Oracle Database Security Features

By

Ashok Kapur

Hawkeye Technology, Inc.

Agenda

- Security Overview
- Secure Network Access
- Secure Database Access
- Secure Data
- Track Access
- Security Updates
- Summary

Security Overview

Security Overview

- Types of Security violations
  - External Breach
  - Internal Breach
  - Inadvertent Breach
- Implementation Tradeoffs
  - Delay legitimate access
  - Prevent unauthorized access

Security Dimensions

- Physical
  - Restrict physical system access
- Personnel
  - Trustworthy admins
- Good Application Design
  - Prevent users from corrupting data
  - Access control and levels of access
- Operating Procedures
  - Limit too much access to a single individual
  - Control Procedures for access and system changes
  - Who can have access to what data

Security Dimensions

- Technical
  - System Access control
  - Transmission of data
- Company Policies and Procedures
  - What data is confidential
  - Are there levels of confidentiality

Security Criteria

- Prevent unauthorized access
- Maintain system and data integrity
- Protect confidential data
- System availability for legitimate users
- Good system performance

Responsibilities

- Security Admin
- Network Admin
- System Admin
- Database Admin

Secure

Network Access

Secure Network Access

- Firewall
- Secure Signon
- Secure Transmission
- Network Parameter Changes

Firewall

- Firewall between external users and middle tier or database servers
- Good first line of defense
- Only prevents external users
- Poking holes in the firewall

Secure Signon

- Are the userid and passwords being passed in open?
  - Set the ORA_ENCRYPT_LOGIN environment variable to TRUE on the client machine.
  - Set the DBLINK_ENCRYPT_LOGIN server initialization parameter to TRUE.

Secure Transmission

- To secure data transmission setup Net8 TCP/IP via SSL
- Pre-requisites:
  - Purchase and install Advanced Security Option
  - Have access to Certification Authority
- Follow instructions in Metalink Bulletin 112490.1 “Configuring NET8 TCP/IP via SSL”

Parameter Changes

- Listener.ora changes:
  - ADMIN_RESTRICTIONS_listenername = ON
  - Set a listener password
  - Remove EXTPROC if not using it
  - Chmod 644 listener.ora (for unix OS)
- Sqlnet.ora parameters:
  - TCP.VALIDNODE_CHECKING=YES
  - TCP.EXCLUDED_NODES = {IP list}
  - TCP.INVITED_NODES = {IP list}

Secure

Database Access

Secure Database Access

- User Management
- Password Management
- Resource Limits

User Management

- Avoid using QUOTA UNLIMITED
- Assign reasonable quota on tablespaces
- REMOTE_OS_AUTHENT=FALSE
- Be VERY frugal with ANY grants
- Only grant necessary privileges

User Management

- Change default passwords on:
  - SYS, SYSTEM, SCOTT, DBSNMP, OUTLN, all JSERV accounts
- Expire and lock unnecessary accounts
  - Alter user <userid> password expire account lock;
- Institute password management
  - Create a profile with password expiration and password complexity scheme

Password Management Example

CREATE PROFILE all_users LIMIT

FAILED_LOGIN_ATTEMPTS 3 (Lock account after 3 attempts)

PASSWORD_LIFE_TIME 30 (expire after 30 days)

PASSWORD_REUSE_TIME 120 (can't user same passwd for 120 days)

PASSWORD_REUSE_MAX UNLIMITED

PASSWORD_VERIFY_FUNCTION chk_passwd (passwd complexity check)

PASSWORD_LOCK_TIME 1/24 (lock passwd for 1 hr if locked due to unsuccessful attempts)

PASSWORD_GRACE_TIME 10; (give 10 grace days after passwd lifetime to change passwd. Issue warnings during this time.)

