Database Security

Chapter 6

Password, Profiles, Privileges, and Roles

+

Objectives

- Define authentication and then implement with SQL Server, MySQL, and Oracle
- Define authorization and then implement with SQL Server, MySQL, and Oracle
- Manage users based on security best practices using SQL Server, MySQL, and Oracle
- Identify and apply password best practices using SQL Server, MySQL, and Oracle

Objectives (cont'd.)

- Define and create roles using SQL Server, MySQL, and Oracle
- Define, grant, deny, and revoke privileges using SQL Server, MySQL, and Oracle

Authentication

- Two main steps in controlling access to data
  - Authentication
  - Authorization
- Authentication
  - Process of confirming the identity of individuals requesting access to a secure environment
  - Done by verifying the login and credentials match those created within that environment

+ Authentication (cont’d.)

- Login
  - Object mapped to a user account within a database
  - Associated to a user by the security identifier (SID)
  - Required for authentication into the environment
  - Different from user account, which controls activities in the environment
- Default logins are created during database installation
  - Must be managed correctly

+ Authentication (cont'd.)

- Credential
  - Piece of information used to verify identity
- Examples of credentials
  - Person's username and password
  - Application's secure ID
  - Host's network name and address
- Types of credentials used to verify identity:
  - Depend on the authentication processes of a particular system or environment

Authentication (cont'd.)

- Authentication can be verified a few times and at different levels during a single logon attempt
- Third-party applications can add security to authentication process
  - They may use password encryption to keep network environment secure
- Three levels of authentication in a database environment
  - OS level, database level, third-party support

Operating System Authentication

- Credentials authenticated primarily through the OS
  - Account must reside on operating system
  - OS account credentials must be used to access the system
  - In some cases, the OS login alone can be used to authenticate users to the database
  - Advantages: convenience to the user, centralized account administration

+ Database Authentication

- User must have a local database account to check credentials and gain access
- User may be required to access several systems before reaching the database
- Challenges
  - Users must keep different credentials for different systems
  - Difficulty often leads to weak passwords and poor password practices
  - Administration is more difficult with separate accounts

+

Network or Third-Party Authentication

- Can be used for remote and physical environments
- Users not required to have an account on OS or database
  - Must have a network account recognized by the third-party application
- Types of external authentication
  - Smart card uses PIN for authentication
  - Kerberos uses symmetric-key cryptology
  - Public key infrastructure (PKI)
  - Digital certificate

Network or Third-Party Authentication (cont'd.)

- Third-party or external authentication not recommended for use alone
    - Can be combined with OS and server authentication

Database Vendor–Specific Authentication Components

- SQL Server authentication information
  - Server uses Windows Authentication and Mixed Mode Authentication
- Windows Authentication
  - Users logging in to the database must have a Windows login to access
  - Known as trusted authentication
  - Recommended authentication mode for SQL Server

Database Vendor–Specific Authentication Components (cont'd.)

- Mixed Mode Authentication
  - Allows both Windows Authentication and SQL Server authentication to be used to obtain database access
  - Most appropriate for environments with older OS and mixed OSs
  - Known as an untrusted connection
    - Not as secure as Windows Authentication
    - Protocols such as Kerberos cannot be used
- MySQL authentication information
  - Three pieces of information used to verify user's identity
    - Host name, MySQL username, password

Database Vendor–Specific Authentication Components (cont’d.)

- Oracle authentication information
  - Oracle supports many authentication options
  - Database servers, database links, and environment passwords can all be used as credentials
  - Additional security applications are available
- Advanced Security
  - Comprehensive security application
  - Encrypts information both transmitting across the network and stored within the database
  - Provides strong and proxy authentication strategies
  - Support and integrate with Kerberos, PKI, and SSL

+

Database Vendor–Specific Authentication Components (cont'd.)

- Middleware applications
  - Designed to monitor external requests for database access
- Database links
  - Feature that enhances authentication support
  - Link between two databases resulting in one logical storage unit
  - Enables applying common policies
  - Links can be public or private
  - Two authentication methods: current user and connect to user

+ Password Policies

- Most intrusions originate from a cracked or stolen password
- Password policy implementation
  - Organization’s first defense against compromised passwords
  - Can be enforced within database server application
  - More effective than written policy
- Both written and server-defined policies should be used for maximum effectiveness

+
Database-Enforced Password Policies

- Password policy options are often vendor specific
  - Most server applications share similar configuration settings
- Four password attributes can be enforced in almost every database server
  - Complexity
  - Failed attempts
  - Expired passwords
  - Password reuse

+

Written Password Policies

- Included in equipment usage agreement between an organization and its employees
  - Usage agreement must be flexible enough to be consistently enforced
  - And strict enough to ensure users abide by the policy
- Common standards likely to be included in an equipment usage agreement
  - Password discretion – not to tell their password to anyone in the organization
  - Password sharing – not to share their password with any other employee in the organization
  - Password storage – not to store

Database Vendor–Specific Password Management

- SQL Server password policy
  - Available password policy methods
    - Password complexity, password expiration, and enforcing password policy
- SQL Server password complexity requirements
  - Passwords should be unique and not include common or reserved words, or usernames
  - Length between 8 and 128 characters
  - Can include underscore(_), dollar sign($), and number sign(#)

Database Vendor–Specific Password Management (cont’d.)

