



UNIVERZITET U NIŠU
ELEKTRONSKI FAKULTET



Marko Đorđević

Sistemi za upravljanje bazama podataka – Seminarski rad

Tema: Sigurnost Oracle baze podataka

Profesor:

Doc. dr Aleksandar Stanimirović

Student:

Marko Đorđević, br. ind. 1168

Niš, Maj 2021.

Sadržaj

1. Uvod	3
2. Sigurnost Oracle baze podataka.....	4
2.1 Bezbednost korisnika.....	5
2.1.1. Kreiranje korisničkih naloga.....	7
2.1.2. Izmena korisničkog naloga	10
2.1.3. Brisanje korisničkih naloga.....	11
2.2. Autentifikacija.....	12
2.2.1. Autentifikacija administratora baze podataka.....	13
2.2.2. Autentifikacija korisnika baze podataka	14
2.2.3. Kerberos i CyberSafe način autentifikacije	15
2.2.4. Autentifikacija korišćenjem lozinke	15
2.2.5. Postavke lozinke u podrazumevanom profilu.....	16
2.2.6. Životni ciklus lozinke	17
2.2.7. Primer promene lozinke korisnika korišćenjem ugrađene funkcije za verifikaciju korisničkih lozinki	18
2.3. Autorizacija korisnika	19
2.3.1. Konfiguracija privilegija i autorizacija uloga	19
2.3.2. Kreiranje profila korisnika.....	20
2.3.3. Dodeljivanje privilegija korisnicima.....	24
2.3.4. Kreiranje uloga (eng. Role).....	27
3. Zaključak	30
4. Literatura.....	31

1. Uvod

Sigurnost baza podataka je važna komponenta posla administratora baza podataka, iz razloga što je sama količina osjetljivih informacija smeštena u tim bazama velika i što o njoj često zavisi jako veliki broj ljudi. Bez detaljnog i sveobuhvatnog plana i primene bezbednosnih mera, integritete baze podataka će se dovesti u pitanje. Svaki administrator mora poznavati bezbednosne mehanizme u svom okruženju da bi obezbedio da samo ovlašćeni korisnici pristupaju podacima u bazi i vrše njihova ažuriranja.

Kako su se razvijali DBMS-ovi (eng. Data Base Management System), tako je postojala sve veća potreba za njihovom sigurnošću. Svakom novom verzijom dodavane su brojne sigurnosne opcije kao i sigurnosne nadogradnje koje bi uklonile ranjivost. Sve većom upotrebom Interneta javio se imperativ osiguravanja baza podataka odnosno sigurnost baze podataka. Ove mere bezbednosti težile su da obezbede tajnost, nepromenljivost i dostupnost podataka ovlašćenim licima.

Sigurnost baze podataka predstavlja sistem procesa i postupaka kojima se štiti baza podataka od neželjenih aktivnosti. Pod neželjenim aktivnostima se može smatrati zloupotreba pravog korisnika, zlonamerni napadi ili greške izazvane nepažnjom a koju je načinio korisnik sistema ili proces. Kod zaštite informacija postoje tri ključna elementa:

1. Poverljivost – zaštita da je informacija dostupna samo onima koji imaju ovlašćeni pristup traženoj informaciji (zaštita podataka od neovlašćenog čitanja)
2. Integritet – zaštita postojanja, tačnosti i kompletnosti informacije kao i procesnih metoda (zaštita od nedozvoljenog pristupa podacima)
3. Dostupnost – zaštita da samo autorizovani korisnici imaju mogućnost pristupa informaciji i uslugama.

Najosnovija metoda zaštite osjetljivih informacije koje se čuvaju u bazi podataka je ograničenje pristupa podacima samo određenoj grupi korisnika. Na ovaj način se osigurava poverljivost podataka. Kontrola pristupa se može ostvariti na dva načina:

1. Autentifikacijom – predstavlja proces identifikovanja i verifikovanja identiteta
2. Autorizacijom – određuje ograničenja nad podacima unutar baze podataka za određenog korisnika.

Oracle baza podataka pruža bogat skup bezbednosnih funkcija za upravljanje korisničkim nalogima, potvrdu identiteta, privilegije, sigurnost aplikacije, šifrovanje, mrežni saobraćaj i reviziju.

U ovom seminarskom radu biće obrađena tema sigurnost Oracle baze podataka i na praktičnom primeru biće pokazan kako Oracle baza podataka upravlja korisničkim nalogima, na koji način autentifikuje i autorizuje korisnike na sam sistem.

2. Sigurnost Oracle baze podataka

Oracle baza podataka nudi mogućnost konfiguracije bezbednosti na sledećim nivoima bezbednosti :

1. Nivo korisničkih naloga: Nakon kreiranja korisničkih naloga, korisničke naloge možemo zaštititi na različite načine. Možemo kreirati profile lozinke i ograničenost dostupnosti resursa za korisnički nalog. Oracle pruža veliki skup predefinisanih korisničkih naloga za administrativne, neadministrativne i primere šeme.

2. Metode autentifikacije: Oracle baza podataka pruža nekoliko načina za konfigurisanje potvrde identiteta korisnika i administratora baze podataka.

3. Nivo privilegije i uloge: Ovaj tip ograničenja se koristi da bi se ograničio korisnički pristup podacima-autorizacija.

4. Sigurnost aplikacije: Prvi korak u kreiranju aplikacije baze podataka je osigurati da smo na pravi način ugradili sigurnost aplikacije u svoje politike bezbednosti aplikacije.

5. Informacija o korisničkoj sesiji pomoću koneksta aplikacije: Kontekst aplikacije je par ime-vrednost koji sadrži informacije o sesiji. Možemo preuzeti informacije o sesiji o korisniku, poput korisničkog imena ili terminala, i da tom korisniku ograničimo pristup bazi podataka i aplikacijama na osnovu tih informacija.

6. Pristup bazi podataka na nivou reda i kolone pomoću virtuelne baze podataka. Politika virtuelne baze podataka dinamički je ugrađen u klauzuli WHERE u SQL izrazu koju korisnik izdaje.

7. Klasifikacija i zaštita podataka u različitim kategorijama: Možemo stvoriti politike transparentne osetljive zaštite podataka da bismo pronašli sve kolone tabele u bazi podataka koji sadrže osetljive podatke, klasifikovanje podataka, a zatim kreirali politiku koja ove podatke štiti u celini za datu klasu.

8. Mrežno šifrovanje podataka. PL/SQL paket DBMS_CCRYPTO možemo koristiti za šifrovanje podataka tokom mrežne komunikacije da bismo sprečili neovlašćeni pristup tim podacima.

9. Konfiguracija tanke JDBC (eng. Java Database Connectivity Client) mreže klijenata radi sigurnosnog povezivanja sa Oracle bazama podataka.

10. Konfiguracija snažne autentifikacije. Možemo konfigurisati svoje baze podataka tako da koriste snažnu potvrdu identiteta pomoću Oracle adaptera za potvrdu identiteta koji podržavaju različite usluge provere identiteta. Oracle baza podataka pruža sledeću snažnu podršku identiteta: Centralizovana potvrda identiteta i jedinstvena prijava, Kerberos, Udaljena korisnička usluga za autentifikaciju, SSL provera identiteta.

11. Revizija aktivnosti baze podataka: Možemo revidirati aktivnosti baze podataka u opštim crtama, poput revizije svih SQL izraza, SQL privilegija, objekata šeme i mrežne aktivnosti. Možemo izvršiti reviziju na detaljan način, na primer kada se koriste IP adrese izvan kooperativne mreže.

2.1 Bezbednost korisnika

Korisničke naloge možemo osigurati pomoću jakih lozinki i određivanjem posebnih ograničenja za korisnike. Svaka Oracle baza podataka (CDB i PDB) ima listu važećih korisnika baze podataka. Da bi pristupio CDB-u ili PDB-u, korisnik mora pokrenuti aplikaciju baze podataka i povezati se sa instancom baze podataka koristeći važeće korisničko ime definisano u bazi podataka. Kada kreiramo korisničke naloge, možemo da odredimo ograničenja za svaki kreirani korisnički nalog. Takođe možemo postaviti ograničenja na količinu različitih sistemskih resursa dostupnih svakom korisniku kao deo bezbednosnog domena tog korisnika. Oracle baza podataka pruža skup prikaza baze podataka koje možemo pretražiti da bi pronašli informaciju kao što su resursi i informacije o sesiji. Dostupni su i profili. Profil u Oracle bazi podataka predstavlja kolekciju atributa koji se odnose na korisnika. Omogućava jednu referentnu tačku za bilo kog od više korisnika koji dele iste atribute.

Oracle baza podataka pruža skup unapred definisanih administrativnih, neadministrativnih i jednostavnih šema korisničkih naloga. Status svih naloga u bazi podataka, kao i korisnička imena čuvaju se u posebnoj tabeli *DBA_USERS*.

Primer pretraživanja korisničkih naloga i statusa korisničkih naloga na sistemu je sledeći:

```
SELECT username, account_status FROM DBA_USERS;
```

Nakon izvršenja ovakvog upita, rezultat će biti prikazan kao na slici 1.

	USERNAME	ACCOUNT_STATUS
1	SYS	OPEN
2	SYSTEM	OPEN
3	XS\$NULL	EXPIRED & LOCKED
4	TESTUSER	EXPIRED
5	TESTUSER1	OPEN
6	LBACSYS	LOCKED
7	OUTLN	EXPIRED & LOCKED
8	TESTUSER12	OPEN
9	DBSNMP	EXPIRED & LOCKED
10	APPQOSSYS	EXPIRED & LOCKED
11	DBSFUSER	EXPIRED & LOCKED
12	GGSYS	EXPIRED & LOCKED
13	ANONYMOUS	EXPIRED & LOCKED
14	HR	EXPIRED & LOCKED
15	CTXSYS	EXPIRED & LOCKED
16	DVSY	LOCKED
17	SI_INFORMTN_SCHEMA	EXPIRED & LOCKED

Slika 1: Rezultat upita koji pretražuje korisničke naloge sistema i prikazuje status korisničkog naloga

Sa slike 1 možemo videti deo korisničkih naloga napravljenih za određenu bazu podataka. Pored sistemskih korisničkih naloga sa slike možemo videti i korisnički kreirane naloge, testuser, testuser1. Korisnički nalozi SYS i SYSTEM kreiraju se nakon kreiranja Oracle baze podataka.

