The Case for: Database Activity Monitoring, Auditing, and Real-Time Protection
Agenda

Database Activity Monitoring, Auditing, and Real-Time Protection

Most of the world’s sensitive data is stored in commercial database systems such as Oracle, Microsoft SQL Server, IBM DB2 and Sybase – making databases an increasingly favorite target for criminals. This may explain why SQL injection attacks jumped 134 percent in 2008, increasing from an average of a few thousand per day to several hundred thousand per day according to a recently published report by IBM. To make matters worse, Forrester reports that 60 percent of enterprises are behind in applying database security patches.

This presentation will discuss why your databases are at such a high level of risk and how you can address these risks. While compliance is often a business driver for making changes in your organization, putting the right security measures in place for your databases will also allow you to meet your auditing requirements.
Could this happen to your company?

- Heartland settles with Discover over breach
  Dan Kaplan
  September 01, 2010

Heartland Payment Systems, the New Jersey-based credit card processor that fell victim to the largest reported data breach of all time, announced on Wednesday that it will settle with Discover for $5 million. Heartland already has settled with Visa for $60 million and MasterCard for $41.4 million over the breach, which exposed an estimated 130 million credit and debit card numbers to organized criminals. The settlement money will be used by Discover to recoup costs related to reissuing cards and any incidents of fraud consumers may have experienced.
What is DAM? Database Activity Monitoring

The process of monitoring access to and modification of selected database objects and resources within databases (OLTP & DW) and retaining a detailed record of the access, and if applicable, proactively trigger real-time actions and can be retrieved and analyzed as needed......

Tells you who did what, to which data, when, how and why

To address:

• Database Security - Protect your corporate data assets

• Regulatory Compliance
Key Business Drivers for DAM: Database Activity Monitoring

1. Prevent data breaches
   • Mitigate external and internal threats
   • Protect customer data & corporate secrets (IP)

2. Ensure data integrity/governance
   • Prevent unauthorized changes to sensitive data (business user vs. privileged user)

3. Reduce cost of compliance
   • Automated, continuous and centralized controls
     ▪ Across DBMS platforms and applications
     ▪ Across SOX, PCI, SAS70, …
   • Simplify processes
Database Servers are the Primary Source of Breached Data

% of Records Breached (2010)

- Database servers: 92%
- Laptops & backup tapes: <1%
- Desktop computer: <1%
- All other sources: 7%

“Although much angst and security funding is given to offline data, mobile devices, and end-user systems, these assets are simply not a major point of compromise.”

- 2009 Data Breach Investigations Report

73% of security professionals anticipate the volume of database security attacks will continue to increase

- Enterprise Strategy Group, Databases at Risk, September 2009

2009 75% were database servers

2010 Data Breach Report from Verizon Business RISK Team

Why?

- Database servers contain your most valuable information
  - Financial records
  - Customer information
  - Credit card and other account records
  - Personally identifiable information

- High volumes of structured data

- Easy to access

“Because that’s where the money is.”
- Willie Sutton
Operation Get Rich or Die Tryin’

- Gonzalez sentenced to 20 years (“Largest Hacking Case Ever Prosecuted”)
- Heartland, 7-Eleven, Hannaford: Stole 130M cards via SQL injection, network reconnaissance, malware, sniffers
- Dave & Buster’s Restaurants: Stole admin password file from POS service provider
- TJX, OfficeMax + 6 other retailers: Stole 40M cards via SQL injection & war driving
- San Diego case: International ring (Ukraine, Estonia, PRC, Philippines, Thailand)
  - “Maksik” Yastremskiy earned $11 million from stolen credit cards
  - Hacked 11 Turkish banks; sentenced to 30 years in Turkish prison

“Our most formidable challenge is getting companies to detect they have been compromised”

Kimberly Kiefer Peretti, senior counsel, DoJ
From Latvia to Montana

- 192,000 records stolen from broker-dealer in Montana
- Online extortion scheme by Latvians in Netherlands
- Used SQL injection to compromise database
  - Default blank password
  - Never reviewed logs
  - Only alerted to breach by extortion email
- Company fined $375,000 by regulators (FINRA)
- FBI-USSS Advisory (Feb. 2009)
  - Disable harmful stored procedure calls including:
    - Disable access to the xp_cmdshell functions within MS SQL Server.
    - Remove the "xplog70.dll" file from the server.
    - Require password for "sa" account
    - "Attackers generally create tables into which they store malware or data collected from the enterprise … Restrict the capabilities of accounts used to modify databases."

Cost of a Data Breach

- Forrester survey of 305 IT decision makers
- Secrets (e.g., strategic plans) are twice as valuable as custodial data (personal information, credit card data, etc.)
  - 2/3 of value in corporate information portfolio from non-regulated data (secrets)
- Companies focus mainly on preventing accidents (email, etc.)
  - But deliberate theft of information by employees is much more costly
  - Damage caused by rogue IT administrator = $482K (average)
  - Average cost of accidental leakage = $12K
- Most CISOs don’t really know if their controls really work
- Note: Survey does not address other costs such as fines
  - Australian bank was fined $500K by VISA
  - Heartland breach cost = $140M
- The average total cost of a data breach rose from $6.65 million in 2008 to $6.75 million in 2009
Database Danger from Within

- “Organizations overlook the most imminent threat to their databases: authorized users.” (Dark Reading)

- “No one group seems to own database security … This is not a recipe for strong database security” … 63% depend primarily on manual processes.” (ESG)

- Most organizations (62%) cannot prevent super users from reading or tampering with sensitive information … most are unable to even detect such incidents … only 1 out of 4 believe their data assets are securely configured (Independent Oracle User Group).