ALTER USER myuser PROFILE all_users;

Password Verify Function Example

- CREATE OR REPLACE FUNCTION verify_function
(username varchar2,
password varchar2,
old_password varchar2)
……

- View sample script:
$ORACLE_HOME/rdbms/admin/utlpwdmg.sql

Resource Limits

- Resource Limits can be used to limit:
  - CPU Time
  - Logical Reads
  - Concurrent Sessions per User
  - Idle Time
  - Amount of Private SGA for Shared Sessions
- Can limit resources at Session Level or Call Level
- Set Init Parameter RESOURCE_LIMIT=TRUE
- Resource limits setup in PROFILE and assigned to users

Resource Limits Example

```
CREATE PROFILE res_lim LIMIT
  SESSION_PER_USER 2
  CPU_PER_CALL 120000  /* 2 min */
  IDLE_TIME 30     /* 30 min */
;
```

Profile Example

```
CREATE PROFILE def_profile
  LIMIT
      SESSIONS_PER_USER 2
      CPU_PER_CALL 120000
      IDLE_TIME 30
  FAILED_LOGIN_ATTEMPTS 3
  PASSWORD_LIFE_TIME 30
  PASSWORD_REUSE_TIME 120
  PASSWORD_REUSE_MAX UNLIMITED
  PASSWORD_VERIFY_FUNCTION chk_passwd
  PASSWORD_LOCK_TIME 1/24
  PASSWORD_GRACE_TIME 10;

ALTER USER app_user1 PROFILE def_profile;
```

Secure

Data

Secure Data

- Views
- Grants
- FGAC or VPD
- Column Encryption
- Triggers and Procedures

Limit Access to Data

- Grants
  - System Roles
  - System Privileges
  - Object Privileges
- Enable dictionary protection with
  - O7_DICTIONARY_ACCESSIBILITY = FALSE