SYS korisnički nalog poseduje DBA ulogu. DBA je standardna uloga koju administrator može dodeliti drugom administratoru. Obuhvata sve sistemske privilegije i treba ga dodeliti samo korisnicima kojima najviše veruju i koji su kvalifikovani. Dodeljivanje ove uloge korisniku omogućava da upravlja bazom podataka. Sve osnovne tabele i pogledi se čuvaju u šemi SYS. Ove tabele i prikazi su presudni za rad Oracle baze podataka, tabelama u SYS šemi manipuliše samo baza podataka. Korisnik ili administrator baze podataka ih nikada ne bi smeo menjati.

SYSTEM korisnički nalog, takođe poseduje DBA ulogu. Korisnik SYSTEM može da kreira dodatne tabele i poglede koji prikazuju administrativne informacije, kao interne tabele i poglede koje koristi razne opcije i alati Oracle baze podataka. Unapred definisana DBA uloga automatski se kreira pri svakoj instalaciji baze podataka. Ova uloga sadrži većinu privilegija sistema baze podataka. Razlika između SYS i SYSTEM korisničkog naloga je ta, što SYSTEM korisnički nalog ne može izvršiti opravak i backup baze podataka kao i ažuriranje baze podataka.

Sa slike 1 možemo videti da svaki korisnički nalog ima različiti status naloga. Kod Oracle baze podataka postoje sledeći statusi korisničkih naloga:

1. OPEN – Korisnički nalog je dostupan
2. EXPIRED- Korisnički nalog je „istekao“. Pristup bazi podataka je odbijen
3. EXPIRED(GRACE) – Korisnički nalog je „istekao“, ali je i dalje dostupan. Dobićemo X dana da promenimo lozinku kako bismo promenili status korisničkog naloga na OPEN.
4. LOCKED(TIMED) – Korisnički nalog je zaključan zbog neuspelog pokušaja prijave.
5. LOCKED – Korisnički nalog je zaključan od strane administratora
6. EXPIRED & LOCKED(TIMED)
7. EXPIRED(GRACE) & LOCKED(TIMED)
8. EXPIRED & LOCKED
9. EXPIRED(GRACE) & LOCKED

2.1.1.Kreiranje korisničkih naloga

Podaci o korisnicima sistema čuvaju se u tabeli DBA_USERS, naredba koja pokazuje sve korisničke naloge i tipove autentifikacije je sledeća:

```
SELECT username,authentication_type FROM DBA_USERS;
```

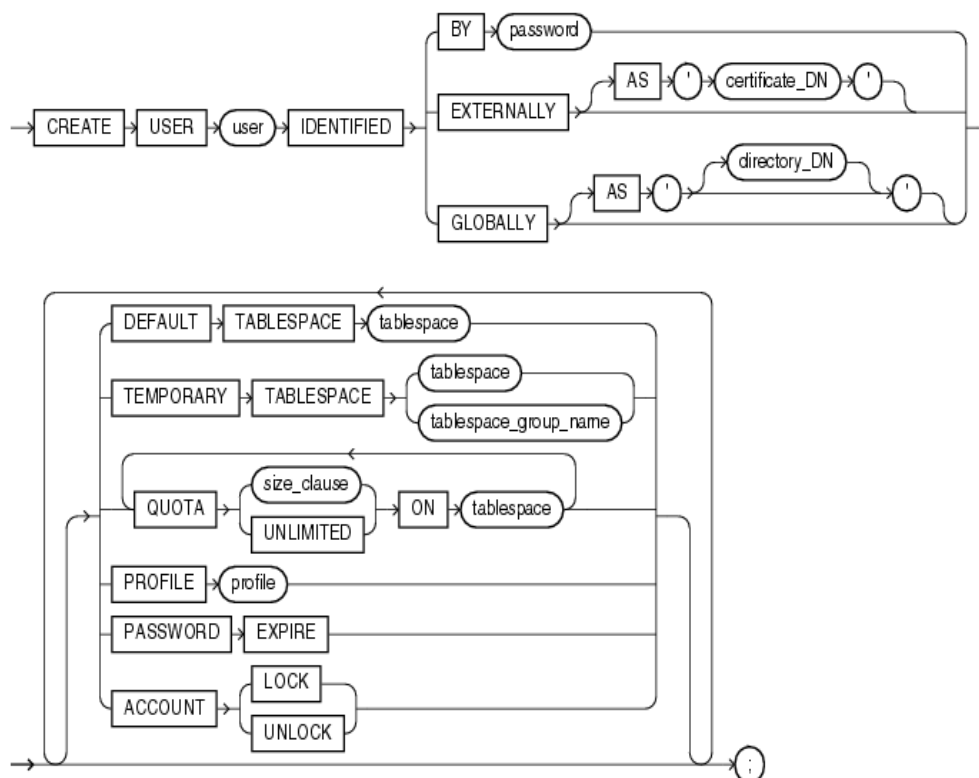
Rezultat izvršenja naredbe je prikazano na slici 2:

USERNAME	AUTHENTICATION_TYPE
1 SYS	PASSWORD
2 SYSTEM	PASSWORD
3 XS\$NULL	PASSWORD
4 TESTUSER	PASSWORD
5 TESTUSER1	PASSWORD
6 LBACSYS	NONE
7 OUTLN	PASSWORD
8 TESTUSER12	PASSWORD
9 DBSNMP	PASSWORD
10 APPQOSSYS	PASSWORD
11 DBSFUSER	PASSWORD
12 GGSYS	PASSWORD
13 ANONYMOUS	PASSWORD
14 HR	PASSWORD
15 CTXSYS	PASSWORD
16 DVSYS	NONE
17 SI_INFORMTN_SCHEMA	PASSWORD
18 DVF	NONE

Slika 2: Rezultat izvršenja naredbe koja pokazuje sve korisničke naloge i tipove autentifikacije

Da bi mogli da kreiramo korisničke naloge potrebno je imati dodeljenu privilegiju CREATE USER. Budući da je CREATE USER sistemska privilegija, administrator baze podataka ili sistem administrator obično je jedini korisnik koji ima ovu privilegiju. Ukoliko postoji potreba da se nekom korisniku dodeli ova sistemska privilegija neophodno je uz klauzulu CREATE USER pridodati klauzulu WITH ADMIN OPTION.

Sintaksa naredbe za krciranje korisničkog naloga data je na slici 3.



Slika 3: Sintaksa CREATE USER naredbe

Primer jedne pravilno napisane naredbe za kreiranje korisnika je sledeća:

```
CREATE USER korisnik1
IDENTIFIED BY korisnickalozinka1
DEFAULT TABLESPACE system
QUOTA 10M ON system
TEMPORARY TABLESPACE temp
QUOTA 5M ON system ;
```

Nakon klauzule CREATE USER stoji ime korisnika koje mora biti jedinstveno u celom sistemu. Ukoliko se ne navede jedinstveno ime, baza podataka će obavestiti korisnika o problemu vezanim za ime korisnika. Lista svih korisničkih imena baze podataka moguće je videti izvršenjem sledećeg upita:

```
SELECT username FROM DBA_USERS;
```


Sva imena korisničkih naloga u bazi podataka sačuvana su velikim slovima, ukoliko postoji potreba za drugačijim definisanjem, neophodno je ime korisnika staviti pod navodnicima, tada će biti sačuvano u formatu koji korisnik zahteva.

Kaluzula IDENTIFIED u CREATE USER naredbi dodeljuje korisniku lozinku. U ovom primeru, korisnička lozinka je *korisnickalozinka1*.

Klauzula DEFAULT TABLESPACE definiše tablespace za datog korisnika, odnosno definiše prostor tabela za datog korisnika. Svaki korisnik treba da ima zadati prostor tabela. Ukoliko korisnik izvršava DDL naredbe bez navođenja prostora tabela tada će se objekti skladištiti u korisničko definisanom prostoru tabela. DEFAULT TABLESPACE omogućava odvajanje korisničkih podataka od sistemskih podataka kao što su podaci koji su uskladišteni u prostoru tabela SYSTEM. Dobra praksa je ne kreirati korisnički nalog čiji je prostor tabela *system* već potrebno je kreirati sopstveni prostor tabela za tog korisnika. U našem slučaju kao primer stavljen je da korisnik ima pristup prostoru tabela sistema, ukoliko želimo da promenimo prostor tabela neophodno je izvršiti komandu ALTER USER čime ćemo specificirati na koji prostor tabela se referencira korisnik. Komanda za promenu korisničkog prostora tabela je sledeća :