Oracle Survey: Most Organizations Have Very Weak Database Controls

- 3 of 4 organizations can’t prevent privileged users from reading or tampering with data in their databases
- 2 out of 3 cannot detect or prove that privileged DB users are not abusing their privileges
- Almost 64% don’t monitor database activity, do so ad hoc, or don’t know if anyone is monitoring
  - Less than 1/3 of those monitoring are watching sensitive data reads & writes
- 85% ship production data to developers & outside parties
- Majority don’t apply Critical Patch Updates in timely manner, nor take steps to ensure Web-facing apps are not subject to SQL injection attacks

Forrester: The Truth About Database Security!

- DBAs spend less than 5% on database security.
- 20% of enterprises take advanced security measures—basic dbms security is not enough.
- 70% behind in database security patches, making databases highly vulnerable.
- 75% of attacks are internal, which are harder to detect.
- 80% don’t have a database security plan.
- Most don’t implement data security policies.
- Most attacks on databases are difficult to detect.
What The Analysts Are Saying

Jeff Wheatman

“Most enterprises are paying too little attention to the very real security risks associated with their databases … Native logging isn’t the answer [lack of granularity, separation of duties not supported, high overhead].”

Noel Yuhanna

“Noel Yuhanna

“Basic database security is no longer sufficient to protect private data … Critical databases have hundreds or even thousands of connections per second, so it is humanly impossible to view and detect security anomalies.”

Jon Oltsik

“Databases house a higher percentage of confidential data than any other type of repository … In most organizations (63%), database security depends primarily on manual or ad hoc processes … no match for well-organized cybercriminals, malicious insiders and accidental events.”
Examples of Insider Threat

- **Unauthorized changes to financial/ERP data**
  - DBA accidentally deleted critical financial table during production hours (was doing a favor for application developer, bypassing change process)
  - Outsourcer erased logs showing he made changes during the day (because it was more convenient than during the night)

- **Theft of sensitive data**
  - Departing employees stealing design information & other intellectual property
  - Foreign hackers stealing design information, oil drilling data
  - DBAs and outsourcers have sold customer information to competitors, crime syndicates and tax authorities

- **Internal fraud**
  - Mortgage processor: Insider changed credit scores to make loans look better
  - Mobile telecom: Insider created & sold pre-paid phone cards
  - Electric utility: Insider gave free service to friends and family as part of low-income assistance program
  - Health provider: Insider sold medical identities for insurance fraud
Database admin steals 2.3M consumer records at Fidelity National subsidiary

The data included names, addresses, birth dates, bank account and credit card information.

July 03, 2007 (Computerworld) -- Call it the case of hiring a fox to guard the hen house. A senior database administrator at a subsidiary of Fidelity National Information Services Inc. who was responsible for defining and enforcing data access rights at the company instead took data belonging to about 2.3 million consumers and sold it to a data broker.

The broker in turn sold a subset of the data to other marketing companies. The stolen data included names, addresses, birth dates, bank account and credit card information, the company said in a statement released today.

For the moment at least, it appears that the companies that bought the information have mainly used the data to send marketing solicitations to affected individuals, Fidelity said in its statement.

Fidelity National Information Services describes itself as a provider of core financial institution processing, card issuer and transaction processing services, mortgage loan processing and related information products and outsourcing services to financial institutions.
An Insider Tale

- Certegy – public company (Jacksonville, Fla.)
  - Check authorization & check cashing services

- Senior DBA sold 8.5 million customer records to data broker
  - Names, addresses, birth dates, bank account & credit card info – was paid $580,000

- Data theft came to light after retailer reported correlation between transactions and receipt of external marketing offers by its customers
  - Called in U.S. Secret Service, which found data had come from separate company owned by Certegy DBA
  - “Why did it take Certegy more than five years to find out that confidential consumer information was being sucked out of its database?” (*St. Petersburg Times*)

- Settled class-action suit for $4 million
  - Plus $975,000 in fines from Attorney General
  - Plus mandatory security audit every year
  - Plus 2 years of credit monitoring services ($180 per customer)

- Rogue DBA sentenced to nearly 6 years in prison
Insider’s Tale

“While it took years for his employer to develop its sophisticated computer code, it allegedly only took Samarth Agrawal days to steal it.”

U.S. Attorney Preet Bharara

- Trader for Wall Street firm
- Had access to codes for proprietary trading algorithms (IP)
- Stole data shortly before resigning to join another firm
- Theft only noticed after employee resigned
- What IT Security missed
  - Unauthorized access to sensitive data
    - Policy in place but no ability to enforce
  - After-hours access – did not review logs
  - Credentials not sufficient – need continuous real-time monitoring
Insider’s Tale – cont.

“While it took years for his employer to develop its sophisticated computer code, it allegedly only took Samarth Agrawal days to steal it.”  
U.S. Attorney Preet Bharara

• Trader for Wall Street firm
• Had access to codes for proprietary trading algorithms (IP)
• Stole data shortly before resigning to join another firm
• Theft only noticed after employee resigned
• What IT Security missed
  • Unauthorized access to sensitive data
    • Policy in place but no ability to enforce
  • After-hours access – did not review logs
  • Credentials not sufficient – need continuous real-time monitoring
Another Insider Tale

Database Administrator Sentenced to Fed Time fro Insider Breach (7/6/10)

The "insider threat" is a favorite term of vendors pushing access monitoring solutions, but research has shown malicious insider breaches are not the most common ways for data to leak out of your organization.

Still, when they happen, the results can be serious, both for the business and the person who gets caught. A former senior database administrator for GEXA Energy in Houston found that out the hard way today when he was sentenced to 12 months in prison for hacking into his former employer's computer network.