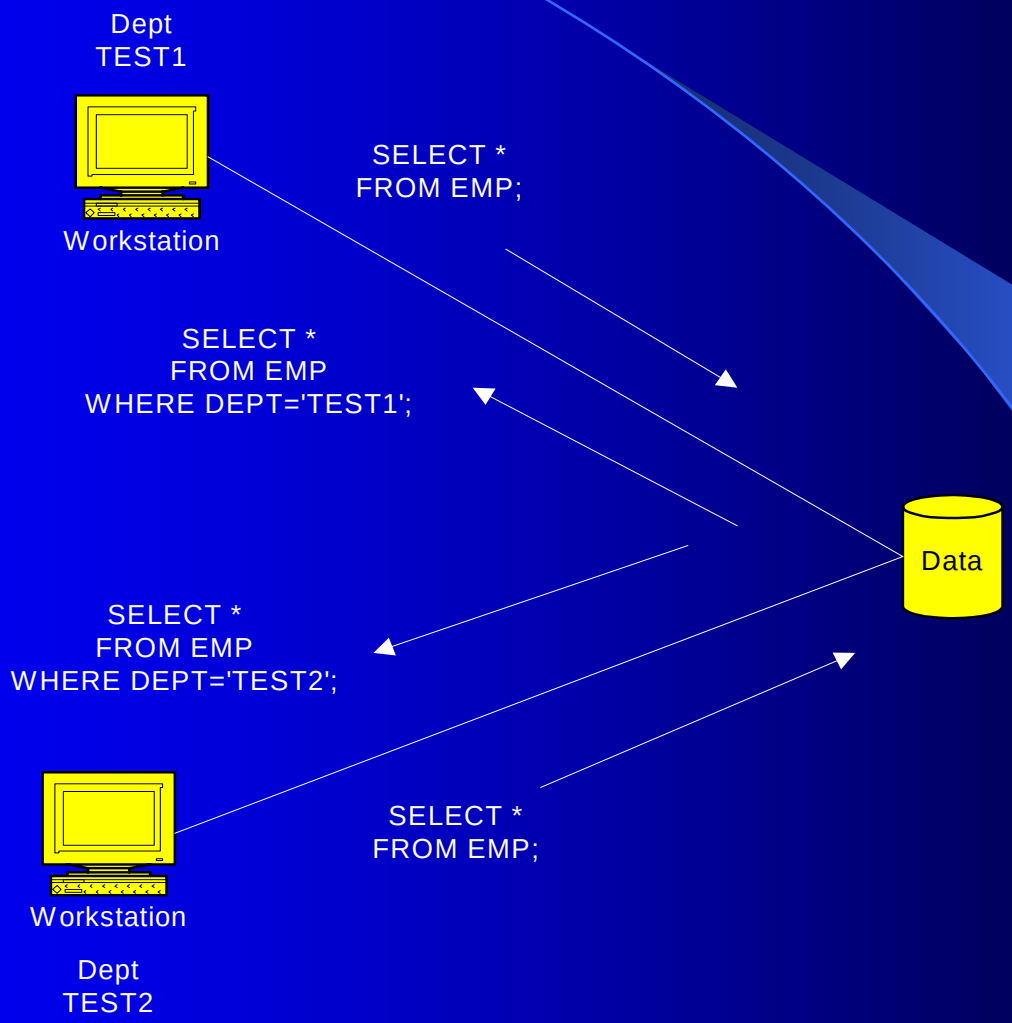
Secure Data

- Review PUBLIC and revoke unnecessary privileges
  - Review EXECUTE on UTL_SMTP, UTL_TCP, UTL_HTTP, UTL_FILE, DBMS_RANDOM
- Set UTL_FILE_DIR to specific directories and not *
- Consider creating a separate schema for application objects and users having limited access to the schema.

Fine Grain Access Control

- FGAC restricts the rows returned by the query
- Dynamically modifies the query to filter the rows returned to the user.
- Query modifications are done during parse time
- Modified queries are optimized and participate in query reuse algorithm

Fine Grain Access Control



Fine Grain Access Control

- Create Set Application context package
  - Use DBMS_SESSION.SET_CONTEXT
- Create Context
  - Use CREATE CONTEXT
- Create Application context package
  - Use SYS_CONTEXT and return where clause predicate
- Add a Security Policy
  - Use DBMS_RLS.ADD_POLICY

Data Encryption

- Column data can be encrypted via DBMS_OBFUSCATION_TOOLKIT
- Data Encryption Issues
  - How to generate a good key
  - Where to store the key
  - How to hide the key from users

Encryption Decisions

- Can hard-code a key or generate it via GET_KEY function of DBMS_OBFUSCATION_TOOLKIT
- Can store the key on the OS, Application Code or in the Database.
- Good idea to periodically get a new set of keys and re-encrypt all encrypted data
- Consider performance implications with encryption-decryption

Track

Data Access

Track Access

- One of the important steps in securing a system is monitoring access
- Oracle provides AUDIT facility to monitor access to the database and access to sensitive data

Auditing

- Auditing Levels
  - Session auditing
  - Privilege auditing
  - Statement auditing
  - Object auditing
- Set AUDIT_TRAIL=DB|OS|NONE to enable/disable the auditing feature
- Setup auditing options via AUDIT SQL statement

Session Audit Example

- **AUDIT SESSION** [WHENEVER SUCCESSFUL | WHENEVER NOT SUCCESSFUL]

- **SELECT os_username, username, timestamp, logoff_time, action_name, returncode, logoff_lread, logoff_pread FROM dba_audit_session**

```
OS_USERNM USERNAME TIMESTAMP           LOGOFF_TIME         ACT_NM  RT_CD  LRD   PRD
--------- -------- ------------------- ------------------- ------- ----- ----- ---
AK837609  SYSTEM   01/09/2002 12:46:01 01/09/2002 12:53:16 LOGOFF      0  542   0
AK837609  SYS      01/09/2002 12:53:16                    LOGON    1017
AK837609  SYSTEM   01/09/2002 12:53:20                    LOGON       0
```