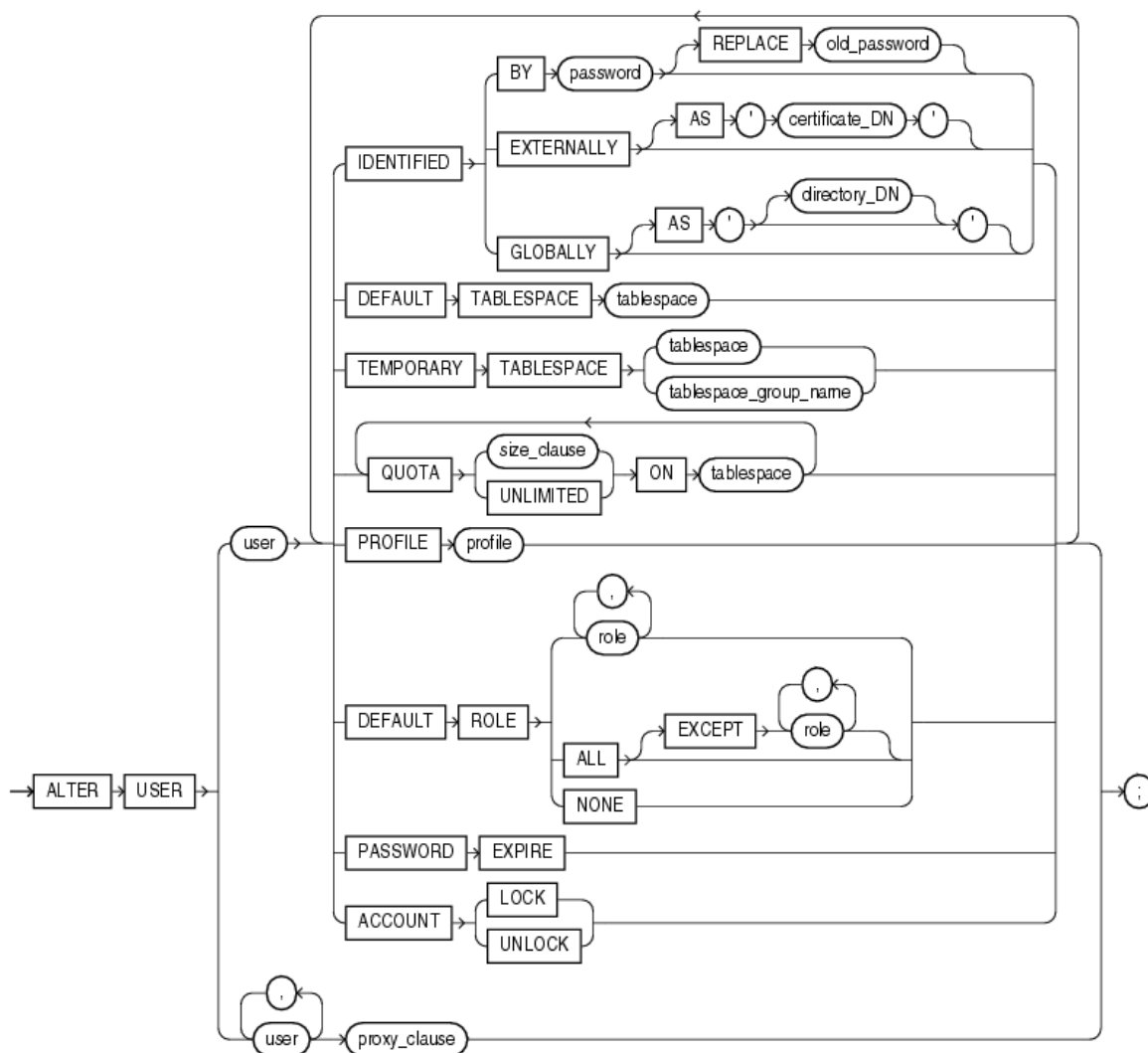
`ALTER USER korisnik1 DEFAULT TABLESPACE tbs1;`

Nakon kreiranja prostora tabela korisnika, potrebno je i definisati veličinu memorije za smeštanje korisničkih podataka. Veličina prostora tabela zadaje se klauzulom QUOTA iza čega se navodi veličina prostora i nakon toga se navodi na koji prostor tabela se odnosi (ON <ime_default_tablespace-a>). Korisniku se može dodeliti određena veličina memorije za bilo koji prostor tabela osim za privremeni prostor tabela. Maksimalna količina dodeljenog prostora tabela može iznositi 2TB, a ukoliko je postoji potreba za većim prostorom potrebno je specificirati vrednost na UNLIMITED za klauzulu QUOTA.

Klauzula TEMPORARY TABLESPACE sadrži prostor tabela za takozvane prolazne podatke koje korisnik koristi u okviru sesije. Svaki korisnik bi trebao da sadrži privremeni prostor tabela. Korisniku se može dodeliti posebno kreirani TEMPORARY TABLESPACE. Ako se korisniku ne dodeli privremeni prostor tabela Oracle baza podataka dodeljuje korisniku zadani prostor tabela koji je naveden prilikom kreiranja baze podataka. Ukoliko ne postoji podrazumevani prostor tabela tada je podrazumevani prostor tabela SYSTEM.

2.1.2. Izmena korisničkog naloga

Ukoliko želimo da izmenimo stavke korisničkog naloga, to može uraditi korišćenjem naredbe ALTER USER koja je takođe naredba koja ima sistemsku privilegiju. Sintaksa naredbe data je na slici 4.



Slika 4: Sintaksa ALTER USER naredbe

Ovom naredbom možemo za korisnika „user“ promeniti način autentifikacije na sistem, promeniti lozinku na primer, možemo promeniti prostor tabela, profil, i tako dalje.

Vrlo bitno je naglasiti da promena bezbednosnih stavki korisnika utiče na buduće sesije korisnika a ne na trenutnu sesiju korisnika. Naredbu ALTER USER mogu izvršavati samo korisnici koji imaju sistemsku privilegiju. Administratori sistema su obično jedini korisnici koji imaju ovu sistemsku privilegiju, jer omogućava modifikovanje bilo kojeg korisničkog bezbednosnog domena.

2.1.3. Brisanje korisničkih naloga

Brisanje korisničkih naloga je moguće ukoliko korisnik nije ulogovan na sistem, ako je otvorena sesija i ako korisnik sadrži objekte šeme baza podataka u okviru sesije. Brisanje korisničkih naloga je moguće ukoliko korisnički nalog poseduje privilegiju za brisanje korisnika (DROP USER). Nakon brisanja korisničkog naloga biće obrisani svi podaci vezani za korisnički nalog. Naredba za brisanje korisničkog naloga data je na slici 5:



Slika 5: Sintaksa DROP USER naredbe

Klauzula CASCADE u sql naredbi drop user označava brisanje svih korisničkih objekata šeme pre nego što se obriše sam korisnik. Ovo je obavezna klauzula ukoliko brišemo korisnika čija šema sadrži bilo kakve objekte.

2.2. Autentifikacija

Autentifikacija znači verifikovanje identiteta korisnika ili drugih entiteta koji se povezuju sa bazom podataka. Potvrđivanjem ovog identiteta uspostavlja se odnos poverenja za dalje interakcije. Autentifikacija takođe omogućava odgovornost omogućavanjem povezivanja pristupa i radnji sa određenim identitetima. Nakon autentifikacije, procesi autorizacije mogu dozvoliti ili ograničiti nivoe pristupa i radnje dozvoljene tom entitetu.

Možemo potvrditi identitet korisnika baze podataka i korisnika koji ne postoji u bazi podataka za Oracle bazu podataka. Radi jednostavnosti, isti metod potvrde identiteta se obično koristi za sve korisnike baze podataka, ali Oracle baza podataka omogućava jednoj instanci baze podataka da koristi bilo koji ili sve metode. Oracle baza podataka zahteva posebne postupke potvrde identiteta za administratore baze podataka, jer oni izvršavaju posebne operacije baze podataka. Oracle Database takođe šifrira lozinke tokom prenosa kako bi osigurao sigurnost mrežne autentifikacije.

Da bismo potvrdili identitet, odnosno autentifikovali korisnika, baze podataka i sprečili neovlašćenu upotrebu korisničkog imena baze podataka, Oracle nuudi sledeće metode autentifikacije:

1. Autentifikacija korisnika od strane operativnog sistema – Neki operativni sistemi dozvoljavaju Oracle bazi podataka da koriste informacije koje se održavaju na autentifikaciju korisnika. Ovakav tip autentifikacije ima prednost tako što jednom kada ih operativni sistem potvrdi, korisnici se mogu lakše povezati sa Oracle bazom podataka, bez navođenjem korisničkog imena ili lozinke. Primer ovakve autentifikacije jeste autentifikovanje samog administratora sistema. Ukoliko je Oracle baza podataka konfigurisana na određenom uređaju tada je automatski dodeljena autentifikacija za datog korisnika uređaja i samim tim korisnik prilikom logovanja na sistem je dovoljno da u komadnoj liniji izvrši sledeću komandu koja će ga automatski autentifikovati na Oracle sistem kao administrator.

SQLPLUS /

Sa kontrolom nad autentifikacijom korisnika od strane operativnog sistema, Oracle ne mora da skladišti i upravlja korisničkim lozinkama, iako i dalje održava korisnička imena u bazi podataka. Ukoliko se operativni sistem koristi za autentifikaciju korisnika baze podataka, upravljanje distribuiranim okruženjima baze podataka i vezama do baze podataka zahteva posebnu pažnju.

2. Autentifikacija korisnika koristeći autentičnost putem mreže – Provera autentičnosti putem mreže upravlja SSL protokol ili nezavisne usluge kao što su na primer Kerberos, PKI-Based, Radius ili usluge zasnovane na direktorijumu.

3. Autentifikacija korisnika putem Oracle baze podataka – Oracle baze podataka mogu potvrditi identitet korisnika koji pokušavaju da se povežu na samu bazu podataka, koristeći informacije smeštene u samoj bazi podataka. Da bismo koristili ovakav način autentifikacije potrebno je kreirati, za svakog korisnika, poseban korisnički nalog sa pridruženom lozinkom. Korisnik mora da navede kreirano korisničko ime i lozinku kada

pokušava da uspostavi vezu. Ovaj postupak sprečava neovlašćeno korišćenje baze podataka, jer će veza biti uskraćena ako korisnik navede netačnu lozinku. Oracle baza podataka čuva korisničke lozinke u tabele u šifrovanom formatu kako bi sprečio neovlašćene promene lozinke u bilo kom trenutku.

4. Više nivooska autentifikacija i autorizacija (eng. Multitier Authentication and Authorization) – U okruženju sa više nivoa, Oracle kontroliše bezbednost aplikacija srednjeg nivoa ograničavajući njihove privilegije, čuvajući identitet klijenta kroz sve slojeve.