Steven Jinwoo Kim, 40, of Houston pleaded guilty on Nov. 16, 2009, to one count of intentionally accessing a protected computer without authorization and recklessly causing damage. In addition to a year in prison, he was also ordered to pay $100,000 in restitution to GEXA Energy and to serve three years of supervised release when his prison term is completed.

According to court documents, GEXA Energy fired Kim Feb. 5, 2008, from his job as DBA and revoked all his administrative rights and access to the company's computer network. As part of his guilty plea, Kim admitted that in the early hours of April 30, 2008, he used his home computer to connect to the company's network and a database that contained information on some 150,000 customers.

Once connected to the network, Kim wreaked havoc by inputting various Oracle database commands, and also copied and saved to his home computer a database file containing personal information on GEXA Energy customers, including names, billing addresses, social security numbers, dates of birth and drivers license numbers.

All totaled, Kim's actions caused a $100,000 loss to GEXA Energy, according to court documents.

Key Business Drivers for DAM: Database Activity Monitoring

1. **Prevent data breaches**
   - Mitigate external and internal threats
   - Protect customer data & corporate secrets (IP)

2. **Ensure data integrity/governance**
   - Prevent unauthorized changes to sensitive data (business user vs. privileged user)

3. **Reduce cost of compliance**
   - Automated, continuous and centralized controls
     - Across DBMS platforms and applications
     - Across SOX, PCI, SAS70, …
   - Simplify processes
Growing Compliance Mandates

- Explosion in successful breaches has resulted in growing regulation of sensitive data
  - SOX
  - HIPPA
  - PCI DSS
  - State-specific data privacy laws
  - Gramm-Leach-Bliley

- U.S. corporations often subject to international regulations
  - EU Data Privacy Directive and supporting local laws
  - C-SOX
  - FIEL
  - etc.
And If You Aren’t Covered by Well Known Regulations….

Common Law:
Due to Common Law precedent, the management of publicly owned firms are expected to exercise due care in protecting the assets under their care. This includes putting in place reasonable controls to protect sensitive data, consistent with industry practices. Failure to implement reasonable (“best practices”) constitutes negligent behavior, which can result in negative legal outcomes.
Impact to Your Organization

- Failed audit causes loss of customer trust and brand reputation
- Failure to comply with regulations have high financial penalties
- Ensuring compliance minimizes the risks of fraud ***
- High profile data breaches can be expensive

The average total cost of a data breach rose from $6.65 million in 2008 to $6.75 million in 2009

Average enterprise is subject to 100s of regulations
***Good Compliance ≠ Good Security

e.g. Heartland Payment Systems was PCI compliant; certified 4/08

Good Security = Good Compliance
Database Security is hard - mix of stakeholders

SECURITY OPERATIONS
- Real-time policies
- Secure audit trail
- Data mining & forensics

COMPLIANCE AUDIT
- Separation of duties
- Best practices reports
- Automated controls

APPLICATION & DATABASE
- Minimal impact
- Change management
- Performance optimization

Need 100% database activity visibility & a unified view also increases the challenge
Complexity of Environment increases the challenge

- Large number of databases makes it challenging.
- Support for heterogeneous DBMS’s making it difficult to protect databases uniformly.
- Lack of understanding the business data, especially not knowing where private/sensitive data is stored.
- Providing excessive data access to users has always been a common practice across many IT organizations.
- Dealing with multiplexed connections from applications makes it difficult to differentiate a user from a hacker.
Is your organization asking these questions? – they should be!

– Are there unauthorized programs changing data?
– Who is changing database schemas?
– Who is extracting credit card data?
– What are DBAs or outsourced staff doing to the databases?
– What data is being accessed from which network node?
– What data is being accessed by which application?
– What are the access patterns based on time of day?
– What database errors are being generated?
– What is the exposure to sensitive objects?
– Where is my sensitive data located and who is using it (including priv. users)?
– When is someone attempting an SQL injection attack?
– How many failed login attempts have occurred?
– How is data being accessed?
– How can I enforce access & change control at the database object level?
– How do I check for vulnerabilities & lock-down database configurations?
– How do I reduce costs by automating & centralizing compliance controls?
Growing List of Database Concerns

- How do we know that DBAs and other privileged users aren't abusing their privileges?
- In other words, who watches the watchers?
- Super users (SYSADM, DBADM):
  - Can do anything to any data.
  - Database auditing can be used to verify the integrity and accuracy of the DBA group.
- How do we immediately alert Security whenever:
  - There are more than 3 failed logins?
  - Someone changes a critical table in PeopleSoft (fraud?)
  - There's suspicious access from the AppServer account and not from the AppServer IP?
- How do I prevent outsourced personnel from viewing sensitive data?
- We've had lots of acquisitions - where the *&x%* is our sensitive data?
- Was the actual change made to the data/database consistent with the approved change?
- How do I know if my database is patched & configured securely?
- How do I produce all these *&x%* SOX & PCI reports?
Top data security question from auditors

“Can you prove that privileged users have not jeopardized the integrity and/or privacy of your sensitive data?”
Perimeter Defenses No Longer Sufficient

“A fortress mentality will not work in cyber. We cannot retreat behind a Maginot Line of firewalls.”

(The fortification system successfully dissuaded a direct attack. However, it was strategically ineffective)

- William J. Lynn III,
  U.S. Deputy Defense Secretary

Outsourcing

Web-Facing Apps

Legacy App Integration/SOA

Employee Self-Service, Partners & Suppliers

Insiders
(DBAs, developers, outsourcers, etc.)