- SQL Server password complexity requirements (cont’d.)
  - Must include at least one digit and one alphabetic character
  - Cannot begin with a number
- MySQL password policy
  - Administrators must rely on operating system and third-party applications (securich.com)
  - Stored in 45-bit encryption in user table
  - Passwords are case sensitive, vary in length, and can include special characters

Database Vendor–Specific Password Management (cont’d.)

- Securich password policies

-
```
+-------------------------------------------+-------+
  | PROPERTY                                  | VALUE |
  +-------------------------------------------+-------+
  | mysql_to_securich_reconciliation_in_progress | 0     | - used by the system
  | password_length                           | 10    | - set by user for password complexity checks
  | password_length_check                     | 1     | - set by user for password complexity checks
  | password_dictionary_check                 | 1     | - set by user for password complexity checks
  | password_lowercase_check                  | 1     | - set by user for password complexity checks
  | password_uppercase_check                  | 1     | - set by user for password complexity checks
  | password_number_check                     | 1     | - set by user for password complexity checks
  | password_special_character_check          | 1     | - set by user for password complexity checks
  | password_username_check                   | 1     | - set by user for password complexity checks
  | sec_mode                                  | 0     | - security mode is 0 (lenient) or 9 (strict)
  | priv_mode                                 | safe  | - privilege mode is safe in order to not loose any
privileges when syncing
  | admin_user                                | root  | - admin user set by system
  +-------------------------------------------+-------+
```

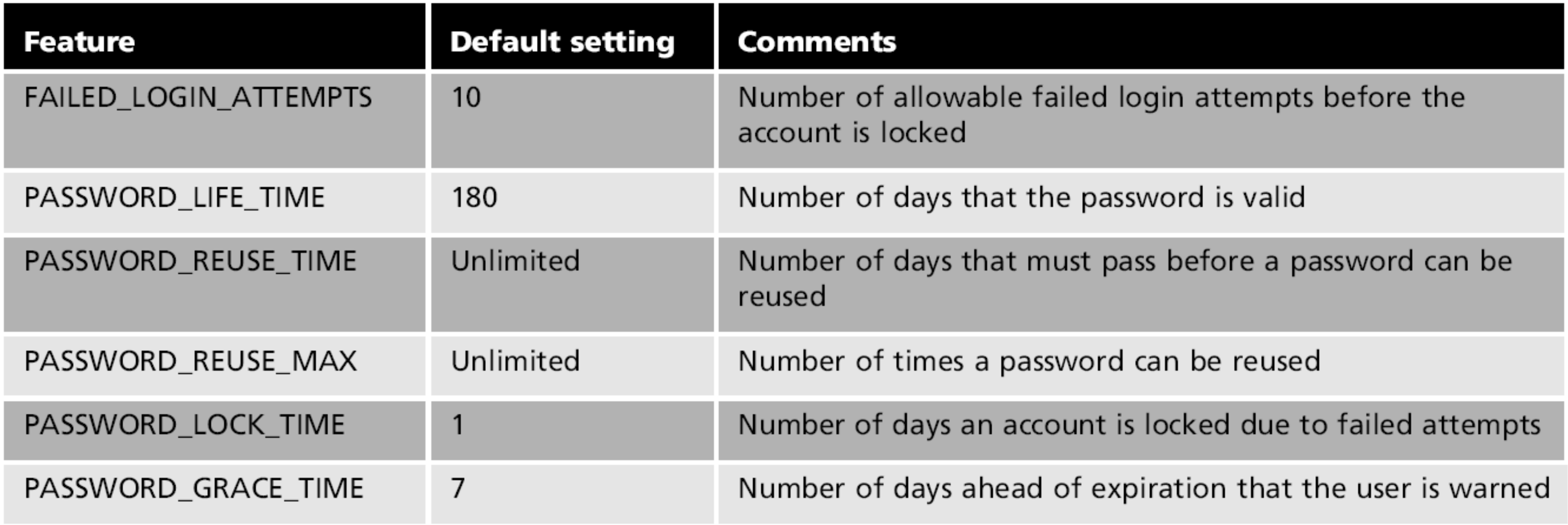
+

Database Vendor–Specific Password Management (cont’d.)

- Oracle password policy
  - Stored encrypted in DBA_USER table
  - Several built-in password protection services
    - Examples: case sensitivity, password hashing
- Oracle password complexity requirements
  - Passwords should be unique and cannot include simple words, server names, usernames, or server/usernames with numbers appended
  - Length between 8 and 128 characters

Database Vendor–Specific Password Management (cont’d.)