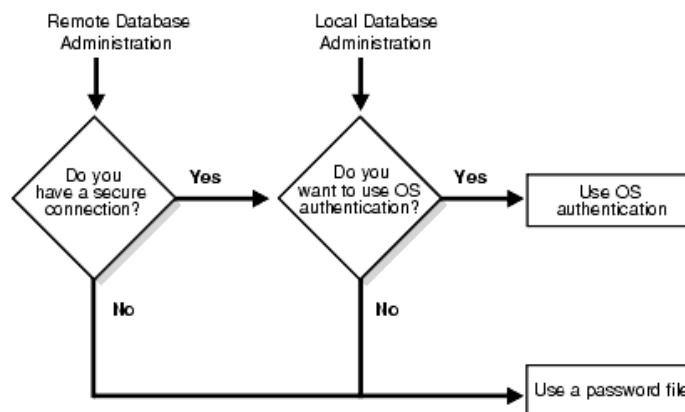
5. SSL autentifikacija – autentifikacija korišćenjem SSL protokola

6. Autentifikacija administratora baze podataka

2.2.1. Autentifikacija administratora baze podataka

Kod Oracle baze podataka, autentifikaciju administratora baze podataka možemo ostvariti pomoću jake potvrde identiteta iz samog operativnog sistema ili iz baze podataka pomoću lozinke.

Slika 6 ilustruje izbore koje posedujemo za autentifikaciju administratora baze podataka. Različiti izbori se primenjuju na lokalno administriranje baze podataka (na mašini na kojoj se nalazi baza podataka) i na administriranje mnogih različitih mašina baza podataka sa jednog udaljenog klijenta.



Slika 6: Izbor prilikom autentifikacije administratora baze podataka

Provera identiteta operativnog sistema za administratora baze podataka obično uključuje uspostavljanje grupe na operativnom sistemu, dodeljivanjem DBA privilegije toj grupi, a zatim dodavanja imena osoba koje bi trebale da poseduju te privilegije u toj grupi. Kod UNIX sistema ova grupa se naziva *dba* grupa, dok kod Windows sistema se koristi SYSDBA privilegija.

Korišćenjem datoteka sa lozinkom može predstavljati sigurnosni rizik, zato dobra praksa prilikom autentifikacije administratora koristiti jake metode potvrde identiteta. Jedan od razloga zašto lozinka nije pogodna za autentifikaciju administratora je ta što je sama lozinka jako ranjiva i može se naći u bazi, takođe korisnici gotovo uvek prilikom odabira lozinke koriste samo slova i brojeve pa je samim tim lozinka „slaba“. Ukoliko se koristi način

autentifikacije korišćenjem lozinke, poželjno je primeniti već ugrađene metode koje se koriste za proveru „jačine“ same lozinke. O ovom delu, detaljnije će biti posvećena tema u nastavku.

2.2.2. Autentifikacija korisnika baze podataka

Autentifikacija korisnika baze podataka podrazumeva korišćenje informacija unutar same baze podataka za obavljanje same autentifikacije.

Oracle baza podataka može da autentifikuje korisnike koji pokušavaju da pristupe bazom podataka koristeći informacije smeštene u toj bazi podataka. Da bi konfigurisali Oracle bazu podataka za ovakav način autentifikacije korisnika, neophodno je kreirati korisnički nalog za svakog korisnika baze podataka i pridružiti im lozinku. Baza podataka generiše jednosmerno heširanje korisničke lozinke i pamti je za upotrebu prilikom navedene lozinke prilikom prijavljivanje na sistem. Da bi podržao starije klijente, Oracle baza podataka može biti konfigurisana da generiše jednosmerno heširanje korisničke lozinke koristeći niz različitih algoritama heširanja. Dobijene heš vrednosti su poznati kao verzije lozinke, koji imaju kratka imena 10G, 11G, 12G. Ovako kratka imena služe za detalje jednosmernih algoritama heširanja lozinke. 10G verzija predstavlja oznaku za raniju case-insensitive Oracle verziju lozinke, 11G predstavlja verziju lozinke bazirane na SHA-1 verziji, dok 12G predstavlja verziju lozinke bazirane na SHA-512 verziji lozinke.

Tri su prednosti upotrebe baze podataka za autentifikaciju korisnika:

1. Korisnički nalozi i sva autentifikacija kontrolišu se iz baze podataka.
2. Oracle baza podataka pruža snažne funkcije upravljanja lozinkom za poboljšanje sigurnosti prilikom upotrebe autentifikacije baze podataka.
3. Jednostavnije je administrirati kada postoje male korisničke zajednice.

Kod Oracle baze podataka informacije o autentifikaciji korisnika kao što su uloge koji korisnici imaju ili profili koje koriste nalaze se tabelama koje su prikazane u tabeli 1.

Ime tabele	Opis
DBA_PROFILES	Prikazuje informacije o profilima, uključujući njihova podešavanja i ograničenja
DBA_ROLES	Prikazuje način autentifikacije koja se koristi prilikom prijavljivanja na sistem
DBA_USERS	Pored informacija o korisničkim nalogima prikazuje na koji način se korisnički nalozi autentifikuju, verziju lozinke
DBA_USERS_WITH_DEFPWD	Prikazuje informaciju da li je lozinka korisničkog naloga podrazumevana lozinka
PROXY_USERS	Sadrži informacije koji su trenutno ovlašćeni za povezivanja na bazu podataka putem srednjeg nivoa.
V\$DBLINK	Prikazuju korisničke naloge za postojeće veze do baze podataka

V\$PWFIL	Prikazuje imena i dodeljene administratorske privilegije administratorskih korisnika koji su uključeni u datoteku lozinke, takođe navode i verzije lozinke ovih korisnika.
V\$SESSION	Prikazuje istovremene prijavljene korisnike na trenutni PDB

Tabela 1: Tabele kod Oracle baze podataka koji sadrže informacije o autentifikaciji korisnika

2.2.3. Kerberos i CyberSafe način autentifikacije

Kerberos je pouzdani nezavisni sistem za potvrdu identiteta koji je kreirao Massachusetts Institute of Technology. Na internetu se pruža besplatno.

Kerberos se oslanja na zajedničke tajne. Pretpostavlja se da je treća strana sigurna i pruža mogućnosti jedinstvene prijave, centralizovano skladištenje lozinke, autentifikaciju veze baze podataka i poboljšanu sigurnost računara. To čini putem Kerberos servera za autentifikaciju ili putem CyberSafe ActiveTrust, komercijalnog Kerberos servera za autentifikaciju.

Jednokratna prijava Kerberos pruža brojne prednosti. Sa samo jednim centralizovanim skladištem lozinke, smanjuje administrativne troškove i zahteva od korisnika da pamte samo jednu lozinku. Omogućava kontrolu mrežnog vremena pristupa, a pomoću DES šifriranja i integriteta CRC-32 osigurava od neovlašćenog pristupa i ponovne reprodukcije paketa. Dalje, omogućava trenutne veze do baze podataka korisnika. Baze podataka sa omogućenom Kerberos mogu da šire identitet klijenta u sledeću bazu podataka za korisnike Kerberos-a koji se povezuju jedinstvenom prijavom preko Kerberosa.

CyberSafe je komercijalna verzija Kerberosa, koja dodaje određene dodatne funkcije i podršku, uključujući podršku za CyberSafe ActiveTrust server. CyberSafe centralizira bezbednost i omogućava jedinstvenu prijavu. Kao i Kerberos, zasnovan je na lozinkama, ali pruža mnogo jači mehanizam za potvrdu identiteta.

2.2.4. Autentifikacija korišćenjem lozinke

Lozinke su jedan od osnovnih oblika autentifikacije. Korisnik mora da obezbedi tačnu lozinku prilikom uspostavljanja veze kako bi sprečio neovlašćeno korišćenje baze podataka. Na ovaj način korisnici koji pokušavaju da se povežu sa bazom podataka mogu da se autentifikuju korišćenjem podataka uskladištenih u toj bazi podataka. Lozinke se dodeljuju kada se kreiraju korisnici. Baza podataka može da sačuva korisničku lozinku u rečniku podataka u šifrovanom formatu. Korisnici mogu da promene lozinke u bilo kom trenutku.

Sigurnosni sistemi baze podataka koji zavise od lozinke zahtevaju da lozinke uvek budu u tajnosti. Ali, lozinke su podložne krađi, falsifikovanju i zloupotrebi.

Oracle Database pruža skup ugrađenih zaštita lozinkom dizajniranih da zaštite lozinke korisnika baze podataka.

Ove zaštite lozinkom su sledeće:

1. Šifrovanje lozinke. Oracle Database automatski i transparentno šifrira lozinke na mreži (klijent-server i server-server), koristeći Advanced Encryption Standard (AES) pre nego što ih pošalje preko mreže. Međutim, lozinka koja je navedena u SQL izrazu (poput `CREATE USER user_name IDENTIFIED BY passvord;`) se i dalje prenosi mrežom u plain tekstu u mrežnim datotekama praćenja. Iz tog razloga bi trebalo imati omogućeno šifrovanje matične mreže ili konfigurisano šifrovanje korišćenjem Secure Sockets Layer (SSL).

2. Provera složenosti lozinke. U podrazumevanoj instalaciji, Oracle Database pruža funkcije verifikacije lozinke `ora12c_verify_function` i `ora12c_strong_verify_function` kako bi se osiguralo da su nove ili promenjene lozinke dovoljno složene da spreče metod grube sile nad lozinkom.

3. Sprečavanje uspešnog dešifrovanja lozinke. Ukoliko se napadač ili korisnik baze podataka više puta prijavljuje na Oracle bazu podataka sa netačnom lozinkom, Oracle baza podataka odlaže novu prijavu za jednu sekundu. Ovakav tip zaštite smanjuje broj lozinke koje bi napadač mogao pokušati u određenom vremenskom periodu tokom napada na sistem. Neuspelo odlaganje usporava svaki dalji neuspeli pokušaj prijave povećavajući vreme potrebno za izvršavanje lozinke. Prilikom kreiranja korisničkog naloga moguće je podesiti koliko je puta dozvoljeno prijavljivanje na sistem sa nevalidnom lozinkom nakon čega će korisnički nalog biti zaključan. Ovim je Oracle obezbedio svojim korisnicima sigurnost od napada grube sile.

