



JSYSTEMS

Administracja PostgreSQL

**KONFIGURACJA
PARAMETRÓW**
cheatsheet



Konfiguracja parametrów - cheatsheet

Sprawdzanie obowiązującego ustawienia parametru dla naszej sesji	show nawa_parametru; #np: show log_line_prefix;
Sprawdzanie wartości wszystkich parametrów obowiązujących dla naszej sesji	show all;
Słownik z ustawieniami parametrów i ich poziomami konfiguracji	select * from pg_settings;
Sprawdzanie ustawienia parametru za pomocą słownika pg_settings	select name,setting from pg_settings where name=<nazwa parametru>; #np: select name,setting from pg_settings where name='log_line_prefix';
Słownik z ustawieniami parametrów wynikającymi z plików (czyli obowiązujących dla całego klastra)	select * from pg_file_settings;
Sprawdzanie dostępnych poziomów konfiguracji	select distinct context from pg_settings;
Ustawienie parametru na poziomie sesji	set <parametr>=<wartość>; #np: set work_mem='16MB';
Reset ustawienia parametru na poziomie sesji	reset <nazwa parametru>; #np: reset work_mem;
Reset wszystkich parametrów na poziomie sesji	reset all;
Zmiana parametru poprzez update na słowniku	update pg_settings set setting=<wartość> where name='<nazwa parametru>;' #np: update pg_settings set setting='32MB' where name='work_mem';
Zmiana parametru na poziomie bazy danych	alter database <nazwa bazy> set <parametr>=<wartość>; #np: alter database test_config set work_mem='123MB';
Sprawdzenie parametrów ustawionych na poziomie bazy danych	select datname,setconfig from pg_db_role_setting pdrs join pg_database pd on pdrs.setdatabase=pd.oid;
Zmiana parametru na poziomie użytkownika	alter user <nazwa użytkownika> set <parametr>=<wartość>; #np: alter user mapet set work_mem='40MB';
Sprawdzenie parametru na poziomie użytkownika	select username, useconfig from pg_shadow;
Wyszukanie użytkowników posiadających indywidualnie ustawiony wybrany parametr	select * from (select username,unnest(useconfig) as config from pg_user) e where config like '%work_mem%';
Zmiana parametru na poziomie użytkownika w bazie danych	alter user <nazwa użytkownika> in database <nazwa bazy> set <parametr>=<wartość>; #np: alter user mapet in database test_config set work_mem='64MB';
Sprawdzanie indywidualnych ustawień dla bazy danych,	SELECT r.rolname, d.datname, rs.setconfig FROM pg_db_role_setting rs

użytkownika i użytkownika w bazie danych	<pre>LEFT JOIN pg_roles r ON r.oid = rs.setrole LEFT JOIN pg_database d ON d.oid = rs.setdatabase;</pre>
Resetowanie ustawień na poziomie bazy danych	<pre>alter database <nazwa bazy> reset <nazwa parametru>; #np: alter database test_config reset work_mem;</pre>
Resetowanie ustawień na poziomie użytkownika	<pre>alter role <nazwa użytkownika> reset <nazwa parametru>; #np: alter role mapet reset work_mem;</pre>
Resetowanie ustawień na poziomie użytkownika w bazie danych	<pre>alter role <nazwa użytkownika> in database <nazwa bazy> reset <nazwa parametru>; #np: alter role mapet in database test_config reset work_mem;</pre>
Edycja pliku konfiguracyjnego w celu zmiany parametrów na poziomie klastra	<pre>nano /data_pg/postgresql.conf</pre>
Zmiana parametru na poziomie klastra za pomocą SQL	<pre>alter system set <nazwa parametru>=<wartość>; #np: alter system set work_mem='32MB'; #przeładowanie konfiguracji lub restart w zależności od parametru psql -c "select pg_reload_conf()" #lub: pg_ctl -D /data_pg restart #alternatywnie do w.w.: sudo systemctl restart postgresql15</pre>
Dodatkowy plik z parametrami (umieszczamy w postgresql.conf)	<pre>include '<nazwa pliku>' #np: include 'memory.conf'</pre>
Dodatkowy katalog zawierający pliki z parametrami (umieszczamy w postgresql.conf)	<pre>include_dir 'nazwa_katalogu' #np: include_dir 'additional_configs' pg_ctl -D /data_pg restart #alternatywnie do w.w.: sudo systemctl restart postgresql15 #pliki muszą mieć rozszerzenie .conf</pre>