Oracle Database Security and Audit

Beyond Checklists

Learning objectives

- Key issues
  - Data access
  - Privilege escalation
  - Defaults
  - The PUBLIC role

The key issue

- Everything in database security comes down to …
  - Data access
- We are trying to protect the data from unauthorized access

Database access is issue #1

- A database can only be accessed if you have three pieces of information
  - The IP Address or hostname
  - The Service name / SID of the database
  - A valid username / password

Privilege exploitation

- The second biggest issue in an Oracle database relates to privileges
- These can be easily exploited by malicious user or inadvertently by authorized user

Privileges - %ANY%

- Sweeping system privileges give access beyond what may be intended
  - Privileges with the "ANY" modifier allow access to all schemas (accounts) in the database!

Example:
Any user with the system privilege CREATE ANY TRIGGER can use it to steal data from any table, even with no access to the table!

Defaults

- Default software installed
- Default configuration and settings
- Default users and roles installed
- PUBLIC grants by default

Default functionality

- Every version of Oracle increases the available functionality
- Most do not do custom installations
  - Use the Database Creation Assistant (dbca) to create databases
    - Default functionality installed (APEX, JVM, etc.)
- 11gR1 brought
  - APEX by default
    - Weak authentication in the application layer
    - Not needed unless you want an APEX

Exploiting default functionality

- Default schemas create an attack surface
  - Excessive system privileges
  - Coding issues (SQLi, etc.)
- Functions and features installed that expose the operating system can be attacked
- SQL Injection can escalate privilege inside the database and to the operating system

Stored coding issues

- DEFINER vs. INVOKER rights
- **DEFINER** rights run the code with all privileges as the owner of the code (think suid in UNIX or "Run As" in Windows)
  - Default method
- **INVOKER** rights run the code under the privileges of the account calling the code
  - All objects referenced in the called code must be accessible to the account executing the code

Open system

- A big problem with Oracle is that its open by "default"
  - Database options installed by default
  - Access to the PUBLIC role (therefore, everyone)
  - Networking functionality
- This is to allow as many people to use it out of the box
  - But it makes Oracle insecure
- When combined with too many features and functions, DEFINER rights, and the PUBLIC issue, the database is insecure

Where are the installed defaults

- Operating system binaries and DLLs
- Database options (schemas)
- Both

How are defaults installed

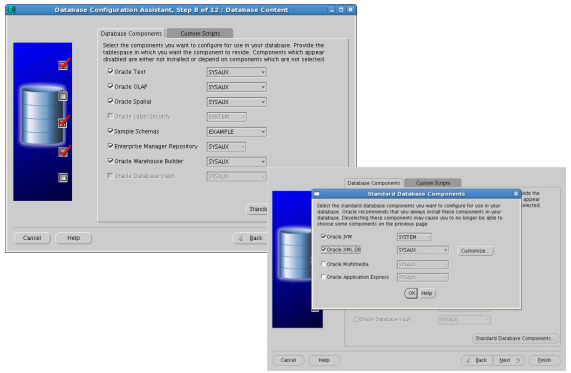
- Two sources
  - Universal installer (OUI)
  - Database creation Assistant (dbca)
- Others
- Many defaults installed

Universal installer (oui)

- Java based GUI to simplify and automate the software installation process
- When installing Oracle software, few choose the "Custom" install option

Database creation assistant (dbca)

- Java based GUI to simplify and automate the creation or maintenance of databases



Default features

- Many schemas are installed by default
  - 9iR2 @ 30 by default
  - 10gR2 @ 27 by default
  - 11g R1 @ 35 by default
  - 11g R2 @ 36 by default
  - 12c R1 @ 35 by default

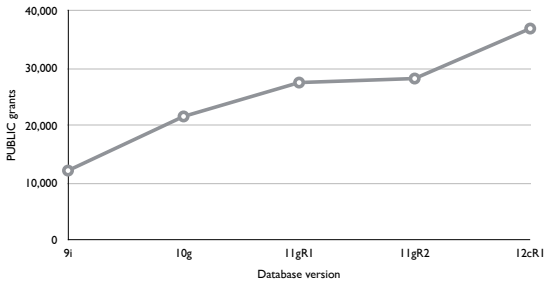
The PUBLIC role

- Started out as a group in concept - Oracle v4
- Has now evolved into a role
- All database users are members of the PUBLIC role
- Issues with database options
  - Most DB options give access to the PUBLIC role

PUBLIC access growth

- Access to objects by the PUBLIC gets bigger – (figures can vary based on installation)
  - Added functionality in the database (database options)
- 9iR2 – 12,132
- 10gR2 – 21,530 - 77.4% increase over 9iR2
- 11gR1 – 27,461 - 27.5% increase over 10gR2
- 11gR2 – 28,160 -1% increase over 11gR1
- 12cR1 – 36,866 - 24% increase over 11gR2

PUBLIC growth



PUBLIC privileges

- This is one of the biggest problems with Oracle
- All database schemas and users are members of the PUBLIC role
- There is no such thing as a read-only user
  - Every user can have 28,000+ privileges
- Major cause of vulnerabilities and access control issues
- Access to key OS / Network packages exist by default
  - Addressed by ACLs implemented in XML (if XDB is installed) in 11g

Q&A