4. Osetljivost na mala i velika slova za lozinke. Lozinke su osetljive na mala i velika slova kod Oracle baze podataka. Oracle je ovu opciju uveo za verzije starije od 12g.

5. Heširanje lozinke pomoću verzije 12C – heširanje lozinke.

Lozinke u Oracle bazi podataka mogu biti najviše 30 bajtova. Postoje različiti načini na koje možemo da zaštitimo lozinke, od zahteva da lozinke budu „razumne“ dužine do kreiranje prilagođenih skripti za verifikaciju složenosti lozinke koje postoje ugrađene na nekoj veb lokaciji.

2.2.5. Postavke lozinke u podrazumevanom profilu

Ukoliko se korisniku dodeli podrazumevani profil, takav korisnik ne može prekoračiti inicijalna, podrazumevana ograničenja. Prikaz podrazumevanih ograničenja data su u tabeli 2.

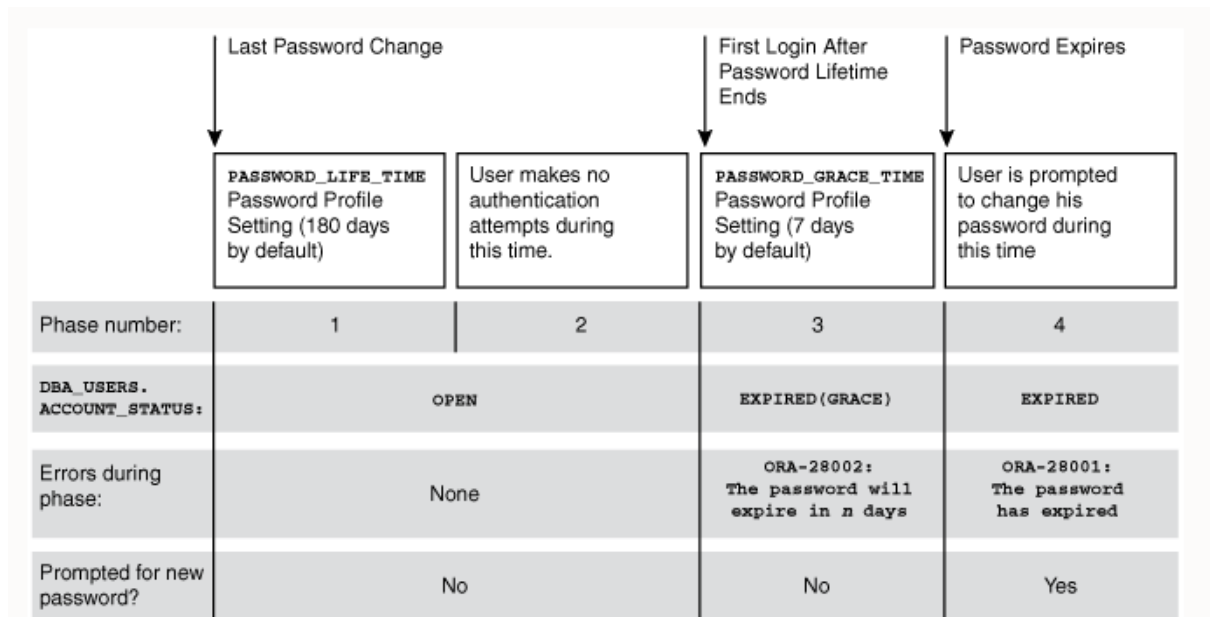
Parametar	Inicijalno podešavanje	Opis
INACTIVE_ACCOUNT_TIME	UNLIMITED	Opcija koja zaključava korisnika baze podataka koji se nije prijavio na instancu baze podataka u određenom broju dana
FAILED_LOGIN_ATTEMPTS	10	Broj neuspešnih pokušaja prijave na instancu baze podataka nakog čega se korisnik „zaključava“

PASSWORD_GRACE_TIME	7	Broj dana tokom kojih korisnik baze mora da promeni lozinku pre nego što istekne
PASSWORD_LIFE_TIME	180	Broj dana u kojima korisnik može koristiti svoju trenutnu lozinku.
PASSWORD_LOCK_TIME	1	Broj dana nakon koga je opet moguće prijaviti se na sistem ukoliko je došlo do velikog broja prijave na sistem sa pogrešnom lozinkom.
PASSWORD_REUSE_MAX	UNLIMITED	Broj promena lozinke potrebnih pre ponovne upotrebe trenutne lozinke
PASSWORD_REUSE_TIME	UNLIMITED	Broj dana pre kojih lozinka ne može biti ponovo upotrebljena.

Tabela 2 : Opis parametara lozinke kod podrazumevanog profila

2.2.6. Životni ciklus lozinke

Nakon što se kreira lozinka, lozinka prolazi kroz svoj životni ciklus i grejs period u 4 faze. Slika 7 ilustruje životni cikljus lozinke kod Oracle baze podataka



Slika 7: Životni ciklus lozinke

Faza 1: Nakon kreiranja korisničkog naloga ili promene lozinke postojećeg naloga, tada počinje životni ciklus lozinke.

Faza 2: Ova faza predstavlja vremenski period nakon završetka životnog veka lozinke, ali pre nego što se korisnik prijavi na sistem tačnom lozinkom . Oracle baza podataka ažurira status naloga tek kada se korisnik prijavi na sistem sa važećom lozinkom, u suprotnom status korisničkog naloga će ostati nepromenjen. Oracle baza podataka nema nijedan pozadinski

postupak za ažuriranje statusa naloga. Sve promene statusa naloga pokreće proces servera Oracle baze podataka u ime autentifikovanih korisnika.

Faza 3: Kada se korisnik prijavi na sistem, počinje grejs period. Oracle baza podataka ažurira kolonu DBA_USERS.EXPIRY_DATE na novu vrednost koristeći trenutno vreme plus vrednost postavke PASSWORD_GRACE_TIME iz profila lozinke naloga. U ovom trenutku korisnik dobija poruku o tome da lozinka ističe u bliskoj budućnosti, to je period u roku od 7 dana nakon koga korisnik mora da promeni lozinku. Korisnik tokom grejs perioda može da se prijavi na sistem bez ikakvih problema.

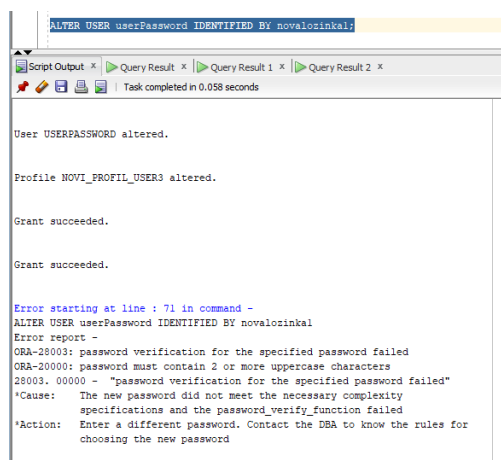
Faza 4: Po završetku grejs perioda korisniku će biti zatražena promena lozinke nakon unosa trenutne tačne lozinke pre nego što može da se nastavi autentifikacija. Ukoliko ne promeni, korisnik neće moći da se prijavi na sistem i moraće da zatraži od administratora baze podataka promenu lozinke.

2.2.7. Primer promene lozinke korisnika korišćenjem ugrađene funkcije za verifikaciju korisničkih lozinke

Kao što smo i naveli u teoretskom delu, Oracle baza podataka nudi svojim korisnicima implementiranu funkciju koja služi za verifikaciju korisničke lozinke. Da bismo dozvolili da se implementirana funkcija primenjuje za korisnički nalog, neophodno je izvršiti sledeću komandu:

```
GRANT EXECUTE ON ora12c_strong_verify_function TO userPassword;
```

Za korisnički nalog userPassword dodeljena je funkcija koja služi za verifikovanje korisničkih lozinke. Nakon izvršene komande ukoliko želimo da promenimo korisničku lozinku pokrenuće se sistemski funkcija koja će korisniku na osnovu zadate lozinke reći da li je lozinka u skladu sa politikom bezbednosti, tako na primer ukoliko lozinka ne zadovoljava politike bezbednosti korisnik će dobiti poruku o tome kako i na koji način treba kreirati svoju lozinku. Primer je prikazan na slici 8.



Slika 8: Primena funkcije za verifikaciju lozinke

2.3. Autorizacija korisnika

2.3.1. Konfiguracija privilegija i autorizacija uloga

Autorizacija dozvoljava samo određenim korisnicima pristup, obradu ili izmenu podataka. Takođe stvara ograničenja za pristup ili radnje korisnika. Ograničenja dodeljena (ili uklonjena) korisniku mogu se primeniti na objekte kao što su šeme, čitave tabele ili redovi tabele. Korisnička privilegija je pravo na pokretanje određene vrste SQL izraza ili pravo na pristup objektu koji pripada drugom korisniku, pokretanje PL / SQL paketa i tako dalje. Vrste privilegija definiše Oracle baza podataka.

Uloge (eng. Role) kreiraju korisnici (obično administratori) da bi grupirali privilegije ili druge uloge. Oni su način da se korisnicima olakša dodela višestrukih privilegija ili uloga.

Privilegije se mogu svrstati u sledeće opšte kategorije:

1. Administrativne privilegije. Administrativne privilegije dizajnirane su za uobičajene administrativne zadatke, poput izvođenja sigurnosnih kopija i oporavka. Oracle baza podataka pruža administrativne privilegije prilagođene određenim administrativnim zadacima, kao što je SYSKM administrativna privilegija za obavljanje zadataka Transparentnog šifrovanja podataka.