- Oracle password complexity requirements (cont’d.)
  - A new password must differ from previous password by at least three letters
  - Must include at least one digit and one alphabetic character
  - Cannot begin with a number
  - Can include an underscore, dollar sign, and number sign
  - Can begin with a special character or contain characters other than _, $, and #, if password is surrounded by quotation marks



Feature	Default setting	Comments
FAILED_LOGIN_ATTEMPTS	10	Number of allowable failed login attempts before the account is locked
PASSWORD_LIFE_TIME	180	Number of days that the password is valid
PASSWORD_REUSE_TIME	Unlimited	Number of days that must pass before a password can be reused
PASSWORD_REUSE_MAX	Unlimited	Number of times a password can be reused
PASSWORD_LOCK_TIME	1	Number of days an account is locked due to failed attempts
PASSWORD_GRACE_TIME	7	Number of days ahead of expiration that the user is warned

Table 6-1 Oracle password-related functions

+ Authorization

- Process of applying permissions to a user
  - Ensures users requesting access have permission to do so
- Determined prior to a user obtaining authentication credentials
- Choosing the most appropriate privileges for each user helps maintain a healthy and secure database

User Account Management

- User management tasks
  - Add, remove, and assign privileges to users
- Administrator must understand:
  - Default user accounts and privileges created during installation of database management system

+ Default User Accounts

- Default user accounts are created with predefined user access
  - True for virtually every type of database
- Most default users are the system or administration accounts
- Default passwords, usernames, rights and privileges can easily be found online
- Need to secure the default accounts to protect data

+ Default User Accounts (cont'd.)

- Default users installed with SQL Server
  - Two administrator accounts
    - SA and BUILT-IN\Administration
  - One general PUBLIC account
    - Guest
- Default users installed with MySQL
  - Two root accounts
  - Two anonymous user accounts
  - No passwords are set immediately
    - Should be assigned during installation

Default User Accounts (cont’d.)

- Default users installed with Oracle
  - Number and type of default accounts can vary greatly
    - Depend on installed options, features, and additions
  - Most accounts created to expire and be locked after installation
  - Three accounts remain open for use after installation
    - SYS – owner of base tables and base views
    - SYSMAN – for Oracle Enterprise Manager
    - SYSTEM (DBA)

Adding and Removing Users

- Always change default password of a new user
  - Or force password change prior to server entry
- Save user passwords in an encrypted file
- Enforce strong password policies
- Use different logins and passwords for different applications
- Ensure users read and agree to database usage policies
- Before removing user, perform inventory of user’s created objects

+

Adding and Removing Users (cont'd.)

- Recommended to disable a user account instead of deleting it
  - Always document removals of database user accounts
- Documentation
  - Most important component of adding or deleting accounts

+ User Privileges

- Privilege
  - Smallest unit of authorization
  - Ability to access a specific resource to perform a specific action
- Examples of privileges
  - Deleting a row
  - Creating a table
  - Executing a procedure
- Privileges should be planned out in early stages of database planning

User Privileges (cont'd.)

- Principle of least privilege
  - Security standard
  - Each user given minimum set of privileges needed to conduct legitimate business within the system
- Managing user privileges
  - Granting a privilege
  - Denying a privilege
  - Revoking a privilege
- Two ways to grant a privilege
  - Fixed and single statement

User Privileges (cont'd.)

- Assigning privileges in SQL Server
  - Three levels of permissions can be granted
    - Server-level
    - Database-level
    - Object-level
  - Can grant object permissions to individual users or roles
  - Privileges can be single statements

User Privileges (cont’d.)

- Assigning privileges in MySQL
  - Five levels of privilege
    - Global privileges
    - Database privileges
    - Table object privileges
    - Column object privileges
    - Routine privileges
  - GRANT command used to provide access to a privilege
    - Will create a new user if nonexistent

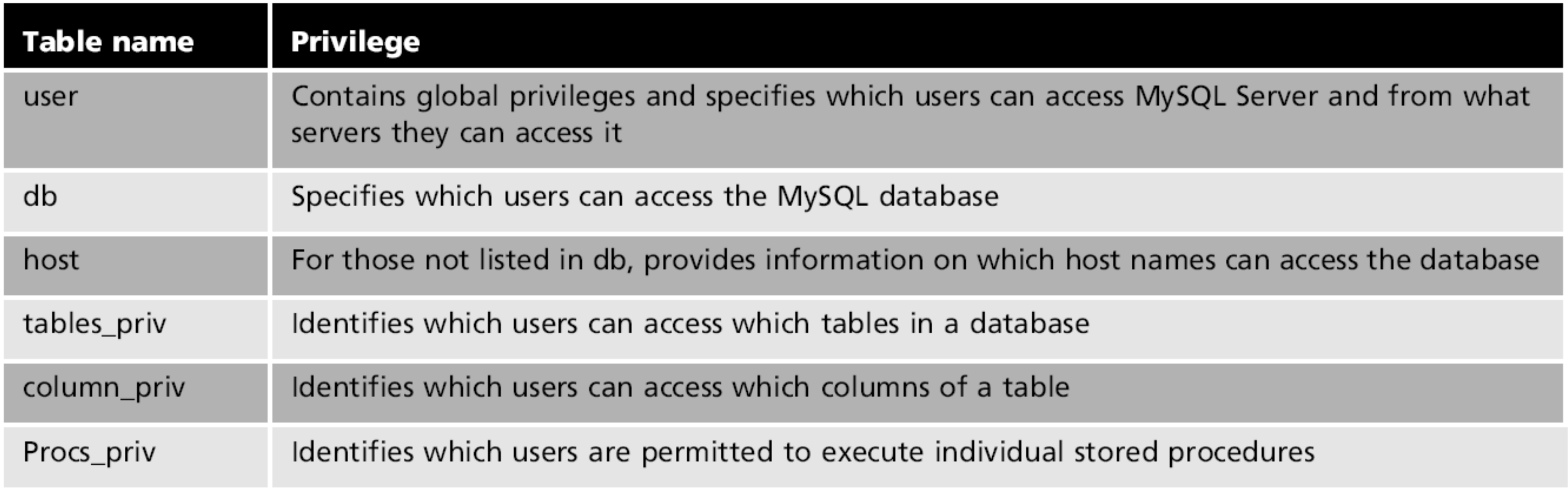


Table name	Privilege
user	Contains global privileges and specifies which users can access MySQL Server and from what servers they can access it
db	Specifies which users can access the MySQL database
host	For those not listed in db, provides information on which host names can access the database
tables_priv	Identifies which users can access which tables in a database
column_priv	Identifies which users can access which columns of a table
Procs_priv	Identifies which users are permitted to execute individual stored procedures

Table 6-2 Grant tables for privilege administration

User Privileges (cont'd.)