Stolen Credentials
(Zeus, etc.)
### Monitoring Database Activity for Security and Compliance

Requires a solution that understands SQL activities.

<table>
<thead>
<tr>
<th>Audit Requirements</th>
<th>SOX</th>
<th>PCI DSS</th>
<th>ISO 27002</th>
<th>Data Privacy &amp; Protection Laws</th>
<th>NIST SP 800-53 (FISMA)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Access to Sensitive Data (Successful/Failed SELECTs)</td>
<td></td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>2. Schema Changes (DDL) (Create/Drop/Alter Tables, etc.)</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>3. Data Changes (DML) (Insert, Update, Delete)</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4. Security Exceptions (Failed logins, SQL errors, etc.)</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>5. Accounts, Roles &amp; Permissions (DCL) (GRANT, REVOKE)</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
</tbody>
</table>

**DDL** = Data Definition Language (aka schema changes)

**DML** = Data Manipulation Language (data value changes)

**DCL** = Data Control Language
Common Security Technologies are Insufficient for DB’s

Perimeter defenses, IDS/IPS, WAF, etc.
- Lack awareness of database-specific policies & protocols
- Doesn’t protect against internal threats

Traditional Data Loss Prevention (DLP)
- Catches sensitive data as it leaves via email or USB
- Can’t stop data theft at the source – in the data center
- Lacks database-focused monitoring, analytics & blocking
- No knowledge about DBMS commands, vulnerabilities & structures

Database encryption – doesn’t protect against
- Hackers who hijack Web/application servers
- Rogue administrators with access to encryption keys

Security Information & Event Management (SIEM)
- Relies on imported DBMS log data
- No real-time monitoring at data level to detect unauthorized access
- Lack database-focused analytics & blocking

Vulnerability Assessments
- Do not look at database activity real-time; lacks database table/object knowledge
- Snapshot “vulnerability” solution; doesn’t look at user behavior

Native Database Auditing/Logging
- Requires Database Changes
- Impacts Performance
- No Separation of Duties
Why SIEM Isn’t Sufficient

- Dependent on native logs
  - Can't capture DBMS activity on their own

- SQL access is much "richer" than UNIX/Windows/Cisco logs
  - DDL (Create/Drop/Alter Tables)
  - DML (Insert/Update/Delete)
  - DCL (Grants/Revokes)
  - SELECTs (read operations)
  - SQL Exceptions (SQL errors, failed logins -> SQL Injection)

- Verizon Breach Report
  - **Database servers accounted for 92% of all records breached**
  - “A much smaller percentage of hacks targeted routers, switches, and other network devices.”
What Database Audit Tools are Enterprises Using Today?

- Native Database Logging
- Perl /Unix Scripts /C+
- Scrape and parse the data
- Central repositories to report

Create reports → Manual review → Manual remediation dispatch and tracking
Why These Processes + SIEM aren’t Practical (Nor Effective)

- Significant performance overhead to provide granular information required by auditors (e.g., audit all SELECTs for PCI-DSS)
  - Which table, from which IP, using which command, which program, etc.

- No separation of duties – can easily be disabled by DBAs or hacked

- Not real-time (batch approach)

- Inconsistent audit policies across DBMS platforms (adds complexity)

- Doesn't provide proactive real-time security (review every 3 months???) –

- No blocking capability based on policy violation; no preventative controls

- Doesn’t track application users in connection pooling, generic service accounts environments for potential fraud (PeopleSoft, SAP, Oracle Financials, etc.)

- Massive storage requirements

- Still need to write scripts to filter log data to find anomalies, error prone

- Still need to write custom scripts to produce compliance reports, error prone
# Ten Database Activities Enterprises Need to Monitor

**Jeffrey Wheatman**

Most enterprises are paying too little attention to the very real security risks associated with their databases. Auditors, security and risk professionals, and data owners need to watch for telltale behaviors that may indicate serious database security problems.

---

<table>
<thead>
<tr>
<th>Privileged Users</th>
<th>Access or changes to data</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Access via inappropriate or unapproved channels</td>
</tr>
<tr>
<td></td>
<td>Schema modifications</td>
</tr>
<tr>
<td></td>
<td>Addition or modification of accounts</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>End Users</th>
<th>Access to excessive or unneeded data</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Data access outside standard hours</td>
</tr>
<tr>
<td></td>
<td>Access via inappropriate or nonapproved channels</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>DevelopersSys. Admins Analysts</th>
<th>Access to live production systems</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Nonapproved changes to databases or applications</td>
</tr>
<tr>
<td></td>
<td>Out-of-cycle patching of production systems</td>
</tr>
</tbody>
</table>

| IT Ops | |
|--------||

SELECTs, DML, DDL, Grants, Revoke

SELECTs

SELECTs, DML, DDL

DDL, DB Vulnerabilities
Ten Database Activities Enterprises Need to Monitor

Jeffrey Wheatman

Most enterprises are paying too little attention to the very real security risks associated with their databases. Auditors, security and risk professionals, and data owners need to watch for telltale behaviors that may indicate serious database security problems.