2. Sistemske privilegije. Sistemske privilegije omogućavaju korisnicima izvršavanje radnji nad objektima šeme. Primeri sistemskih privilegija su mogućnost kreiranja i ažuriranja tabela ili prostora tabela.

3. Uloge. Uloga grupiše nekoliko privilegija i uloga, tako da se mogu istovremeno dodeliti i opozvati od korisnika. Morate omogućiti ulogu za korisnika pre nego što je korisnik može koristiti.

4. Privilegije objekata. Svaka vrsta objekta ima privilegije povezane sa tim objektom. Objekti su objekti šeme, kao što su tabele ili indeksi.

5. Privilegije tabela. Ove privilegije omogućavaju sigurnost nad DML (jezik za manipulaciju podacima) ili DDL (jezik za definisanje podataka) operacijama. DML operacije su ALTER, INDEXES i REFERENCES operacije na tabelama. DDL operacije su DELETE, INSERT, SELECT i UPDATE operacije na tabelama i prikazima.

6. Privilegije pregleda. Privilegije DML objekta možete primeniti na poglede, slično tabelama.

7. Privilegije procedura. Procedurama, uključujući samostalne procedure i funkcije, može se dodeliti privilegija EXECUTE.

8. Privilegije tipova. Imenovanim tipovima (tipovima objekata, VARRAY-ima i ugnjeđenim tabelama) možemo dodeliti sistemske privilegije.

Prilikom dodeljivanja privilegija korisnicima potrebno je voditi računa da se samo određenim korisnicima dodeli odgovarajuća privilegija, jer prekomerno dodeljivanje nepotrebnih privilegija može ugroziti sigurnost sistema.

Privilegije korisnicima možemo zadati na dva načina:

1. Korisnicima možemo eksplicitno dodeliti privilegiju.
2. Možemo dodeliti privilegije ulozi (imenovanoj grupi privilegija), a zatim dodeliti ulogu jednom ili više korisnika.

2.3.2. Kreiranje profila korisnika

Da bismo pružili bolju sigurnost baze podataka, Oracle baza podataka nudi profile kao opciju kojom se korisniku ograničavaju pristupi resursu (autorizacija korisnika) i time se obezbeđuje sigurnost od neželjenih aktivnosti korisnika. Sintaksa naredbe za kreiranje profila je prikazana na slici 9.

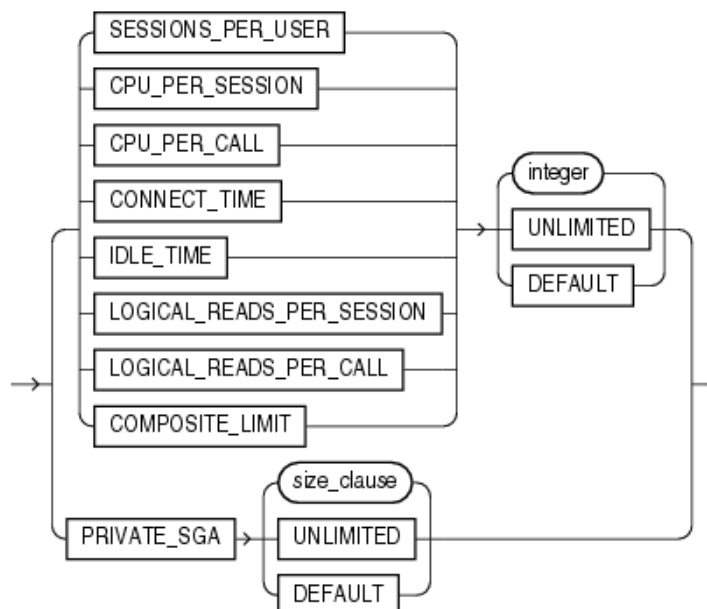
create_profile::=



Slika 9: Sintaksa naredbe za kreiranje porigla

Sintaksa **resource_parameters** klauzule je prikazana na slici 10:

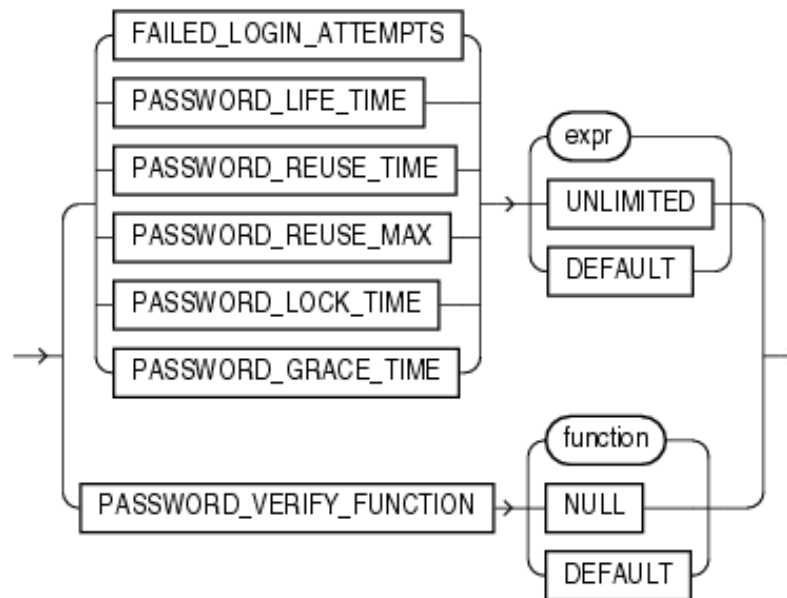
resource_parameters::=



Slika 10: Sintaksa klauzule **resource_parameters**

Sintaksa klauzule **password_parameters** je prikazana na slici 11.

password_parameters ::=



Slika 11: Sintaksa klauzule **password_parameters**

Inicijalno, prilikom kreiranja korisnika, ukoliko se ne navede profil korisnika prilikom kreiranja korisničkog naloga, biće dodeljen inicijalni parametar čiji parametri su prikazani na slici 12.

	RESOURCE_NAME	LIMIT
1	COMPOSITE_LIMIT	UNLIMITED
2	SESSIONS_PER_USER	UNLIMITED
3	CPU_PER_SESSION	UNLIMITED
4	CPU_PER_CALL	UNLIMITED
5	LOGICAL_READS_PER_SESSION	UNLIMITED
6	LOGICAL_READS_PER_CALL	UNLIMITED
7	IDLE_TIME	UNLIMITED
8	CONNECT_TIME	UNLIMITED
9	PRIVATE_SGA	UNLIMITED
10	FAILED_LOGIN_ATTEMPTS	10
11	PASSWORD_LIFE_TIME	180
12	PASSWORD_REUSE_TIME	UNLIMITED
13	PASSWORD_REUSE_MAX	UNLIMITED
14	PASSWORD_VERIFY_FUNCTION	NULL
15	PASSWORD_LOCK_TIME	1
16	PASSWORD_GRACE_TIME	7
17	INACTIVE_ACCOUNT_TIME	UNLIMITED

Slika 12: Inicijalne vrednosti profila korisnika

Ukoliko je naveden vrednost nekog parametara na UNLIMITED to označava da korisnik kome je dodeljen ovaj profil može da koristi neograničenu količinu resursa.

Opis svih parametara koje možemo definisati prilikom kreiranja profila, prikazan je u tabeli 3.

Ime klauzule	Naziv parametara	Opis parametara
Resource_parameters	SESSIONS_PER_USER	Navodimo broj istovremenih sesija na koje ograničavamo korisnika
Resource_parameters	CPU_PER_SESSION	Vremensko ograničenje procesora za vreme trajanja sesije izražene u stotinkama
Resource_parameters	CPU_PER_CALL	Vremensko ograničenje procesora u toku poziva, izraženo u stotinkama
Resource_parameters	CONNECT_TIME	Definiše ukupno proteklo vremensko ograničenje za sesiju, izraženu u minutima
Resource_parameters	IDLE_TIME	Definišemo dozvoljene periode neprekidnog neaktivnog vremena tokom sesije, izraženo u minutima. Dugotrajni upiti ili druge operacije ne podležu ovom ograničenju
Resource_parameters	LOGICAL_READS_PER_SESSION	Definiše broj blokova podataka pročitanih u sesiji, uključujući blokove pročitane iz memorije i diska.
Resource_parameters	LOGICAL_READS_PER_CALL	Definiše dozvoljeni broj blokova podataka koji se čitaju za poziv za obradu sql izraza.
Resource_parameters	PRIVATE_SGA	Definiše količinu privatnog prostora koji sesija može dodeliti
Resource_parameters	COMPOSITE_LIMIT	Definiše ukupne troškove resursa za sesiju, izražene u jedinicama usluge.
Password_parameters	FAILED_LOGIN_ATTEMPTS	Definiše broj neuspešnih pokušaja prijave na korisnički nalog pre zaključavanja naloga.
Password_parameters	PASSWORD_LIFE_TIME	Definišemo broj dana do kada se ista lozinka može koristiti za potvrdu identiteta.

Password_parameters	PASSWORD_REUSE_TIME	Definišemo broj dana nakon kojih lozinku je nemoguće koristiti
Password_parameters	PASSWORD_REUSE_MAX	Definišemo broj promena lozinke potrebnih pre ponovne upotrebe trenutne lozinke.
Password_parameters	PASSWORD_LOCK_TIME	Definišemo broj dana kada će nalog biti zaključen ukoliko je nalog zaključen nakon neuspese autentifikacije.
Password_parameters	PASSWORD_GRACE_TIME	Definiše se broj dana nakon početka grejs perioda tokom kog se izdaje upozorenje i dozvoljava prijava na sistem. Ukoliko se u tom periodu ne promeni lozinka, lozinka će biti nevažeća.
Password_parameters	PASSWORD_VERIFY_FUNCTION	Klauzula koja omogućava da se skripta za potvrdu složenosti lozinke prosledi kao argument naredbe.