- Assigning privileges in Oracle
  - Two levels of privilege
    - System-level
    - Object-level
  - Administrators can grant system-level privilege
  - Object privileges granted by schema owner of an object
  - Privileges can be granted to PUBLIC
    - Grants privilege to all database users
    - Not recommended for security reasons

+ Roles

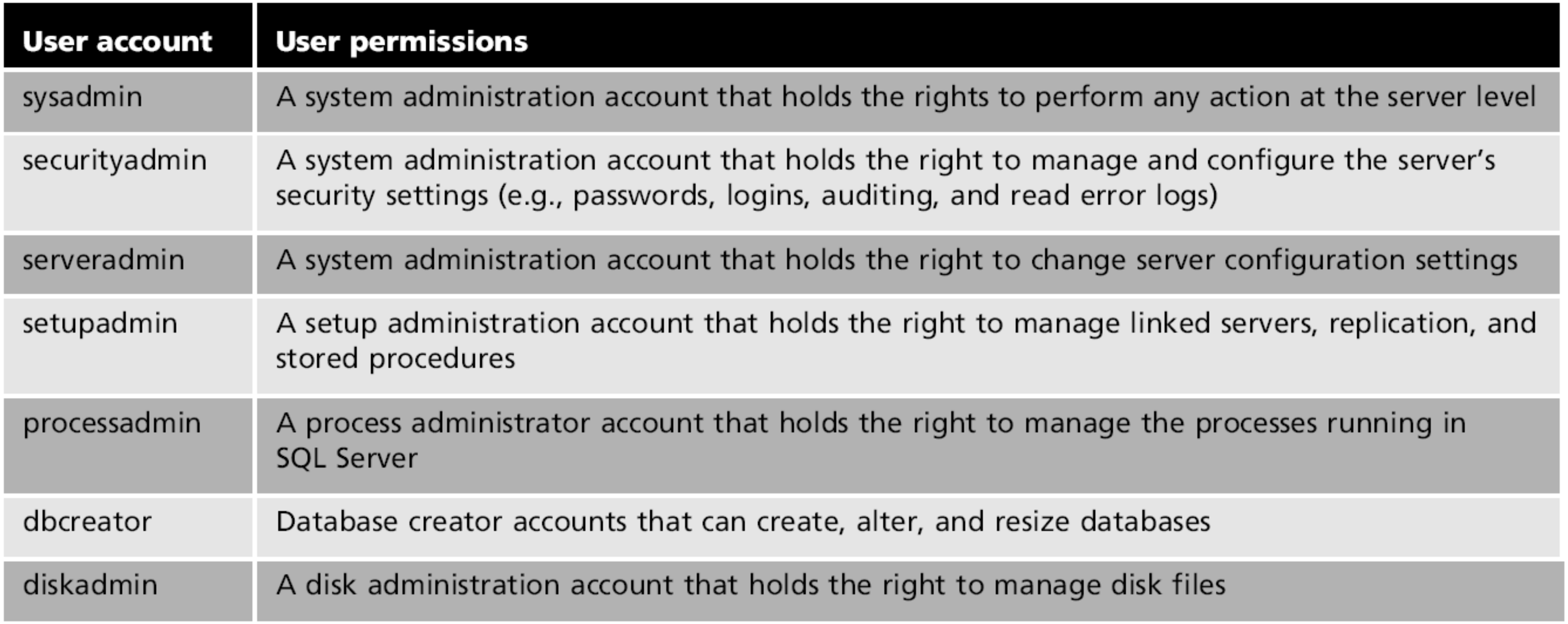
- Related privileges can be combined to create a role
  - Used to centrally manage group of objects or users
- Roles can be created for users, objects, and applications
  - Single role can be assigned to many users
  - Single user can be assigned many roles
- Advantages of using roles
  - Saves time and resources
  - Provides a central location for administration

Roles (cont'd.)