<table>
<thead>
<tr>
<th>Privileged Users</th>
<th>Access or changes to data</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Access via inappropriate or unapproved channels</td>
</tr>
<tr>
<td></td>
<td>Schema modifications</td>
</tr>
<tr>
<td></td>
<td>Addition or modification of accounts</td>
</tr>
<tr>
<td>End Users</td>
<td>Access to excessive or unneeded data</td>
</tr>
<tr>
<td></td>
<td>Data access outside standard hours</td>
</tr>
<tr>
<td></td>
<td>Access via inappropriate or nonapproved channels</td>
</tr>
<tr>
<td>Developers Sys. Admins Analysts</td>
<td>Access to live production systems</td>
</tr>
<tr>
<td>IT Ops</td>
<td>Nonapproved changes to databases or applications</td>
</tr>
<tr>
<td></td>
<td>Out-of-cycle patching of production systems</td>
</tr>
</tbody>
</table>
Non-Invasive, Real-Time Database Security & Monitoring

- Continuously monitors all database activities (including local access by superusers)
- Heterogeneous, cross-DBMS solution
- Does not rely on native DBMS logs
- Minimal performance impact (2-3%)
- No DBMS or application changes

- Supports Separation of Duties
- Activity logs can’t be erased by attackers or DBAs
- Automated compliance reporting, sign-offs & escalations (SOX, PCI, NIST, etc.)
- Granular, real-time policies & auditing
  - Who, what, when, where, how
Scalable Enterprise-Wide Architecture

Integration with LDAP/AD, IAM, change management, SIEM, archiving, …
Addressing the Complete Database Security and Compliance Lifecycle

- Prevent cyberattacks
- Monitor & block privileged users
- Detect application-layer fraud
- Enforce change controls
- Real-time alerts
- Control firecall IDs
- SIEM integration

- Automated & centralized controls
- Cross-DBMS audit repository
- Preconfigured policies/reports
- No database changes
- Minimal performance impact
- Sign-off management
- Entitlement reporting

- Find & classify sensitive data
- Continuously update security policies
- Discover embedded malware & logic bombs

- Assess static and behavioral database vulnerabilities
- Configuration auditing
- Preconfigured tests based on best practices standards (STIG, CIS, CVE)
Proactively Preventing Policy Violations in Real-time

1. Privileged Users
   - Issue SQL
   - Connection terminated

2. Production Traffic
   - SQL
   - Hold SQL

3. Check Policy
   - On Appliance

4. Policy Violation: Drop Connection

- Cross-DBMS policies
- Block privileged user actions
- No database changes
- No application changes
- Without risk of inline appliances that can interfere with application traffic

Oracle, DB2, MySQL, Sybase, etc.

Outsourced DBA

Application Servers

Monitor & Enforce

Session Terminated

Outsourced DBA

Oracle Database 10g Express Edition Release 10.2.0.1.0 - Production

```
[root@osprey ~]# sqlplus system
SQL*Plus: Release 10.2.0.1.0 - Production on Tue May 27 01:13:32 20
Copyright (c) 1982, 2005, Oracle. All rights reserved.
Enter password:
Connected to:
Oracle Database 10g Express Edition Release 10.2.0.1.0 - Production

SQL> select * from creditcard;
SELECT * FROM CREDITCARD
* 
ERROR at line 1:
ORA-03113: end-of-file on communication channel
```

Session Terminated
Control Unauthorized Actions by Insiders

Application Server 10.10.9.244

Database Server 10.10.9.56

Non-App Source AppUser Connection

Category: Security
Classification: Breach
Severity: MED

Not:
- Server IP
- Client IP
- Client MAC
- DB Name
- DB User

And/or Group:
- Production Servers
- Authorized Client IPs

Field Name:
- INVENTORY
- DROP TABLE

Action:
- ALERT PER MATCH

Notification Type: MAIL
Mail User: marc.gamache@guardium.com

Subject: (c1) SQLGUARD ALERT Alert based on rule ID 20267 [non-App Source AppUser Connection]
Category: security
Classification: Breach
Rule # 20267 [non-App Source AppUser Connection]

Request Info:
- Session start: 2009-04-15 06:59:03
- Server Type: ORACLE
- Client IP: 192.168.20.160
- Server IP: 172.16.2.152
- Client Port: 11787
- Server Port: 1521
- Net Protocol: TCP
- DB Protocol: TNS
- DB Protocol Version: 3.8
- DB User: APPUSER
- Application User Name: IDBC THIN CLIENT
- Authorization Code: 1
- Request Type: SQL_LANG
- Last Error: SQL: select * from EmployeeTable
User Quarantine

- ... in response to any policy violation
- Quarantines user access until specified date
  - Eliminates “cat and mouse” with perps
  - Gives time to investigate incident
- Use case example: Quarantine any user attempting to access any object in the vulnerable objects group on the financial server - that does not originate from the financial application

Powerful complement to real-time blocking; prevents repeated attacks (and resulting investigations) when a clear violation has been detected
Identifying Fraud at the Application Layer

- **Issue**: Application server uses generic service account to access DB
  - Doesn’t identify *who* initiated transaction (connection pooling)

- **Solution**: track access to application user associated with specific SQL commands
  - No changes required to applications
  - Deterministic tracking of user IDs
    - Does not rely on time-based “best-guess”

<table>
<thead>
<tr>
<th>DB User Name</th>
<th>Application User</th>
<th>SQL</th>
</tr>
</thead>
<tbody>
<tr>
<td>APPUSER</td>
<td>joe</td>
<td><code>select * from EmployeeRoleView where UserName=?</code></td>
</tr>
<tr>
<td>APPUSER</td>
<td>joe</td>
<td><code>select * from EmployeeTable</code></td>
</tr>
<tr>
<td>APPUSER</td>
<td>marc</td>
<td><code>insert into EmployeeTable values (?,?,?,?,?,?,?)</code></td>
</tr>
</tbody>
</table>
Change Management Monitoring Controls

- Privileged user changes
- Integrate with change management systems (Remedy, Peregrine, etc)
- Automatically documents the change control process
- Provide workflow (sign off, escalations, comments, etc.)
- Provides insight into changes to:
  - Operating System (configuration files, registry settings, environment variables, etc.)
  - Database structure (DDL)
  - Content (DML)
- Report on issues that may compromise data integrity
- Real time alert on critical changes
- Real time prevention for unauthorized changes
- Incorporate detective and preventative controls as control points within your environment
Why Automated Change Control Matters

“... auditors have become more aggressive in evaluating configuration control. Unless the configuration of systems can be verified ... neither the integrity nor the confidentiality of the data they contain can be trusted.”

