Rozdział 16
Uwierzytelnianie i autoryzacja w bazie danych

Uprawnienia, role, synonimy

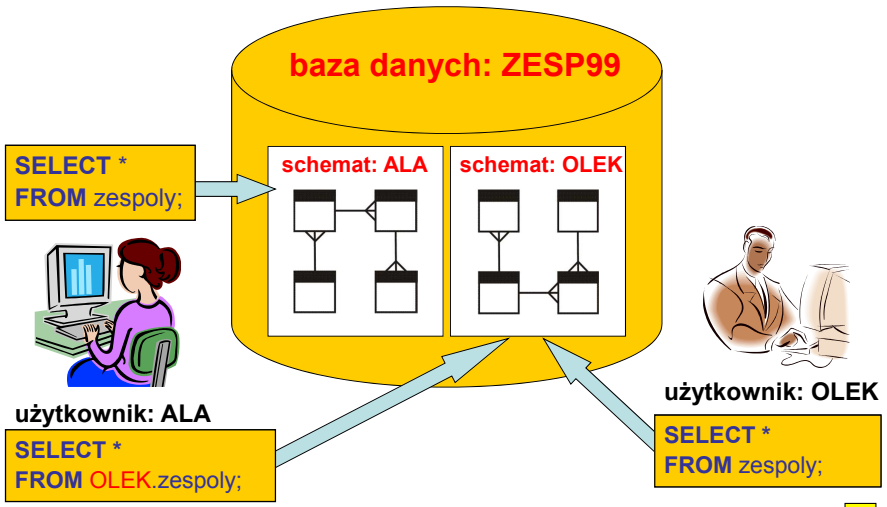
Plan ćwiczenia

- **Użytkownicy i schematy bazy danych.**
- **Uwierzytelnianie i autoryzacja.**
- **Przywileje systemowe i obiektowe.**
- **Role.**
- **Synonimy.**

Użytkownicy i schematy (1)

- **Użytkownik – osoba lub aplikacja, uprawniona do dostępu do danych zgromadzonych w bazie danych.**
- **Schemat – kolekcja logicznych struktur danych (relacji, perspektyw, ograniczeń, indeksów, programów składowanych, itd.),**
  - **schemat posiada swoją nazwę,**
  - **schemat jest własnością określonego użytkownika.**

Użytkownicy i schematy (2)



Informacje o użytkownikach

ALL_USERS	Zawiera nazwy i daty utworzenia wszystkich użytkowników w bazie danych
DBA_USERS	Zawiera nazwę, identyfikator, zakodowane hasło, nazwę domyślnej przestrzeni tabel,nazwę tymczasowej przestrzeni tabel, datę utworzenia i profil każdego użytkownika bazy danych
USER_USERS	Zawiera nazwę, identyfikator, domyślną przestrzeń tabel, tymczasową przestrzeń tabel i datę utworzenia aktualnego użytkownika

```
SELECT * FROM user_users;
```

```
SELECT username FROM all_users;
```


Uwierzytelnianie użytkowników

- Uwierzytelnianie to proces weryfikacji tożsamości użytkownika bazy danych.
- Metody uwierzytelniania, wykorzystywane przez SZBD:
  - uwierzytelnianie przez SZBD,
  - uwierzytelnianie przez system operacyjny,
  - uwierzytelnianie przez usługę sieciową (Kerberos, RADIUS, …),
  - uwierzytelnianie wielowarstwowe.

Autoryzacja użytkowników

- Autoryzacja jest procesem weryfikacji uprawnień użytkownika do wykonania określonej operacji:
  - ograniczenia co do obiektów i operacji, jakie użytkownik może na nich realizować:
    - system przywilejów,
    - role bazodanowe,
  - ograniczenie wykorzystania zasobów systemowych (czasu procesora, czasu połączenia, itd.) przy realizacji operacji:
    - profile.