- Defining roles in SQL Server
  - Roles defined at either server or database level
- Server roles
  - Grant rights to manipulate the server environment
  - Rights granted to login accounts
- Database roles
  - Grant access to database objects
  - Rights granted to user accounts

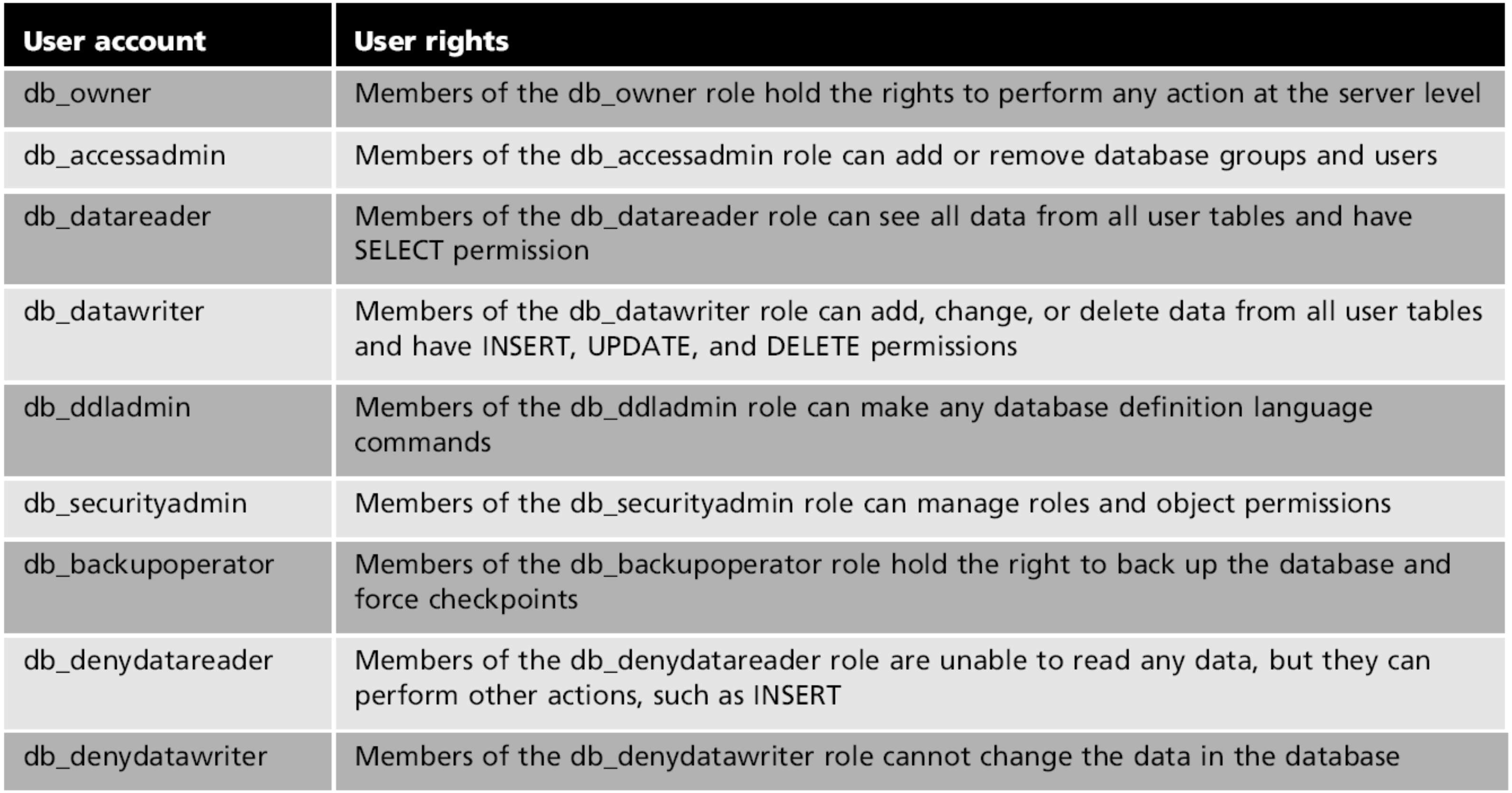
Roles (cont'd.)

- Five types of roles available within SQL Server
  - Fixed server, fixed database, user-defined, application, and public
- Fixed server roles
  - Provide server-level privileges
  - Cannot be changed or deleted
  - Users can be added to them
- Fixed database roles
  - Provides privileges specific to the database
  - Cannot be altered, yet users can be added



User account	User permissions
sysadmin	A system administration account that holds the rights to perform any action at the server level
securityadmin	A system administration account that holds the right to manage and configure the server's security settings (e.g., passwords, logins, auditing, and read error logs)
serveradmin	A system administration account that holds the right to change server configuration settings
setupadmin	A setup administration account that holds the right to manage linked servers, replication, and stored procedures
processadmin	A process administrator account that holds the right to manage the processes running in SQL Server
dbcreator	Database creator accounts that can create, alter, and resize databases
diskadmin	A disk administration account that holds the right to manage disk files

Table 6-3 Fixed server roles for SQL Server



User account	User rights
db_owner	Members of the db_owner role hold the rights to perform any action at the server level
db_accessadmin	Members of the db_accessadmin role can add or remove database groups and users
db_datareader	Members of the db_datareader role can see all data from all user tables and have SELECT permission
db_datawriter	Members of the db_datawriter role can add, change, or delete data from all user tables and have INSERT, UPDATE, and DELETE permissions
db_ddladmin	Members of the db_ddladmin role can make any database definition language commands
db_securityadmin	Members of the db_securityadmin role can manage roles and object permissions
db_backupoperator	Members of the db_backupoperator role hold the right to back up the database and force checkpoints
db_denydatareader	Members of the db_denydatareader role are unable to read any data, but they can perform other actions, such as INSERT
db_denydatawriter	Members of the db_denydatawriter role cannot change the data in the database