Tabela 3: Opis parametar koji mogu biti definisani za profil korisnika

Primer kreiranje profila korisnika:

```
CREATE PROFILE novi_profil_user3 LIMIT
SESSIONS_PER_USER      UNLIMITED
CPU_PER_SESSION        UNLIMITED
CPU_PER_CALL            3000
CONNECT_TIME            45
LOGICAL_READS_PER_SESSION DEFAULT
LOGICAL_READS_PER_CALL  1000
PRIVATE_SGA             15K
COMPOSITE_LIMIT          5000000
FAILED_LOGIN_ATTEMPTS 5
PASSWORD_LIFE_TIME 60
PASSWORD_REUSE_TIME 60
PASSWORD_REUSE_MAX 5
PASSWORD_VERIFY_FUNCTION verify_function
PASSWORD_LOCK_TIME 1/24
PASSWORD_GRACE_TIME 10;
```

Pored ovih klauzula, takođe moguće je i definisati broj dana nakon koga će profil biti zaključen, to se postiže klauzulom INACTIVE_ACCOUNT_TIME <broj dana>.

Da bismo dodelili korisniku dodeli profil, neophodno je izvršiti sledeću komandu:

```
ALTER USER userPassword PROFILE novi_profil_user3;
```

Nakon uspešne dodele profila korisniku, da bi proverili ispravnost naredbe sledeća komanda daje informacije o korisnicima sistema (korisničko ime, profil i status korisničkog naloga):

```
SELECT USERNAME, PROFILE, ACCOUNT_STATUS FROM DBA_USERS;
```

Rezultat upita prikazan je na slici 13.

USERNAME	PROFILE	ACCOUNT_STATUS
SYSBACKUP	DEFAULT	EXPIRED & LOCKED
REMOTE_SCHEDULER_AGENT	DEFAULT	EXPIRED & LOCKED
PDBADMIN	DEFAULT	OPEN
GSMUSER	DEFAULT	EXPIRED & LOCKED
MINIMUMUSERACCOUNT	DEFAULT	OPEN
SYSRAC	DEFAULT	EXPIRED & LOCKED
OPS\$MARKO	DEFAULT	OPEN
OJVMSYS	DEFAULT	LOCKED
AUDSYS	DEFAULT	LOCKED
DIP	DEFAULT	EXPIRED & LOCKED
OPS\$DESKTOP-74K9VQV\marko	DEFAULT	OPEN
SYSKM	DEFAULT	EXPIRED & LOCKED
ORACLE_OCM	DEFAULT	EXPIRED & LOCKED
SYS\$UMF	DEFAULT	EXPIRED & LOCKED
marko	DEFAULT	OPEN
MARKO	DEFAULT	OPEN
PSMITH	DEFAULT	OPEN
SYSDBG	DEFAULT	EXPIRED & LOCKED
USERPASSWORD	NOVI_PROFIL_USER3	OPEN
KORISNIKI	DEFAULT	OPEN
NOVIKORISNIK	DEFAULT	OPEN

Slika 13: Rezultat upita koja prikazuje korisničke naloge i dodeljene profile

2.3.3. Dodeljivanje privilegija korisnicima

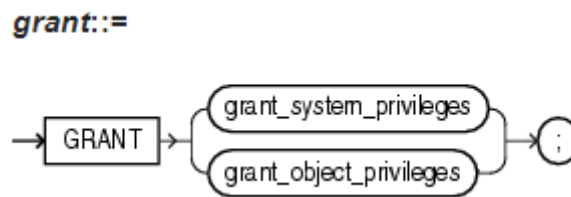
Da bismo korisnicima dodelili privilegije ili ulogama ili svim korisnicima za manipulaciju nad objektima baze podataka, koristimo GRANT SQL naredbu.

Mogu se dodeliti sledeće vrste privilegija:

1. Brisanje podataka iz određene tabele
2. Upis podataka u određenu tabelu

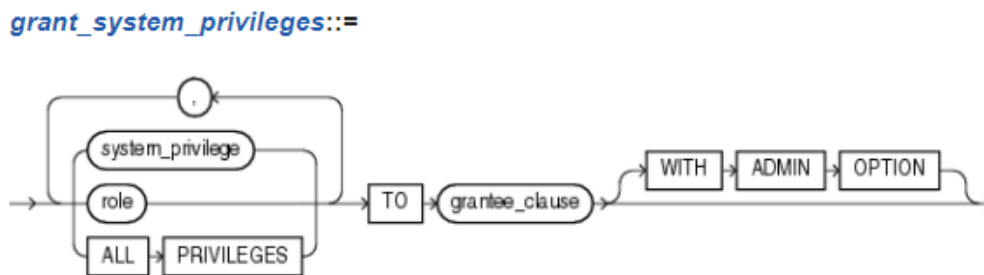
3. Kreiranje stranog ključa na imenovanu tabelu ili na podskup kolona iz tabele.
4. Prikazivanje podataka iz tabele, prikaza ili podskupa kolona u tabeli
5. Kreiranje trigera na određenu tabelu
6. Ažuriranje podataka u tabeli ili podskupu kolona u tabeli
7. Pokretanje određene funkcije ili procedure
8. Korišćenje generatora sekvence ili korisnički definisan tip

Sintaksa GRANT naredbe prikazana je na slici 13.



Slika 13: Sintaksa GRANT naredbe

Grant_system_privileges predstavlja klauzulu koju dodeljuje korisniku sistemsku privilegiju. Sintaksa *grant_system_privileges* prikazana je na slici 14.

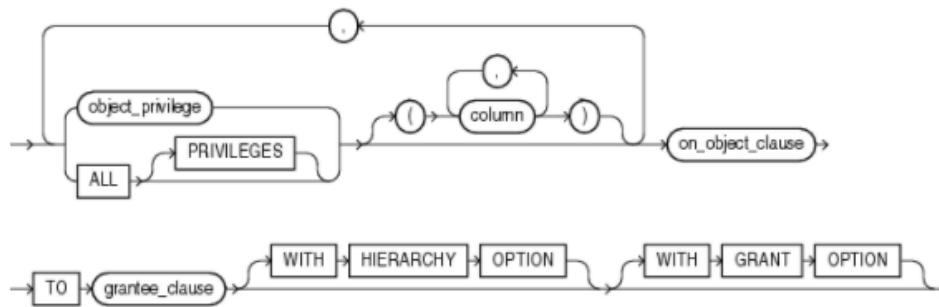


Slika 14: Sintaksa grant_system_privileges klauzule

Klauzula *system_privilege* specificira sistemske privilegije. Za svaki deo administracije baze podataka napravljena je po jedna sistemska privilegija. Ovim je Oracle pružio bezbednost baze podataka i time omogućio da korisnici imaju tačno određenu sistemsku privilegiju da bi izvršavali administrativne zadatke. Ukoliko korisnik poseduje sistemsku privilegiju ALTER DATABASE, tada je korisnik u mogućnosti da menja sadržaj baze podataka. Pored sistemskih privilegija u klauzuli moguće je dodeliti ulogu korisniku ili pak dodeliti sve privilegije. Obično se nikada korisnicima ne dodeljuje privilegija ALL PRIVILEGES jer time može biti narušena bezbednost baze podataka.

Grant_object_privileges predstavlja klauzulu koja dodeljuje korisniku sistema privilegije nad objektima. Sintaksa ove privilegije prikazana je na slici 15.

grant_object_privileges ::=



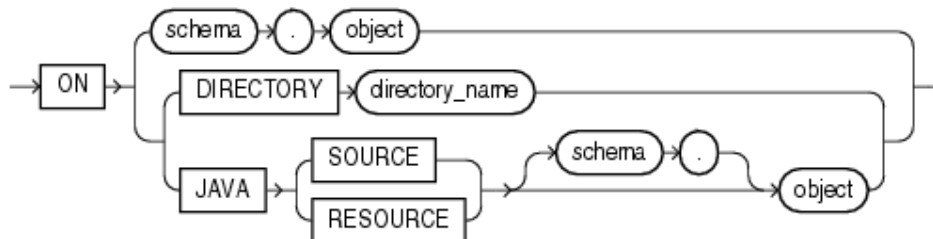
Slika 15: Sintaksa grant_object_privileges klauzule

Privilegije na objektu baze podataka može se odnositi na privilegije tabele, pogleda, sekvence, procedure, funkcije pakete , na svaki objekat baze podataka moguće je dodeliti privilegiju.

Ukoliko navedemo u naredbi WITH GRANT OPTION to će korisniku dati mogućnost da dodeljuje privilegije objektima drugih korisnika ili uloga. Ukoliko navedemo u naredbi WITH HIERARCHY OPTION, tada dodeljujemo definisanu privilegiju i svim podobjektima objekta, kao sto potpogledi kreirani u prikazu.

Sintaksa *on_object_clause* klauzule prikazana je nas slici 16.

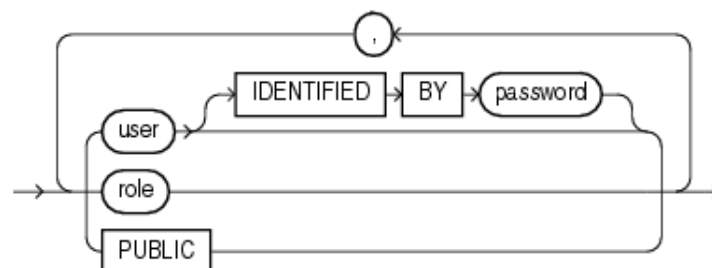
on_object_clause ::=



Slika 16: Sintaksa on_object_clause klauzule

Sintaksa *grantee_clause* klauzule je prikazana na slici 17. Ovom klauzulom specificiramo kom korisniku ili ulozi ili svim korisnicima dodeljujemo privilegiju.

grantee_clause ::=



Slika 17: Sintaksa grantee_clause klauzule

Minimalna privilegija koju korisnik može da poseduje jeste da kreira sesiju. Primer takve komande je prikazan u nastavku

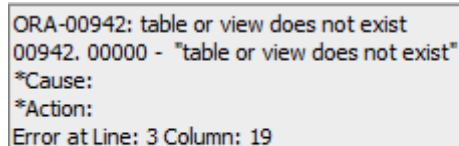
```
GRANT CREATE SESSION TO userPassword IDENTIFIED BY korisnickaLozinka;
```

Nakon izvršenja ovih naredbi korisniku je omogućeno da se prijavi na sistem korišćenjem svog imena i lozinke.

