Top 10 Database Security Threats

Data at Risk

341,749,431

Total publicly known data records taken in the US since 2005.
Data has Value

Imperva Background

**Imperva's mission is simple:**
Protect the data that drives business

**The leader in a new category:**
Data Security

HQ in Redwood Shores CA; Global Presence
+ Installed in 50+ Countries
1,200+ direct customers; 25,000+ cloud users
+ 3 of the top 5 US banks
+ 3 of the top 10 financial services firms
+ 3 of the top 5 Telecoms
+ 3 of the top 5 specialty retailers
+ Hundreds of small and medium businesses

**Research Arm:**
Application Defense Center (ADC)
Agenda

- Top 10 Database Security Threats
  + Definition
  + Analysis
  + Consequence
  + Mitigation
- Imperva Overview
- Questions and Answers

Database Top 10 Threats

- Excessive Privilege Abuse
- Legitimate Privilege Abuse
- Privilege Elevation
- Weak Audit
- SQL Injection
- Database Platform Vulnerabilities
- Denial of Service
- Database Communication Protocol Vulnerabilities
- Weak Authentication
- Backup Data Exposure
Excessive Privilege Abuse

**Database Top 10 Threats**

**Excessive Privilege Abuse**

- **Definition:** Users (or applications) granted database access privileges in excess of “business need-to-know”

  Canada Revenue Agency accused of multiple counts of unauthorized access

  [Image: Feds crack phone clone scam that cost Sprint $15m]

  [Image: Mayo Clinic fires employees for accessing patient records]

  [Image: Federal prosecutors hit cellphones to defraud]

  [Image: The operation dates back to 2002, complaining that they...]

  [Image: Updated: Sep 15, 2010 1:04 PM PDT]
## Database Top 10 Threats
### Excessive Privilege Abuse

**Analysis:**
- Hard to obtain a true list of required privileges
  - Even harder to keep this list updated
- Database ACL semantics are too limited
  - Not enough to specify operations allowed for table by user

**Consequence:**
- Any “minor” breach becomes a major incident!
- See SQL Injection

**Mitigation**
- More granular ACLs: Query ACLs
  - What queries are allowed against the table by this user
- Automatic and Dynamic ACL profiling
Top 10 Database Vulnerabilities

**Mitigation**

**Query Access Control Lists**

**Data Leakage**

via Web Application

\[ \text{Select } * \text{ from students where username = ? And password = ?} \]

**Normal Usage**

\[ \text{Select } * \text{ from users where username = 'john' and password = 'smith'} \]

**SQL Injection**

\[ \text{Select } * \text{ from users where username = 'john' and password = 'smith' or 1=1} \]

**Additional Clause**

---

Legitimate Privilege Abuse
### Database Top 10 Threats

**Legitimate Privilege Abuse**

- **Definition:** Abuse legitimate db privileges for unauthorized purposes

### Analysis

- Use simple and available desktop tools
- Retrieve large quantities of data
- Store sensitive data locally
- Make unauthorized changes
### Database Top 10 Threats

#### Legitimate Privilege Abuse

- **Consequence**
  - Data theft
  - Data loss
  - Embezzlement

- **Mitigation**
  - More granular ACL: Context based ACL
  - ACL augmented with the context of query
    - E.g. Client machine, client software, time-of-day

---

**Privilege Elevation**
**Database Top 10 Threats**

**Privilege Elevation**

- **Definition:** Low privileged user exploit database vulnerabilities to gain administrative privileges.

---

**Database Top 10 Threats**

**Privilege Elevation**

**Part 1**

![Image of database connection interface]
Database Top 10 Threats
Privilege Elevation

Part 2

---

Database Top 10 Threats
Privilege Elevation

Part 3
Database Top 10 Threats
Privilege Elevation

Part 4

SQL> exec cxsys.drillload.validate_stat('grant dba to scott');
BEGIN cxsys.drillload.validate_stat('grant dba to scott'); END;

ERROR at line 1:
ORA-06510: PL/SQL: unhandled user-defined exception
ORA-06512: at "CTXSYS_DRILLDO" line 42
ORA-06510: no statement parsed
ORA-06512: at line 1

Database Top 10 Threats
Privilege Elevation

Part 5

SQL> connect scott/tiger@toe-db
Connected.