*Using Configuration Auditing for Compliance Control*, Gartner

“Effective, automated controls streamline the audit process ... Once validated, auditors can rely on the existence of automated controls as evidence of compliance.” *Compliance Week*

“If we use quality, standards-based processes in IT, compliance becomes a given ... [and] can generate tangible benefits that improve systems, response times and credibility.”

*CIO Decisions*
Enforcing Change Control Policies

Tag DBA actions with ticket IDs

Compare observed changes to approved changes

Identify unauthorized changes (red) or changes with invalid ticket IDs
### Sensitive Object Usage Report for 6 Months

<table>
<thead>
<tr>
<th>Object Name</th>
<th>SQL Verb</th>
<th>DB User Name</th>
<th>Client IP</th>
<th>Network Protocol</th>
<th>Server IP</th>
<th>Database Name</th>
<th>OS User</th>
<th>Client Host Name</th>
<th>Source</th>
<th>Program</th>
<th>Depth</th>
<th>Total access</th>
</tr>
</thead>
<tbody>
<tr>
<td>creditcard</td>
<td>DROP TABLE</td>
<td>BENJI</td>
<td>10.10.9.250</td>
<td>WINDOWS NAMED PIPES</td>
<td>10.10.9.250</td>
<td>CARD_DATA</td>
<td>BENJI</td>
<td>WORKGROUP/BENJI_LAPTOP</td>
<td>SGLPLUS</td>
<td>0</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>creditcard</td>
<td>ALTER TABLE</td>
<td>BENJI</td>
<td>10.10.9.250</td>
<td>WINDOWS NAMED PIPES</td>
<td>10.10.9.250</td>
<td>CARD_DATA</td>
<td>BENJI</td>
<td>WORKGROUP/BENJI_LAPTOP</td>
<td>SGLPLUS</td>
<td>0</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>creditcard</td>
<td>GRANT</td>
<td>BENJI</td>
<td>10.10.9.250</td>
<td>WINDOWS NAMED PIPES</td>
<td>10.10.9.250</td>
<td>CARD_DATA</td>
<td>BENJI</td>
<td>WORKGROUP/BENJI_LAPTOP</td>
<td>SGLPLUS</td>
<td>0</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>creditcard</td>
<td>CREATE TABLE</td>
<td>BENJI</td>
<td>10.10.9.250</td>
<td>WINDOWS NAMED PIPES</td>
<td>10.10.9.250</td>
<td>CARD_DATA</td>
<td>BENJI</td>
<td>WORKGROUP/BENJI_LAPTOP</td>
<td>SGLPLUS</td>
<td>0</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>creditcard</td>
<td>INSERT</td>
<td>BENJI</td>
<td>10.10.9.250</td>
<td>WINDOWS NAMED PIPES</td>
<td>10.10.9.250</td>
<td>CARD_DATA</td>
<td>BENJI</td>
<td>WORKGROUP/BENJI_LAPTOP</td>
<td>SGLPLUS</td>
<td>0</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>creditcard</td>
<td>DROP TABLE</td>
<td>DSMITH</td>
<td>192.168.20.222</td>
<td>TCP</td>
<td>192.168.200.108</td>
<td>CARD_DATA</td>
<td>JDIPIETRO</td>
<td>WORKGROUP/JOE</td>
<td>SGLPLUS</td>
<td>0</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>creditcard</td>
<td>DROP TABLE</td>
<td>JDIPIETRO</td>
<td>Joe</td>
<td>TCP</td>
<td>192.168.200.108</td>
<td>CARD_DATA</td>
<td>JDIPIETRO</td>
<td>WORKGROUP/JOE</td>
<td>SGLPLUS</td>
<td>0</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>creditcard</td>
<td>DROP TABLE</td>
<td>DSMITH</td>
<td>Joe</td>
<td>TCP</td>
<td>192.168.200.108</td>
<td>CARD_DATA</td>
<td>JDIPIETRO</td>
<td>WORKGROUP/JOE</td>
<td>SGLPLUS</td>
<td>0</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>creditcard</td>
<td>DROP TABLE</td>
<td>RODRIGO</td>
<td>192.168.20.58</td>
<td>TCP</td>
<td>192.168.200.108</td>
<td>CARD_DATA</td>
<td>ROOT</td>
<td>RODRIGO-LAP</td>
<td>JDBC THIN CLIENT</td>
<td>0</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>creditcard</td>
<td>DROP TABLE</td>
<td>BENJI</td>
<td>192.168.20.178</td>
<td>TCP</td>
<td>192.168.200.108</td>
<td>CARD_DATA</td>
<td>BENJI</td>
<td>WORKGROUP/BENJI_LAPTOP</td>
<td>SGLPLUS</td>
<td>0</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>creditcard</td>
<td>DROP TABLE</td>
<td>LARRY</td>
<td>192.168.20.119</td>
<td>TCP</td>
<td>192.168.200.108</td>
<td>CARD_DATA</td>
<td>LARRY?METAXOTOS</td>
<td>WORKGROUP/MURCELAGO</td>
<td>SGLPLUS</td>
<td>0</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>creditcard</td>
<td>DROP TABLE</td>
<td>JM</td>
<td>192.168.20.110</td>
<td>TCP</td>
<td>192.168.200.108</td>
<td>CARD_DATA</td>
<td>JANDREWS</td>
<td>GUARDIAN/JANDREWS</td>
<td>SGLPLUS</td>
<td>0</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>creditcard</td>
<td>DROP TABLE</td>
<td>JOE</td>
<td>192.168.20.222</td>
<td>TCP</td>
<td>192.168.200.108</td>
<td>CARD_DATA</td>
<td>JDIPIETRO</td>
<td>WORKGROUP/JOE</td>
<td>SGLPLUS</td>
<td>0</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>creditcard</td>
<td>DROP TABLE</td>
<td>DSMITH</td>
<td>192.168.200.108</td>
<td>TCP</td>
<td>192.168.200.108</td>
<td>CARD_DATA</td>
<td>ORACLE10</td>
<td>GEMODB</td>
<td>SGLPLUS</td>
<td>0</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>creditcard</td>
<td>DROP TABLE</td>
<td>JOE</td>
<td>192.168.200.108</td>
<td>TCP</td>
<td>192.168.200.108</td>
<td>CARD_DATA</td>
<td>JOE</td>
<td>GEMODB</td>
<td>SGLPLUS</td>
<td>0</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>creditcard</td>
<td>CREATE TABLE</td>
<td>DSMITH</td>
<td>192.168.200.222</td>
<td>TCP</td>
<td>192.168.200.108</td>
<td>CARD_DATA</td>
<td>JDIPIETRO</td>
<td>WORKGROUP/JOE</td>
<td>SGLPLUS</td>
<td>0</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>creditcard</td>
<td>CREATE TABLE</td>
<td>JDIPIETRO</td>
<td>Joe</td>
<td>TCP</td>
<td>192.168.200.108</td>
<td>CARD_DATA</td>
<td>JDIPIETRO</td>
<td>WORKGROUP/JOE</td>
<td>SGLPLUS</td>
<td>0</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>creditcard</td>
<td>CREATE TABLE</td>
<td>DSMITH</td>
<td>Joe</td>
<td>TCP</td>
<td>192.168.200.108</td>
<td>CARD_DATA</td>
<td>JDIPIETRO</td>
<td>WORKGROUP/JOE</td>
<td>SGLPLUS</td>
<td>0</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>creditcard</td>
<td>INSERT</td>
<td>DSMITH</td>
<td>192.168.20.222</td>
<td>TCP</td>
<td>192.168.200.108</td>
<td>CARD_DATA</td>
<td>JDIPIETRO</td>
<td>WORKGROUP/JOE</td>
<td>SGLPLUS</td>
<td>0</td>
<td>1</td>
<td></td>
</tr>
</tbody>
</table>