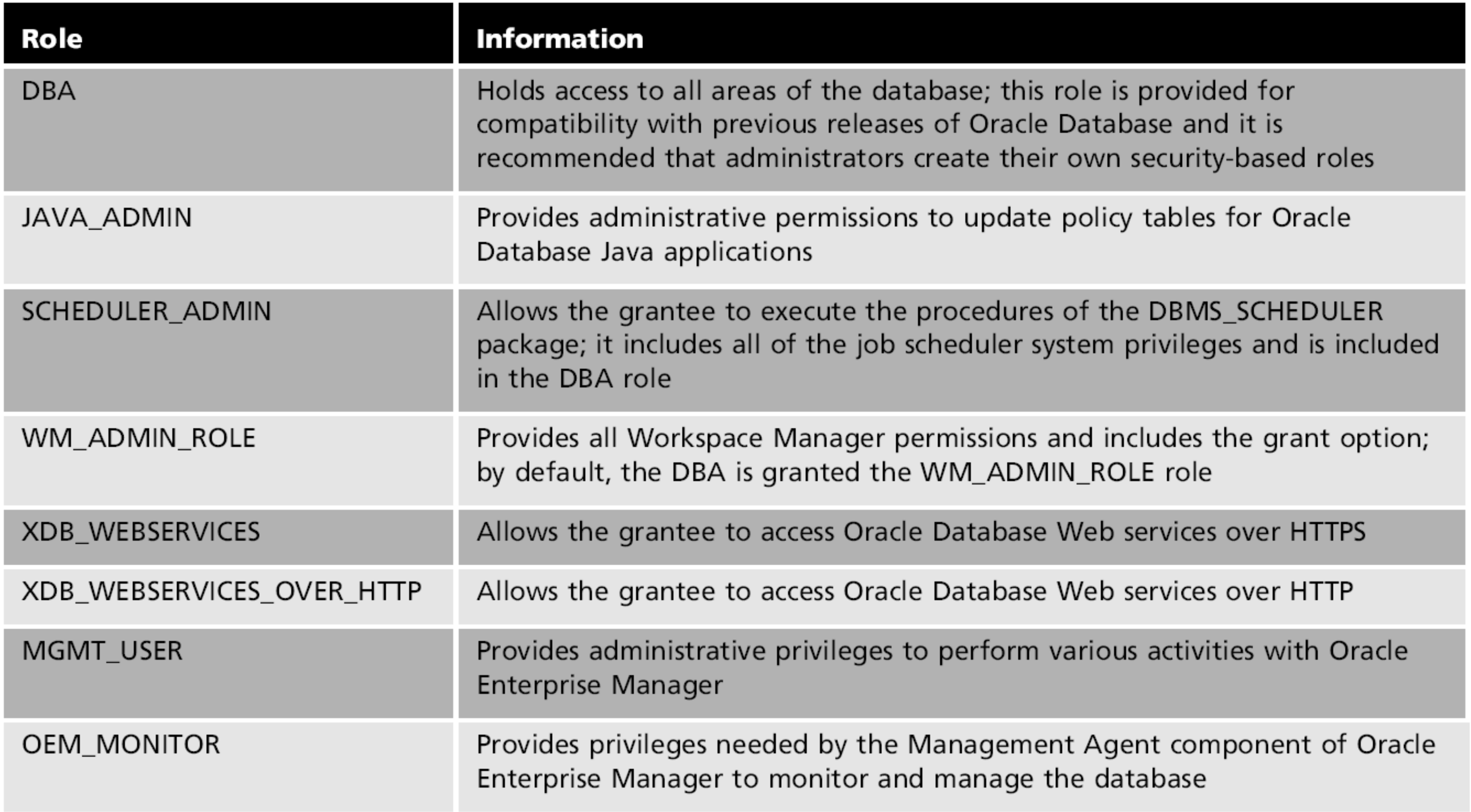
Table 6-4 Fixed database roles for SQL Server

+ Roles (cont'd.)

- User-defined roles
  - Built to control access of objects within the database
- Application roles
  - Created to support security requirements of applications
- PUBLIC role
  - Special role in which every database user is a member
  - Members cannot be removed
  - Provides a way to assign privilege for all users

