) uruchom klaster znajdujący się w “/data_pg” za pomocą “pg_ctl”.
7	Na pierwszym serwerze (primary) skonfiguruj parametry “listen_addresses”, “wal_level”, “max_wal_senders”, “max_replication_slots” w taki sposób, byśmy mogli podłączyć do tego serwera replikę i jednocześnie mogli wykonywać gorącą kopię zapasową za pomocą pg_basebackup (który wykorzystuje ten sam protokół strumieniowania, co replikacja).
8	Włącz logging_collector na serwerze primary, byśmy w przypadku problemów mogli zajrzeć do loga.
9	Zadbaj o to, by zmiana parametrów weszła w życie i zweryfikuj czy faktycznie zostały one zmienione.
10	Na pierwszym serwerze (primary) stwórz użytkownika “replikacja”, który będzie służył do łączenia się przez serwery replica. Pamiętaj, że taki użytkownik musi mieć uprawnienia “replication”. Nadaj mu też uprawnienia do wykorzystania niezbędnych do replikacji funkcji.
11	Na pierwszym serwerze (primary) dodaj wpis do “pg_hba.conf” pozwalający na wykonywanie replikacji przez użytkownika “replikacja” w ramach podsieci lokalnej. Jeśli korzystasz z Azure to adres podsieci będzie najprawdopodobniej 10.0.0.0/16. Przeładuj konfigurację.
12	Na pierwszym serwerze (primary) utwórz tabelę: `create table test_replication(x integer);`
13	Na serwerze 2 utwórz replikę asynchroniczną. Ustaw jej nazwę na “my_replica” i uruchom ją. Zaloguj się do repliki i sprawdź czy istnieje tabela “test_replication”.
14	Na pierwszym serwerze (primary) zweryfikuj za pomocą słownika “pg_stat_replication” czy replikacja działa.

15	Na pierwszym serwerze (primary) utwórz bazę danych “testing” i załadują ją danymi testowymi za pomocą “pgbench”:
	`psql -c "create database testing"`
	`pgbench -i -s 50 testing`

16	Porównamy teraz wydajność replikacji asynchronicznej i synchronicznej. Uruchom poniższe polecenie (potrwa 5 minut) i odnotuj wynik, który pojawi się na samym dole w formacie “tps=XXX.XXXX”. To wartość określająca ile średnio operacji (SELECT, UPDATE, INSERT, DELETE) udało się wykonać na sekundę: `pgbench -T 300 testing -P 1`
17	Przełącz konfigurację tak by nasza replika była repliką synchroniczną (podpowiedź: parametr “sychronous_standby_names” na primary).

18	Zweryfikuj za pomocą słownika “pg_stat_replication” na primary, czy nasza replika działa w trybie synchronicznym.
19	Ponów test wydajnościowy i ponownie jak wcześniej odnotuj wynik “tps”: pgbench -T 300 testing -P 1
20	Czy zauważyłeś różnicę w wydajności replikacji synchronicznej i asynchronicznej? Z czego wynika ta różnica?
21	Sprawdzimy teraz opóźnienia przy replikacji asynchronicznej. Przełącz replikę na replikację asynchroniczną.
22	Nawiąż 3 sesje. Dwie pierwsze do serwera primary, trzecią do serwera replica.
23	Z pierwszej z 3 sesji (tej do serwera primary) stwórz tabelę: create table test_async(x integer);
24	W sesji drugiej i trzeciej (tych do serwera replica) wykonaj polecenie: select count(*) from test_async;
25	W sesji drugiej i trzeciej (tych do serwera replica) uruchom poniższe polecenie, powodujące wykonywanie ostatniej operacji (naszego counta) co 100ms: \watch 0.1
26	Ustaw sesje 2 i 3 (te do serwera replica) tak by ich okna były cały czas widoczne. Z sesji pierwszej wykonaj poniższe polecenia i obserwuj sesje 2 i 3. Pobrany skrypt ładuje do tabeli “test_async” kolejne wiersze: cd /tmp wget http://jsystems.pl/kampania_pg_adm/test_async/data.sql psql -f /tmp/data.sql
27	Jakie było opóźnienie przy replikacji asynchronicznej? Czy było ono odczuwalne?
28	Biorąc pod uwagę wnioski z porównania wydajności i testu opóźnienia zastanów się czy na pewno potrzebujesz replikacji synchronicznej, czy nie wystarczy zwykła replikacja asynchroniczna.