**SQL drill detail with values (option to store values)**

```sql
select * from creditcard where cardnumber like '6011%'
```

* Records: 1 to 7 of 7

---

**SQL drill detail without values**

```sql
select * from creditcard where cardnumber like ?
```

* Records: 1 to 2 of 2

---

© 2010 IBM Corporation
Control Privileges Through Entitlement Reports

Entitlement reports provide information such as:

- Who granted what rights
- What users and roles have system and admin privileges
- Object privileges by account
- Account security also needs to validate entitlements granted within the database
- Don’t overlook these key areas
Tracking Privileged Users Who "su"

Challenge: How do you track users who 'switch' accounts (perhaps to cover their tracks)?

- Native database logging/auditing & SIEM tools can't capture OS user information
- Other database monitoring solutions only provide OS shell account that was used

User activity

```
login as: joe
[joe@192.168.30.152 ~]$ su - oracle
Password:
bash-3.00# sqlplus system
```

SQL*Plus: Release 10.2.0.1.0 - Production on Tue Apr 14 15:17:39 2009
Copyright (c) 1982, 2005, Oracle. All rights reserved.

Enter password:

Connected to:
Oracle Database 10g Express Edition Release 10.2.0.1.0 - Production

```
SQL> insert into AppUser.EmployeeTable values (1001,6,'Joe','Smith','Salary','Bonus',500000,1);
1 row created.
```

What Guardium Shows You

<table>
<thead>
<tr>
<th>DB User Name</th>
<th>Sql</th>
</tr>
</thead>
<tbody>
<tr>
<td>SYSTEM</td>
<td>insert into AppUser.EmployeeTable values (?,?,?,?,?</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>DB User Name</th>
<th>ShellAccount</th>
<th>Sql</th>
</tr>
</thead>
<tbody>
<tr>
<td>SYSTEM</td>
<td>ORACLE</td>
<td>insert into AppUser.EmployeeTable values (?,?,?,?,?</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>DB User Name</th>
<th>ShellAccount</th>
<th>OSUser</th>
<th>Sql</th>
</tr>
</thead>
<tbody>
<tr>
<td>SYSTEM</td>
<td>ORACLE</td>
<td>joe</td>
<td>insert into AppUser.EmployeeTable values (?,?,?,?,?</td>
</tr>
</tbody>
</table>
Good Security Involves Individual Accountability

- DBAs don’t have to deviate from their standard operational practices
- You get visibility into ALL individuals, even with generic accounts…
- Key question:
  - Do you have processes that can be tied back to a single individual user?
**Weekly Database Change Management Process**
Audit process execution began 1/27/09 2:59 PM