Przywileje

- Przywilej – prawo wykonywania przez użytkownika określonej akcji w bazie danych lub dostępu do określonego obiektu, np.:
  - przyłączenie się do bazy danych,
  - odczyt danych określonej relacji,
  - wykonanie dowolnej składowanej w bazie danych procedury,
  - utworzenia perspektywy, itd.
- Rodzaje przywilejów:
  - systemowe,
  - obiektowe.

Przywileje systemowe (1)

- **Prawa wykonania:**
  - **określonej akcji w bazie danych, lub**
  - **określonej operacji na wskazanym typie obiektu we wskazanym/dowolnym schemacie bazy danych.**
- **Przykłady:**
  - **CREATE SESSION**
  - **CREATE TABLE**
  - **CREATE ANY TABLE**
  - **SELECT ANY TABLE**
  - **INSERT ANY TABLE**
  - **DROP ANY VIEW**

Przywileje systemowe (2)

- **Nadawanie przywilejów systemowych:**

```
GRANT <lista_przywilejów_systemowych>
TO <lista_użytkowników> | PUBLIC
[WITH ADMIN OPTION];
```

- **Odbieranie przywilejów systemowych:**

```
REVOKE <lista_przywilejów>
FROM <lista_użytkowników> | PUBLIC;
```

- **Przykład:**

```
GRANT CREATE SESSION, SELECT ANY TABLE TO OLEK;
GRANT CREATE TABLE TO ALA;
REVOKE SELECT ANY TABLE FROM OLEK;
```


Informacje o przywilejach systemowych

- **USER_SYS_PRIVS - uprawnienia systemowe nadane użytkownikowi bezpośrednio**
- **SESSION_PRIVS - uprawnienia systemowe nadane użytkownikowi zarówno bezpośrednio jak i przez role**

```
SQL> SELECT * FROM user_sys_privs;

USERNAME   PRIVILEGE                      ADMIN_OPTION
---------- ------------------------------ ----------------------
OLEK       CREATE SESSION                 NO
OLEK       INSERT ANY TABLE               NO
OLEK       UNLIMITED TABLESPACE           YES
OLEK       CREATE VIEW                    YES
```


Przywileje obiektowe (1)

- **Prawa wykonania określonej operacji na wskazanym obiekcie w określonym schemacie bazy danych.**
- **Przykłady:**
  - **SELECT ON pracownicy**
  - **ALTER ON zespoly**
  - **EXECUTE ON PoliczPracownikow**
  - **REFERENCES ON etaty**
- **Uwaga!**

  Użytkownik posiada wszystkie uprawnienia do obiektu, którego jest właścicielem! Nie można mu tych przywilejów odebrać.

Przywileje obiektowe (2)

Przywilej	Relacja	Perspektywa	Procedura/ funkcja/pakiet	Sekwencja
ALTER	+			+
DELETE	+	+		
EXECUTE			+	
INDEX	+			
INSERT	+	+		
REFERENCES	+	+		
SELECT	+	+		+
UPDATE	+	+		

Nadawanie przywilejów obiektowych

- **Polecenie:**

```
GRANT <lista_przywilejów> | ALL ON <nazwa_obiektu>
TO <lista_użytkowników> | PUBLIC
[WITH GRANT OPTION];
```

  - przywileje INSERT, UPDATE i REFERENCES mogą dodatkowo specyfikować kolumny relacji, np.:
    - INSERT(id_prac, nazwisko)

Odbieranie przywilejów obiektowych

- **Polecenie:**

```
REVOKE <lista_przywilejów> | ALL ON <nazwa_obiektu>
FROM <lista_użytkowników> | PUBLIC
[CASCADE CONSTRAINTS];
```

  - brak możliwości specyfikacji kolumn relacji przy odbieraniu przywilejów INSERT, UPDATE i REFERENCES
  - odebranie przywileju REFERENCES wymaga dodania opcji CASCADE CONSTRAINTS, powodującej automatyczne usunięcie kluczy obcych zdefiniowanych w czasie obowiązywania przywileju