Privilege Audit Example

- audit create table by akapur by access;

```
SELECT user_name, privilege, success, failure
FROM dba_priv_audit_opts;
```

```
USER_NAME  PRIVILEGE           SUCCESS    FAILURE
---------- ------------------- ---------- ----------
           CREATE SESSION      BY ACCESS  BY ACCESS
AKAPUR     CREATE TABLE        BY ACCESS  BY ACCESS
```

```
SELECT username,owner,obj_name,action_name,priv_used,timestamp
FROM dba_audit_object;
```

```
USERNAME  OWNER   OBJ_NAME  ACTION_NAME     PRIV_USED       TIMESTAMP
--------- ------- --------- --------------- --------------- -------------------
AKAPUR    AKAPUR  TEST      CREATE TABLE    CREATE TABLE    01/09/2002 13:43:12
SYSTEM    SYS     AUD$      TRUNCATE TABLE  DROP ANY TABLE  01/04/2002 14:43:28
```

Statement Audit Example

- AUDIT index BY ACCESS WHENEVER SUCCESSFUL
- **SELECT user_name,audit_option,success,failure**
  FROM dba_stmt_audit_opts

```
USER_NAME  AUDIT_OPTION    SUCCESS    FAILURE
---------- --------------- ---------- ----------
           NOT EXISTS      BY ACCESS  BY ACCESS
           INDEX           BY ACCESS  NOT SET
AKAPUR     CREATE TABLE    BY ACCESS  BY ACCESS
```

- **SELECT username,owner,obj_name,action_name,priv_used,timestamp**
  FROM dba_audit_object

```
USERNAME  OWNER   OBJ_NAME   ACTION_NAME     PRIV_USED        TIMESTAMP
--------- ------- ---------- --------------- ---------------- -------------------
SYSTEM    AKAPUR  TEST_IDX1  CREATE INDEX    CREATE ANY INDEX 01/09/2002 14:17:59
AKAPUR    AKAPUR  TEST       CREATE TABLE    CREATE TABLE     01/09/2002 13:43:12
SYSTEM    SYS     AUD$       TRUNCATE TABLE  DROP ANY TABLE   01/04/2002 14:43:28
```

Fine Grain Audit

- Built on Fine Grain Access control.
- FGA allows auditing of SELECT statements.
- For example, audit SELECT on Salary column of EMP table for all users that do not have HR role
- FGA ONLY works with cost based optimizer.
- ALWAYS ANALYZE the table being audited!!
- Auditing decision made during FETCH phase.

FGA Example

- Audit SALARY column of EMP table for records where SALARY > 10000
  - **EXEC DBMS_FGA.ADD_POLICY(OBJECT_SCHEMA=> 'SYSTEM', -**
    OBJECT_NAME=> 'EMP', POLICY_NAME=> 'VIEW_SAL', -
    AUDIT_CONDITION => 'salary > 10000 ', AUDIT_COLUMN => 'SALARY');

- **select * from emp;**
- **select timestamp, db_user, object_schema, sql_text**
  from dba_fga_audit_Trail;
- TIMESTAMP DB_USER OBJECT_SCH SQL_TEXT

  02/12/2002 12:57:51 SYSTEM SYSTEM select * from emp

- **select * from emp where salary < 500;**
  - NOTE: No audit record generated.

Security Updates

Security Bulletins

- Always review Security Bulletins from Oracle via OTN or CERT.
- Apply Security patches or workarounds as soon a possible.

Further Reading

- Label Security
- Enterprise Login
- Oracle Internet Directory for enterprise login and assigning application roles
- Use of SET ROLE and assigning passwords to ROLEs.

Summary

Summary

Security Issues	Solutions
Unauthorized Users	UerId and Password Protection
Unauthorized Access to data	. Levels of Access . FGAC . Data Encyption
Eavesdropping	Network Encryption
Data Corruption	. Good application Design . Good database design

Summary

Security Issues	Solutions
Denial of Service	. Firewall setup . Network parameters . Resource Management
User complexity	. Central account mgmt . Single Signon
Accountability	. Auditing . Distribute access control

Reference

- Oracle Manuals
- OTN
- Metalink Notes:
  - 112490.1, 99721.1, 185703.1
  - 67977.1, 130652.1
- CERT

Questions?

Contact

Ashok Kapur
Hawkeye Technology, Inc.
afkapur@hawkeyetechnology.com
http://www.hawkeyetechnology.com