SQL> select username, password from dba_users;

<table>
<thead>
<tr>
<th>USERNAME</th>
<th>PASSWORD</th>
</tr>
</thead>
<tbody>
<tr>
<td>SYS</td>
<td>D4C6501688B20DCA</td>
</tr>
<tr>
<td>SYSTEM</td>
<td>D0F79E1B3136E37</td>
</tr>
<tr>
<td>OUTLN</td>
<td>403303SE08353CB1</td>
</tr>
<tr>
<td>DBSNMP</td>
<td>E966D244505921CCC</td>
</tr>
<tr>
<td>AURORAJOSSUTILITIES</td>
<td>0000017095594B</td>
</tr>
<tr>
<td>OSESHTPSADMIN1</td>
<td>00000183592758</td>
</tr>
<tr>
<td>AURORAOPUSERAUTHENTICATED</td>
<td>D0000053735240</td>
</tr>
<tr>
<td>ORDSYS</td>
<td>TP60202C7060B64F</td>
</tr>
<tr>
<td>ORDPLUGINS</td>
<td>BB202B2C183931F09</td>
</tr>
<tr>
<td>MDSYS</td>
<td>T2B9898B2D295B</td>
</tr>
<tr>
<td>CTXSVS</td>
<td>28488888B28104C</td>
</tr>
</tbody>
</table>
| SQL>
**Database Top 10 Threats**

**Privilege Elevation**

- **Analysis**
  - Susceptible objects
    - Stored procedures
    - SQL Statements
    - Built-in functions
  - Types of vulnerabilities
    - Buffer Overflow
    - SQL Injection
    - Semantic glitches

- **Consequence**
  - Any “minor” breach becomes a major incident
  - Built-in access control becomes ineffective

- **Mitigation**
  - More granular ACL: Query level ACLs
  - Traditional IPS: Patterns for susceptible objects
  - Correlated detection
Weak Audit

Database Top 10 Threats
Weak Audit

- Definition: Audit policies that rely on built-in database mechanisms suffer a number of weaknesses
**Database Top 10 Threats**

**Weak Audit**

- **Analysis**
  - Performance degradation and DBA attention span
  - Knowing what matters in the mountain of audit data
  - Vulnerability to privilege elevation as well as other database attacks
  - Limited granularity
  - Proprietary

---

- **No end-to-end identity tracking**
  - In 3 tier environments
  - Application server uses a pooled connection policy to access database
  - Built-in mechanism only records account name and have no information with respect to the actual end user.
Database Top 10 Threats

**Weak Audit**

- **Consequence**
  - Regulatory problems
  - Data is not there when you need it

- **Mitigation**
  - Independent auditing device

---

**SQL Injection**
SQL Injection

**Definition:** Attacker inserts an unauthorized SQL **statement** through an SQL **data channel**:
- Data Channel - eg. Parameter of stored procedures or Web form
- Most common attack type on web connected databases

**Analysis:**
- Non-validated input parameters
**Database Top 10 Threats**

**SQL Injection**

- **Consequence**
  - Access to unauthorized data
  - Unauthorized data manipulation
  - Denial of Service
  - Privilege elevation

- **Mitigation**
  - More granular ACL: Query ACLs
  - Automatic and dynamic generation of ACLs
  - Correlation with Web front end
Definition: Vulnerabilities in underlying operating systems and services installed on a database server

Analysis
- OS - Windows 2000, UNIX, etc.
- Additional Services – eg. SNMP, NETBios, DCOM, DNS, etc.
Database Top 10 Threats
Database Platform Vulnerabilities

- Example: Slammer worm on Windows machines running MS SQL Server

Update: Slammer worm slugs Internet, slows Web traffic
By Stacy Cowley and Martyn Williams, IDG News Service
January 25, 2003 12:10 PM ET

IDG News Service - A new worm that has been attacking a known vulnerability in Microsoft SQL 2000 Web servers and that has been slowing down or halting Internet traffic worldwide could prove as tricky a nemesis as security foes Code Red and Nimda, according to firms tracking the outbreak.

Half a dozen security outlets have issued bulletins describing worm W32/SQL Slammer, dubbed "Slammer." Using a buffer overflow to take over a server, the worm sends out a flood of packets, an effect similar to a denial-of-service attack.