Przykłady operacji z przywilejami obiektowymi

- **Nadawanie przywilejów:**

```
ALA> GRANT SELECT ON zespoly TO OLEK;
ALA> GRANT REFERENCES(id_prac) ON pracownicy TO OLEK;
OLEK>       GRANT UPDATE(placa_pod, placa_dod)
       ON pracownicy TO ALA;
```

- **Odbieranie przywilejów:**

```
OLEK> REVOKE UPDATE ON pracownicy FROM ALA;
ALA> REVOKE REFERENCES ON pracownicy FROM OLEK
   CASCADE CONSTRAINTS;
```

Informacje o przywilejach obiektowych

USER_TAB_PRIVS	**Uprawnienia do obiektów, których użytkownik jest właścicielem oraz uprawnienia obiektowe, które użytkownik otrzymał lub przyznał**
USER_TAB_PRIVS_MADE	**Uprawnienia do obiektów, będących własnością użytkownika**
USER_TAB_PRIVS_RECD	**Uprawnienia obiektowe, które użytkownik otrzymał**
USER_COL_PRIVS	**Uprawnienia do atrybutów, których użytkownik jest właścicielem oraz uprawnienia do atrybutów, które użytkownik otrzymał lub przyznał**
USER_COL_PRIVS_MADE	**Uprawnienia do atrybutów będących własnością użytkownika**
USER_COL_PRIVS_RECD	**Uprawnienia do atrybutów, które użytkownik otrzymał**

```
SELECT * FROM user_tab_privs_made;
SELECT * FROM user_tab_privs_recd;
```


Opcja administracyjna przywilejów

- **umożliwia użytkownikowi przekazanie innym użytkownikom otrzymanego przywileju (systemowego lub obiektowego),**
- **przywileje systemowe:**

```
ADMIN: GRANT SELECT ANY TABLE TO SCOTT WITH ADMIN OPTION;
SCOTT: GRANT SELECT ANY TABLE TO JONES;
```

- **odebranie przywileju systemowego użytkownikowi SCOTT nie powoduje odebrania przywileju użytkownikowi JONES**
- **przywileje obiektowe:**

```
SCOTT: GRANT INSERT ON ETATY TO JONES WITH GRANT OPTION;
JONES: GRANT INSERT ON SCOTT.ETATY TO SMITH;
```

- **odebranie przywileju obiektowego użytkownikowi JONES powoduje odebranie przywileju również użytkownikowi SMITH**

Przywilej EXECUTE (1)

- **Domyślnie kod składowany wykonywany jest z zestawem przywilejów użytkownika definiującego (domyślna klauzula AUTHID DEFINER)**

```
ALA> CREATE FUNCTION Ile
RETURN NUMBER AUTHID DEFINER IS
  vIle number;
BEGIN
  SELECT count(*) INTO vIle
  FROM   ala.pracownicy;
  RETURN vIle;
END Ile;
ALA> GRANT EXECUTE ON Ile TO
     olek;
```

```
OLEK> SELECT count(*)
      FROM ala.pracownicy;
ORA-00942: tabela lub
      perspektywa nie istnieje

OLEK> SELECT ala.Ile FROM dual;
14
```


Przywilej EXECUTE (2)

- **Można zdefiniować kod składowany wykonywany z zestawem przywilejów użytkownika wykonującego (dodatkowa klauzula AUTHID CURRENT_USER)**

```
ALA> CREATE FUNCTION Ile …
AUTHID CURRENT_USER …
ALA> GRANT EXECUTE ON Ile TO
     olek;

ALA> GRANT SELECT ON pracownicy
     TO olek;
```

```
OLEK> SELECT count(*)
      FROM ala.pracownicy;
ORA-00942: tabela lub
      perspektywa nie istnieje
OLEK> SELECT ala.Ile FROM dual;
ORA-00942: tabela lub
      perspektywa nie istnieje

OLEK> SELECT ala.Ile FROM dual;
14
```