Primer dodavanja privilegija korisnika za čitanje podataka iz tabele, odnosno dodela privilegija nad objektom baze podataka je prikazan u nastavku

```
GRANT SELECT on sys.agents TO userPassword;
```

Ukoliko pokušamo da pristupimo drugim tabelama šeme baze podataka, dobićemo grešku prikazanu na slici 18.



```
ORA-00942: table or view does not exist
00942. 00000 - "table or view does not exist"
*Cause:
*Action:
Error at Line: 3 Column: 19
```

Slika 18: Rezultat pristupa drugim tabelama ukoliko korisnik nema dovoljno privilegija

Za svaku nedozvoljenu privilegiju, Oracle baza podataka će obavestiti korisnike da ili tabela ili pogled ili bilo koji objekat šeme baze podataka ne postoji, odnosno nema dovoljno privilegija. Za pristup administratorskim privilegijama će takođe biti obavešten da ne postoji dovoljno privilegija za pristup.

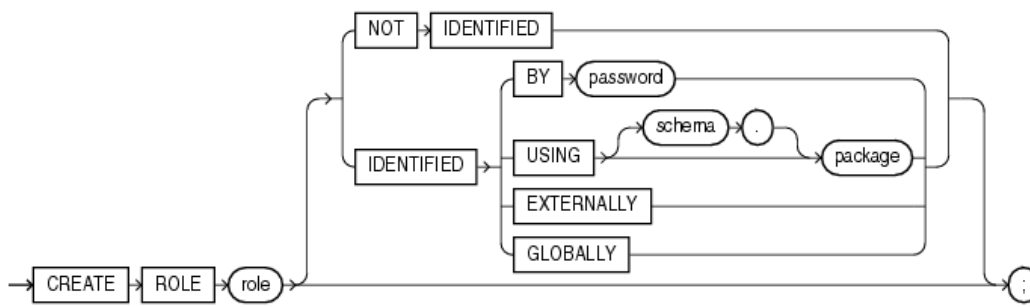
2.3.4. Kreiranje uloga (eng. Role)

Da bismo kreirali uloge, neophodno je koristiti SQL upit CREATE ROLE. Uloge se mogu dodeliti korisnicima ili drugim ulogama. Uloge možemo koristiti za administriranje privilegija baze podataka. Kao dobra praksa pokazalo se da je bezbednije dodeliti privilegije ulogama, a zatim dodeliti ulogu korisniku.

Uloga sadrži sve privilegije dodeljene ulozi i sve privilegije drugih uloga koje su joj dodeljene. Nova uloga je na početku prazna. Da bismo kreirali ulogu neophodno je imati dodeljenu sistemsku privilegiju CREATE ROLE.

Naredba za kreiranje uloga kod Oracle baze podataka data je na slici 19.

create_role::=



Slika 19: Sintaksa naredbe CREATE ROLE

Ukoliko kreiramo ulogu sa klauzulom NOT IDENTIFIED ili IDENTIFIED EXTERNALLY ili BY PASSWORD, tada Oracle baza podataka dodeljuje sa Administrativnom opcijom, ukoliko u naredbi za kreiranje uloga stoji IDENTIFIED GLOBALLY tada baza podataka neće dododeliti administrativnu opciju.

Primer kreiranja uloga prikazan je na slici 20:

```

1  create user testUserRole IDENTIFIED BY marko;
2  Grant CREATE SESSION to testUserRole IDENTIFIED by marko;
3
4
5
6  create role korisnickaRole IDENTIFIED by marko;
7
8  grant select on sys.agents to korisnickaRole;
9  grant select on sys.products to korisnickaRole;
10
11 grant korisnickaRole to testUserRole IDENTIFIED by marko;
12
13 select * from DBA_ROLES ;
14
15 select * from role_role_privs;
16

```

Slika 20: Primer kreiranje uloga

Da bismo kreirali ulogu, prvo ćemo kreirati korisnika baze podataka kao što je prikazano na slici 18 (linija 1). Da bi korisnik mogao da se prijavi na sistem neophodno mu je dodeliti privilegiju CREATE SESSION (linija 2). Nakon što smo uspešno kreirali sesiju, kreiramo ulogu korisnika naredbom CREATE ROLE (linija 6). Dodelićemo prilivegije ulozi i time sada korisnička privilegija, u našem primeru *korisnickaRole* imaće mogućnost čitanja podataka iz tabele *agents* u šemi *sys*. Nakon uspešno izvršene naredbe dodelićemo korisniku ulogu naredbom GRANT.

Kada se korisnik prijavi na sistem, on je inicijalno prijavljen na inicijalnu vrednost uloge. Da bi promenio ulogu neophodno je izvršiti komandu SET ROLE kojom se podešava korisnička uloga. Ukoliko ,prilikom kreiranja korisničke uloge smo, naveli da se korisnička uloga autentifikuje putem lozinke, tada korisnik mora navesti u naredbi SET ROLE i lozinku uloge. Primer je pokazan u nastavku:

```
set role korisnickaRole IDENTIFIED by marko;
```

Ukoliko se iz naredbe izostavi autentifikacija role, tada će korisniku se prikazati sledeća poruka, prikazana na slici 21.

```
Role KORISNICKAROLE succeeded.

Error starting at line : 3 in command -
set role korisnickaRole
Error report -
ORA-01979: missing or invalid password for role 'KORISNICKAROLE'
01979. 00000 - "missing or invalid password for role '%s'"
*Cause:      An attempt was made to enable a role without giving
              the proper password.
*Action:     Use the IDENTIFIED BY clause in SET ROLE to specify
              the correct password.
```

Slika 21: Prikaz ukoliko se izostavi autentifikacija role

Ovim Oracle svojim korisnicima pruža sigurnost u pogledu da samo autorizovani korisnici mogu pristupati dodeljenim resursima. Nakon pravilnog dodeljivanje uloge, korisnik može da izvrši bilo koji SELECT SQL upit na tabelom sys.products. Primer je prikazan na slici 22.

```
7 | select p.product_name from sys.products p where p.list_price>1800;
```

PRODUCT_NAME
1 Trek Fuel EX 8 29 - 2016
2 Trek Slash 8 27.5 - 2016
3 Trek Conduit+ - 2016
4 Surly Karate Monkey 27.5+ Frameset - 2017
5 Trek Fuel EX 9.8 29 - 2017
6 Trek Fuel EX 5 27.5 Plus - 2017
7 Trek Fuel EX 9.8 27.5 Plus - 2017
8 Trek Remedy 9.8 - 2017
9 Trek Domane SL 6 - 2017
10 Trek Silque SLR 7 Women's - 2017
11 Trek Silque SLR 8 Women's - 2017
12 Trek Domane SL Disc Frameset - 2017
13 Trek Domane S 6 - 2017
14 Trek Domane SLR 6 Disc - 2017
15 Trek Emonda S 5 - 2017
16 Trek Madone 9.2 - 2017
17 Trek Domane S 5 Disc - 2017
18 Trek Powerfly 8 FS Plus - 2017
19 Trek Boone 7 - 2017
20 Trek Boone Race Shop Limited - 2017
21 Trek Fuel EX 8 29 - 2018
22 Trek Fuel EX 7 29 - 2018
23 Surly Krampus Frameset - 2018
24 Surly Troll Frameset - 2018
25 Surly ECR 27.5 - 2018
26 Heller Bloodhound Trail - 2018
27 Heller Shagmaw GX1 - 2018

Slika 22: Primer naredbe nakon uspešne dodele uloge korisniku

3. Zaključak

Sigurnost baza podataka neiscrpna je i uvek aktuelna tema. Odnosi se na veliki broj alata, kontrola i mera dizajniranih za uspostavljanje i očuvanje poverljivosti, integriteta i dostupnosti same baze podataka. Baze podataka sadrže važne informacije o poslovanju. Podaci u njima predstavljaju sliku trenutnog stanja firme i poslovnih procesa. Upravo zbog toga vrlo je važno baze podataka zaštititi i upravljati njima na pravi način. Što je baza veća i što se više koristi, to je više podložna bezbednosnim propustima. Profesor Ros Anderson sa Kembridž univerziteta definisao je pravilo koje kaže da nije moguće napraviti bazu koja je skalabilna, funkcionalna i sigurna, jer ako se velika baza projektuje tako da joj se lako pristupa i koristi ona tako postaje nesigurna, ali ako se isprojektuje da bude neprobojna onda ju je nemoguće koristiti.

Važno je razumeti da sigurnost kao gotov proizvod ne postoji. Sigurnost je proces sa kojim nastojimo da očuvamo resurse u onom obliku i na onaj način na koji smo mi zadovoljni u našem poslovanju. Gledano sa aspekta baze to predstavlja zaštitu podataka, od bilo kog autorizovanog i neautorizovanog pristupa. Onaj korisnik koji poseduje pristup on može i da modifikuje podatke.

U ovom radu dali smo osvrt na deo bezbednosti koju nudi Oracle baza podataka svojim korisnicima. Detaljnije smo obradili teme autentifikacije i autorizacije korisnika baze podataka. Videli smo kako je moguće kreirati korisnike baze podataka, kako je moguće dodeliti im privilegije, odnosno način autorizacije korisnika. Pokazali smo načine autentifikacije korisnika i na praktičnom primeru autentifikacije korisnika korišćenjem lozinke pokazali smo koje sve opcije korisnicima nudi Oracle u pogledu zaštite korisničkih naloga i lozinki.

4. Literatura

- https://docs.oracle.com/en/database/oracle/oracle-database/21/dbseg/part_1.html