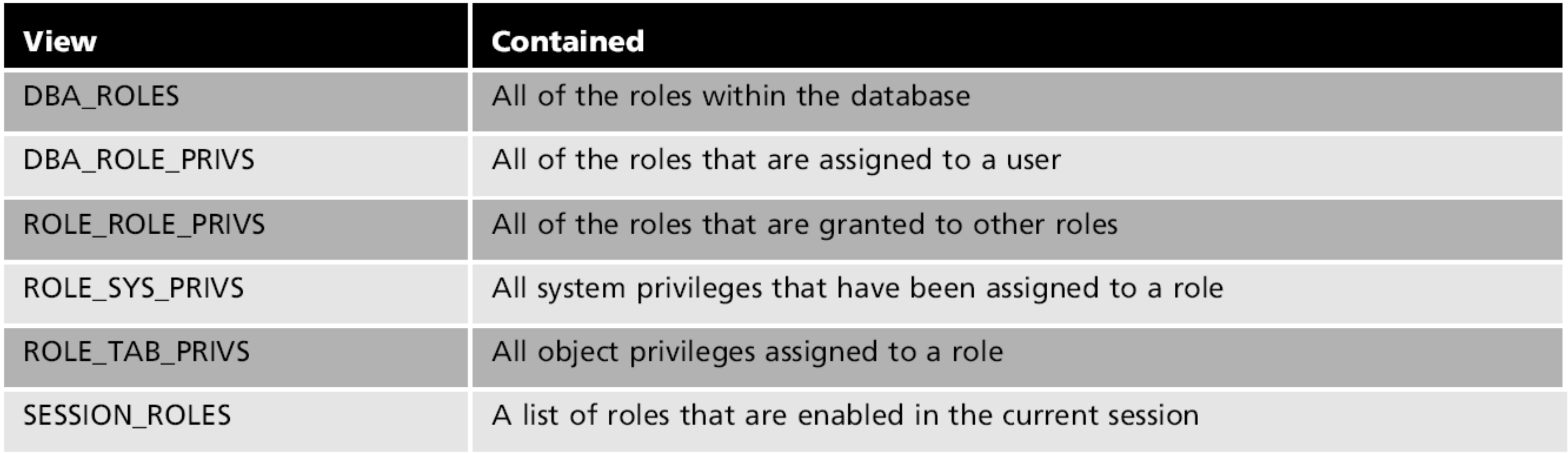
+ Roles (cont'd.)

- Defining roles in MySQL
  - Roles are not included in MySQL Server alone
  - Roles may be created using scripting and third-party applications
- Defining roles in Oracle
  - Several roles are built-in
  - Roles provide privileges at system and object levels
  - Roles can be granted to other roles
  - 33 roles exist for the Oracle database alone



Role	Information
DBA	Holds access to all areas of the database; this role is provided for compatibility with previous releases of Oracle Database and it is recommended that administrators create their own security-based roles
JAVA_ADMIN	Provides administrative permissions to update policy tables for Oracle Database Java applications
SCHEDULER_ADMIN	Allows the grantee to execute the procedures of the DBMS_SCHEDULER package; it includes all of the job scheduler system privileges and is included in the DBA role
WM_ADMIN_ROLE	Provides all Workspace Manager permissions and includes the grant option; by default, the DBA is granted the WM_ADMIN_ROLE role
XDB_WEBSERVICES	Allows the grantee to access Oracle Database Web services over HTTPS
XDB_WEBSERVICES_OVER_HTTP	Allows the grantee to access Oracle Database Web services over HTTP
MGMT_USER	Provides administrative privileges to perform various activities with Oracle Enterprise Manager
OEM_MONITOR	Provides privileges needed by the Management Agent component of Oracle Enterprise Manager to monitor and manage the database

Table 6-5 Common predefined Oracle roles



View	Contained
DBA_ROLES	All of the roles within the database
DBA_ROLE_PRIVS	All of the roles that are assigned to a user
ROLE_ROLE_PRIVS	All of the roles that are granted to other roles
ROLE_SYS_PRIVS	All system privileges that have been assigned to a role
ROLE_TAB_PRIVS	All object privileges assigned to a role
SESSION_ROLES	A list of roles that are enabled in the current session

Table 6-6 Locating roles in Oracle

Inference

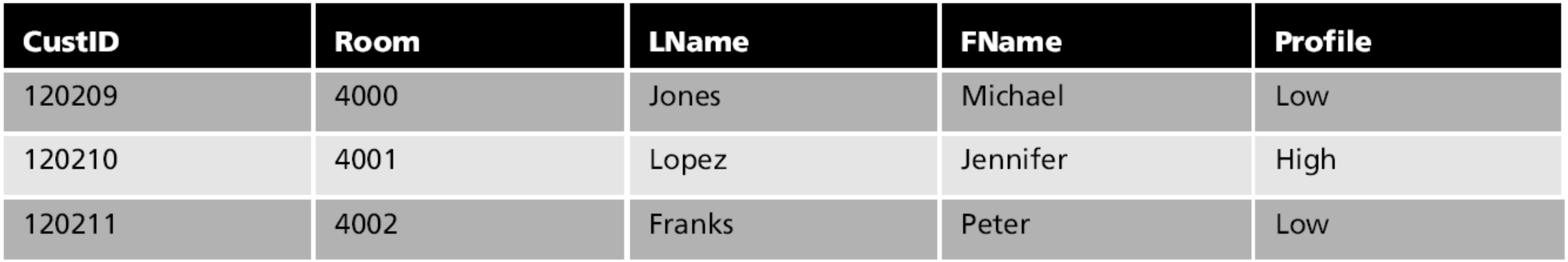
- Method for unauthorized users to obtain sensitive information
  - Making assumptions based on database's reactions or query responses
- Unauthorized users can draw conclusions about the database
  - Enables knowledge or understanding of the data
  - Users may be internal or external
- Inference is a great security threat
  - Difficult to predict, detect, and eliminate

+ Examples of Inference

- Two primary means of inference
  - Using logic
  - Using statistics
- Logic, relationship, and constraint interference
  - Well organized, logical tables are vulnerable to inference
- Example of logical inference
  - Hotel database table includes customer ID, last name, first name, and profile level
  - Customer ID is primary key