Role (1)

- **Rola – nazwany zbiór powiązanych przywilejów.**
- **Ułatwiają zarządzanie systemem przywilejów.**
- **Użytkownik może mieć nadanych wiele ról, rola może zostać przyznana wielu użytkownikom lub rolom.**
- **Przykłady zastosowań:**
  - **role aplikacyjne – grupują przywileje niezbędne do działania określonej aplikacji,**
  - **role użytkowników – tworzone dla grup użytkowników, wymagających tych samych przywilejów.**

Role (2)

- **Predefiniowane role w SZBD Oracle (niektóre):**
  - **CONNECT** – przywilej: CREATE SESSION,
  - **RESOURCE** – przywileje:
    - CREATE CLUSTER,
    - CREATE INDEXTYPE,
    - CREATE OPERATOR,
    - CREATE PROCEDURE,
    - CREATE SEQUENCE,
    - CREATE TABLE,
    - CREATE TRIGGER,
    - CREATE TYPE
  - **DBA** – wszystkie przywileje systemowe z opcją administracyjną (WITH ADMIN OPTION).

Role (3)

- **Tworzenie roli:**
  - **bez uwierzytelniania:**

```
CREATE ROLE <nazwa_roli> [NOT IDENTIFIED];
```

  - **uwierzytelnianej przez SZBD:**

```
CREATE ROLE <nazwa_roli> IDENTIFIED BY <hasło>;
```

  - **uwierzytelnianej przez pakiet:**

```
CREATE ROLE <nazwa_roli> IDENTIFIED USING <nazwa_pakietu>;
```

- **Usunięcie roli:**

```
DROP ROLE <nazwa_roli>;
```


Role (4)

- **Nadawanie przywilejów roli:**

```
GRANT <lista_przywilejów> [ON <nazwa_obiektu>] TO <lista_ról>;
```

- **Nadawanie roli użytkownikowi/roli:**

```
GRANT <lista_ról> TO <lista_użytkowników> | <lista_ról> | PUBLIC
[WITH ADMIN OPTION];
```

- **Odbieranie przywilejów roli:**

```
REVOKE <lista_przywilejów> [ON <nazwa_obiektu>] FROM <lista_ról>;
```

- **Odbieranie roli użytkownikowi/roli:**

```
REVOKE <lista_ról> FROM <lista_użytkowników>|<lista_ról> PUBLIC;
```


Role (5)

- **Przykład:**

```
CREATE ROLE KASJER;
GRANT SELECT, UPDATE, DELETE ON PRACOWNICY TO KASJER;
GRANT KASJER TO KOWALSKI;
```


Role (6)

- **Zablokowanie/odblokowanie roli dla bieżącej sesji:**

```
SET ROLE [ <nazwa_roli> [ IDENTIFIED BY <hasło> ]
         | ALL [EXCEPT <nazwa_roli>
         | NONE ];
```

- **Przykład:**

```
CREATE ROLE KIEROWNIK IDENTIFIED BY pass123;
GRANT ALL ON PRACOWNICY TO KIEROWNIK;
GRANT KIEROWNIK TO NOWAK;
```

  - **użytkownik NOWAK włączy rolę poleceniem:**

```
SET ROLE KIEROWNIK IDENTIFIED BY pass123;
```


Informacje o rolach

USER_ROLE_PRIVS	**Role przyznane użytkownikowi**
ROLE_ROLE_PRIVS	**Role przyznane innym rolom**
ROLE_SYS_PRIVS	**Uprawnienia systemowe przyznane rolom**
ROLE_TAB_PRIVS	**Uprawnienia obiektowe przyznane rolom**
SESSION_ROLES	**Role użytkownika aktywowane w bieżącej sesji**

```
SELECT role, table_name, privilege
FROM role_tab_privs;
```

```
SELECT * FROM session_roles;
```