- Consequence
  + Server is compromised
  + Direct access to database files
  + Local access through admin roles
  + Install backdoors

- Mitigation
  + Network ACLs: Simple FW to allow access only to required services
  + Network IPS: Traditional detection of known vulnerabilities
Denial of Service

**Database Top 10 Threats**

**Denial of Service**

- Definition: Attacks that affect the availability of information from the database to users
Denial of Service

**Analysis**
- Specific vulnerabilities: SQL injection, platform vulnerabilities, database vulnerabilities
- Resource oriented attacks: Exhaustion of specific resources such as bandwidth, CPU and database connections

**Consequence**
- Critical for modern day organizations
- Paralyzing the entire operation of an organization or part of it

**Mitigation**
- Specific mechanisms for specific vulnerabilities
Database Top 10 Threats
Denial of Service

Mitigation (Cont.)

+ Specific mechanisms for specific vulnerabilities
+ Resource control mechanisms
  - Timing responses
  - Sizing responses
  - Connection control
+ Problem detection
  - Timing latency in system
    - If there is a dramatic increase in latency then DoS detected and addressed
**Database Communication Protocol Vulnerabilities**

- **Definition:** Tampering with db related network protocol messages
- **Analysis**
  - Each vendor relies on proprietary network protocol to communicate data and commands
  - Such complex (and mostly obscure) protocols are prone to security vulnerabilities

- **Consequence**
  - Unauthorized data access and manipulation
  - Denial of Service

- **Mitigation**
  - Protocol validation engine (addresses even unknown vulnerabilities)
    - Only let through normal client generated messages
    - Throw out requests that use hidden qualities or features of the protocols
  - Reactive protocol validation (addresses known vulnerabilities)
    - Checks for specific known attacks
Weak Authentication

Database Top 10 Threats
Weak Authentication

- **Definition:** Weak account names and/or passwords
- **Analysis**
  - Account name often adhere to some organizational standard (e.g. John.Smith, Jane.Doe, JSmith, J.Doe)
  - Bad (or rather predictable) choice of passwords by users
Database Top 10 Threats
Weak Authentication

- Consequence
  + Credential theft
  + Brute force attacks are feasible

If Your Password Is 123456, Just Make It HackMe

By ASHLEE VANCE
Published January 26, 2010

Back at the dawn of the Web, the most popular account password was “12345.”

MOST POPULAR PASSWORDS
Nearly one million RockYou users chose these passwords to protect their accounts.

1. 123456 17 michael
2. 123456 18 ashley
3. 123456789 19 654321

Today, it’s one digit longer but hardly safer: “123456.”

Despite all the reports of Internet security breaches over the years, including the recent attacks on

- Brute force
- Unauthorized use of credentials

Database Top 10 Threats
Weak Authentication

- Mitigation
  + Use two factor authentication
  + Enforce strong password policy
  + Detect and identify related attacks
    - Brute force
    - Unauthorized use of credentials
  + Actively assess authentication mechanism
    - Make sure users choose strong passwords
Backup Data Exposure

**Database Top 10 Threats**

**Backup Data Exposure**

- **Definition:** Unencrypted data on Back-up Tapes and Disk
- **Analysis**
  - Many recent incidents where backup media is lost or stolen
**Backup Data Exposure**

**Consequence**
- Exposure of huge amounts of sensitive information

**Mitigation**
- **End-to-end encryption:**
  - Problematic: Application dependent, complex key management, persistent exposure if user’s key is lost
- **Disk encryption:** data have to be encrypted again for backup
- **Database encryption:** Performance degradation
  - Indexing encrypted information
- A better solution is yet to be found
What we do in 60 Seconds

Question & Answer
<table>
<thead>
<tr>
<th>More Information: <a href="http://www.imperva.com">www.imperva.com</a></th>
</tr>
</thead>
<tbody>
<tr>
<td>Blog</td>
</tr>
<tr>
<td>iTunes/Podcasts</td>
</tr>
<tr>
<td>YouTube</td>
</tr>
<tr>
<td>Twitter</td>
</tr>
<tr>
<td>Linkedin</td>
</tr>
<tr>
<td>Facebook</td>
</tr>
</tbody>
</table>