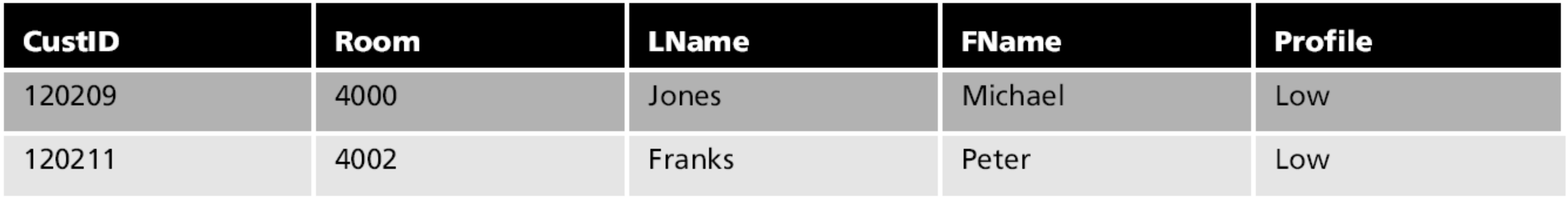
Examples of Inference

- Example of logical inference (cont'd.)
  - Security rule or constraint ensures only hotel managers can view information about high profile guests' rooms
  - Desk clerk cannot see room 4001 in the table
  - Desk clerk tries to book but cannot
  - Can infer room 4001 is occupied by a high profile guest



CustID	Room	LName	FName	Profile
120209	4000	Jones	Michael	Low
120210	4001	Lopez	Jennifer	High
120211	4002	Franks	Peter	Low

Table 6-7 Guest table view



CustID	Room	LName	FName	Profile
120209	4000	Jones	Michael	Low
120211	4002	Franks	Peter	Low

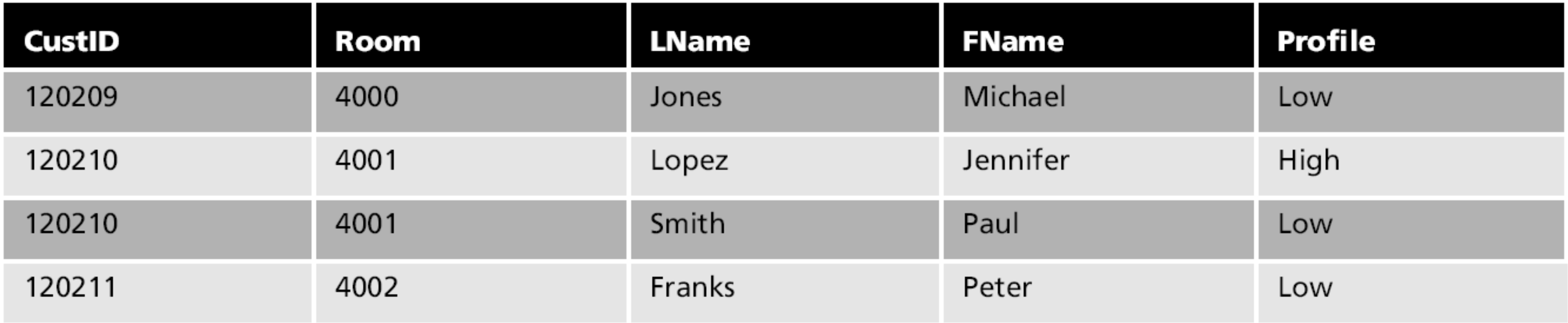
Table 6-8 Secured available room view

Examples of Inference

- Statistical inference
  - Statistical queries analyze the data but do not return actual data
  - Can be easily manipulated to retrieve sensitive information
- Example of statistical inference
  - Database user queries average of her salary and a co-worker’s
  - Uses basic arithmetic to determine co-worker’s salary

Minimizing Inference

- Techniques to limit a person’s ability to infer
  - Polyinstantiation
  - Log, monitor, and alert of events
  - Limit user capability
  - Limit query responses
- Polyinstantiation
  - Strategy that allows database to contain multiple instances of a record
  - Creates “fake” records
  - Downside: confusing false records with real ones



CustID	Room	LName	FName	Profile
120209	4000	Jones	Michael	Low
120210	4001	Lopez	Jennifer	High
120210	4001	Smith	Paul	Low
120211	4002	Franks	Peter	Low

Table 6-9 Polyinstantiation view

Minimizing Inference (cont'd.)

- Other ways to minimize
  - Less disruptive to database environment than polyinstantiation
- Log, monitor, and alert of events
  - Monitor activities
  - Set baseline and threshold alert for unusual user activity
  - Capture and analyze database activity logs

+ Minimizing Inference (cont'd.)

- Limit user capability
  - Limit user's query size
  - Allow only aggregate operators
- Limit query responses
  - Return classes and ranges instead of exact numbers

Summary

- Authentication: process of verifying user’s identity
- Authorization: process of verifying user’s permission to access a resource
- Credentials are used to authenticate and authorize
  - Can be required at different levels of an environment
- Operating system authentication requires user to have an account local to the server’s OS
- Database authentication checks user’s credentials against account residing in the database

+ Summary (cont'd.)

- Third-party applications can be used to verify a user's identity
  - Use security protocols such as Kerberos and PKI
- Types of authentication vary between database vendors
- Server-enforced password policies are vital to data security
- Related user privileges can be combined to create roles
  - Allows for centralized management and security