Efekty operacji GRANT i REVOKE (1)

- **Efekty operacji nadawania i odbierania przywilejów systemowych i obiektowych komukolwiek (użytkownikom, rolom, grupie PUBLIC) wchodzą w życie natychmiast w bieżącej sesji użytkownika.**

```
ALA> GRANT SELECT ON pracownicy
     TO olek;
```

```
OLEK> SELECT count(*)
      FROM ala.pracownicy;
ORA-00942: tabela lub
      perspektywa nie istnieje

OLEK> SELECT count(*)
      FROM ala.pracownicy;
COUNT(*)
----------
        14
```


Efekty operacji GRANT i REVOKE (2)

- **Efekty operacji nadawania i odbierania ról komukolwiek (użytkownikom, innym rolom, grupie PUBLIC) wchodzą w życie, gdy użytkownik:**
  - w bieżącej sesji wykona polecenie SET ROLE włączając rolę, lub
  - zakończy bieżącą sesję i rozpocznie nową.

```
ALA> CREATE ROLE dla_olka;
ALA> GRANT SELECT ON pracownicy
     TO dla_olka;
ALA> GRANT dla_olka TO olek;
```

```
OLEK> SELECT count(*)
      FROM ala.pracownicy;
ORA-00942: tabela lub
      perspektywa nie istnieje
OLEK> SET ROLE dla_olka;
OLEK> SELECT count(*)
      FROM ala.pracownicy;
COUNT(*)
----------
        14
```

Uwaga! Usunięcie roli powoduje natychmiastowe odebranie związanych z nią przywilejów.

Synonimy (1)

- **Synonim – alternatywna nazwa dla obiektu.**
- **Cele stosowania:**
  - **uproszczenie konstrukcji poleceń SQL,**
  - **ukrycie nazwy oryginalnego obiektu,**
  - **ukrycie lokalizacji oryginalnego obiektu (np. z innego schematu, ze zdalnej bazy danych).**
- **Rodzaje synonimów:**
  - **synonim prywatny – jest własnością określonego użytkownika, dostępny tylko dla właściciela oraz użytkowników, którzy uzyskają prawo dostępu,**
  - **synonim publiczny – dostępny dla każdego użytkownika bazy danych.**

Synonimy (2)

- **Tworzenie:**

```
CREATE [PUBLIC] SYNONYM <synonim> FOR <nazwa_obiektu>;
```

- **Usuwanie:**

```
DROP [PUBLIC] SYNONYM <synonim>;
```


Synonimy (3)

Przykład:

- **użytkownik OLEK wykonuje:**

```
OLEK> GRANT SELECT, INSERT ON pracownicy TO ALA;
```

- **użytkownik ALA wykonuje:**

```
ALA> SELECT * FROM OLEK.pracownicy;
ALA> CREATE SYNONYM prac_olek FOR OLEK.pracownicy;
ALA> SELECT * FROM prac_olek;
```

Informacje o synonimach

ALL_SYNONYMS	**Synonimy publiczne i prywatne dostępne dla użytkownika**
USER_SYNONYMS	**Synonimy prywatne, będące własnością użytkownika**

```
SELECT synonym_name, table_name
FROM user_synonyms;
```


Podsumowanie

- **Użytkownik zostaje uwierzytelniony przed przyłączeniem do bazy danych.**
- **Autoryzacja jest procesem weryfikacji uprawnień użytkownika do wykonania określonej operacji w bazie danych.**
- **Przywilej to prawo wykonywania przez użytkownika określonej akcji w bazie danych lub dostępu do określonego obiektu bazy danych.**
- **Rola to nazwany zbiór przywilejów, ułatwiający zarządzanie systemem autoryzacji.**
- **Synonim jest alternatywną nazwą obiektu w bazie danych.**