Wikiprint Book

Title: ARTYKUŁ W TRAKCIE TWORZENIA

Subject: eDokumenty - elektroniczny system obiegu dokumentów, workflow i CRM - AdminGuide/UpgradePostgresql1213

Version: 20

Date: 05/06/24 14:49:24

Table of Contents

ARTYKUŁ W TRAKCIE TWORZENIA 3
Aktualizacja bazy danych PostgreSQL do wersji 12 lub nowszej 3
1. Hasła scram-sha-256 lub md5 3
2. Konfiguracja pgbouncer 3
3. Usunięcie OIDs przed odtworzeniem bazy 3
4. Wyłączenie JUST-IN-TIME 4

ARTYKUŁ W TRAKCIE TWORZENIA

Aktualizacja bazy danych PostgreSQL do wersji 12 lub nowszej

1. Hasła scram-sha-256 lub md5

Od wersji 12 PostgreSQL domyślnie ustawia wartość dla haseł:

```
password_encryption = scram-sha-256
```

Po ustawieniu haseł w PostgreSQL wygląda to następująco:

```
  usename   | usesysid | usecreatedb | usesuper | userepl | usebypassrls |
------------+----------+-------------+----------+---------+--------------+-----------------------------------------------
http        |    16385 | f           | f        | f       | f            | SCRAM-SHA-256$4096:D1NPKy+ZDiGGF1/E8q5M/g==$kD4a
edokumenty |    16384 | f           | f        | f       | f            | SCRAM-SHA-256$4096:9Q2v9SU06DQ2bHV/tCu1XA==$GqcL
postgres   |       10 | t           | t        | t       | t            | SCRAM-SHA-256$4096:2gkT6rlhnnvIWi9K5L403Q==$J3bm
```

Jeśli wykonujemy pg_upgradecluster z wersji gdzie domyślnie korzystaliśmy szyfrowania md5. Do nowego PostgreSQL zostanie przeniesione to samo ustawienia. Dlatego jeśli chcemy przejść na scram-sha-256 należy zmienić ustawienie w PostgreSQL

```
vim /etc/postgresql/12/main/postgresql.conf
password_encryption = scram-sha-256
:wq
```

A następnie restart usługi postgreSQL

Po zalogowaniu os psql wykonać polecenie

```
SELECT
   rolname, rolpassword ~ '^SCRAM-SHA-256\$' AS has_upgraded
FROM pg_authid
WHERE rolcanlogin;
```

Jeśli zwróci nam FALSE, konieczne będzie zmiana hasła np:

```
ALTER user postgres with encrypted password 'hasło';
```

Zmiana konfiguracji pg_hba.conf

```
vim /etc/postgresql/13/main/pg_hba.conf
```

```
# "local" is for Unix domain socket connections only
local   edokumenty             edokumenty                 scram-sha-256
local   edokumenty             http                       scram-sha-256
```

2. Konfiguracja pgbouncer

Zmiana konfiguracji w przypadku skorzystania z szyfrowania scram-sha-256 została opisana tym artykule.

3. Usunięcie OIDs przed odtworzeniem bazy

Podczas odtwarzania starej bazy czyli na przykład gdy mamy od czasu systemu od wersji 4.10 możemy napotkać problem z OIDs, co podczas odtwarzania bazy lub wykonania pg_upgradecluster otrzymamy komunikat:

Podczas odtawrznia bazy restore

```
ERROR:  tables declared WITH OIDS are not supported
```

Podczas pg_upgradecluster

```
pg_dump: warning: WITH OIDS is not supported anymore (table ""documents_2_2020"")
pg_dump: warning: WITH OIDS is not supported anymore (table ""log"")
pg_dump: warning: WITH OIDS is not supported anymore (table ""ws_incoming_messages"")
pg_dump: warning: WITH OIDS is not supported anymore (table ""ws_box_incoming_messages"")
pg_dump: warning: WITH OIDS is not supported anymore (table ""ws_box_incoming_messages_2_2020"")[[Image()]]
```

przed aktualizacja należy usunąć OIDy z bazy. Ten problem rozwiązuje aktualizacja do wersji 6.64.4, którą możemy wykonać na PostgreSQL w wersji 10 lub 11. Po wykonaniu patcha 6.0/041 OIDs zostaną usunięte z bazy. Następnie będziemy mogli przejść do aktualizacji PostgreSQL 12 lub nowszej.

4. Wyłączenie JUST-IN-TIME

Po aktualizacji do PostgreSQL 12/13 należy wyłączyć jit, która domyślnie jest włączona. Z naszych testów wynika, że aplikacja Ready_™ działa szybciej z wyłączonym jit.

```
jit = off # allow JIT compilation
```