### Distribution Status:

<table>
<thead>
<tr>
<th>Receiver</th>
<th>Status</th>
<th>Action Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>Joe (Joe DiPietro)</td>
<td>Signed</td>
<td>Review and Sign</td>
</tr>
<tr>
<td>Role dba: User TSE (TSE Guardium)</td>
<td>Viewed</td>
<td>Review Only</td>
</tr>
<tr>
<td>Role dba: User audit (audit audit)</td>
<td>Viewed</td>
<td>Review Only</td>
</tr>
<tr>
<td>Role dba: User user1 (accelerator account)</td>
<td>Viewed</td>
<td>Review Only</td>
</tr>
<tr>
<td>Role dba: User Larry (Larry Metaxotos)</td>
<td>Viewed</td>
<td>Review Only</td>
</tr>
<tr>
<td>Role infsec: User TSE (TSE Guardium)</td>
<td>Signed</td>
<td>Review and Sign</td>
</tr>
<tr>
<td>Role infsec: User audit (audit audit)</td>
<td>Signed</td>
<td>Review and Sign</td>
</tr>
<tr>
<td>Role infsec: User user1 (accelerator account)</td>
<td>Signed</td>
<td>Review and Sign</td>
</tr>
<tr>
<td>Role infsec: User Marc (Marc Gamache)</td>
<td>Signed</td>
<td>Review and Sign</td>
</tr>
<tr>
<td>Role audit: User TSE (TSE Guardium)</td>
<td>Signed</td>
<td>Review and Sign</td>
</tr>
<tr>
<td>Role audit: User audit (audit audit)</td>
<td>Signed</td>
<td>Review and Sign</td>
</tr>
<tr>
<td>Role audit: User user1 (accelerator account)</td>
<td>Signed</td>
<td>Review and Sign</td>
</tr>
<tr>
<td>Role audit: User dele (Dele Brocklehurst)</td>
<td>Signed</td>
<td>Review and Sign</td>
</tr>
</tbody>
</table>

### Comments:

<table>
<thead>
<tr>
<th>Timestamp</th>
<th>User</th>
<th>Comment for Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>2009-01-27 22:39:33 0</td>
<td>Dale</td>
<td>This week's process is now closed...Thanks for everyone's help!</td>
</tr>
<tr>
<td>2009-01-27 22:35:23 0</td>
<td>Marc</td>
<td>Checked out security assessment, and found an issue. Someone changed a config file without a change ticket. It was the TNS NAMES.ORA file and moved it to the default port 1521. This was resolved and we terminated the contractor.</td>
</tr>
<tr>
<td>2009-01-27 22:31:05 0</td>
<td>Larry</td>
<td>Failed logins on DB2 are OK...We were testing new application scripts and had to work out a few issues.</td>
</tr>
<tr>
<td>2009-01-27 21:28:47 0</td>
<td>Joe</td>
<td>Change Management looks good - Security Assessment is below last week's results, please have security look into details - Failed login report please look at db2 database for unusual activity</td>
</tr>
</tbody>
</table>


**Security Assessment: Security Assessment [ oracle enterprise assessment] Overall Value: 31**

**Classification Process: Discover Sensitive Data [Find SSN Process]**


**Report: SQL Errors report [SQL Errors] Overall Value: 140**
Automated Sign-offs & Escalations for Compliance

- Automates entire compliance workflow
  - Report distribution to oversight team
  - Electronic sign-offs
  - Escalations
  - Comments & exception handling
- Addresses auditors’ requirements to document oversight processes
- Results of audit process stored with audit data in secure audit repository
- Streamlines and simplifies compliance processes
The Enterprise Patching Issue

- Nearly half of companies lack a formal patch management process
- 62% typically take 3 months or more to apply Critical Patch Updates (IOUG)
- Only 18% measure patch success via configuration scanning
- "The least mature areas of patching seem to correlate almost directly with the fastest-growing areas of attacks, such as ... database servers [and] business application servers."

"Patch management is one of the most fundamental functions of IT departments, yet in our research we discovered it remains one of the biggest pain points for many organizations."

Rich Mogull, Securosis

Assess & Harden

http://www.darkreading.com/database_security
http://www.securosis.com/projectquant
What Do You Do if the Database Can’t be Patched?

Typical Database Patch Cycles

How quickly do you apply Oracle Critical Patch Updates?

- 30% Before the next batch is released
- 25% Within 3 to 6 months of release
- 11% Never applied them
- 8% Within a year of release
- 8% Within 9 to 12 months of release
- 10% Within 6 to 9 months of release
- 8% Don’t know/not sure

Data: Security Patching Practices, a February 2009 report detailing a survey of 150 DBAs conducted jointly by the Independent Oracle User Group and Oracle
Are Patch Levels Current?

Results for Security Assessment: Comprehensive Oracle Assessment
Assessment executed 2009-08-20 15:58:49.0

From: 2009-08-19 15:58:49.0
To: 2009-08-20 15:58:49.0
Client IP or IP subnet: Any
Server IP or IP subnet: Any

Test Result History

PASS

FAIL

Oracle Patch Level
oracle - 9.59 - system
(ORACLE)

Fail

No valid patch levels are defined for version: ORACLE '10.2.0.1.0'.

The Oracle instance is not patched to your standard level for this version. We recommend that you upgrade this database to an accepted patch level.
Vulnerability Management Architecture Makes a Difference

- Basic Vulnerability Assessment (VA) = Database tests (patches, etc.)
- Intermediate Vulnerability Assessment = Database + OS tests
- Advanced Vulnerability Assessment = Database + OS tests + User Activity
- Vulnerability Management = Database + OS tests + User Activity + Policy Enforcement
<table>
<thead>
<tr>
<th>Privilege</th>
<th>Test Name</th>
<th>Data Source</th>
<th>P/F</th>
<th>Sev.</th>
<th>Reason</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><strong>No Authorizations To System Level Privileges</strong></td>
<td>ORACLE: oracle - 10.10.9.128</td>
<td>Fail</td>
<td>Critical</td>
<td>Users or roles, other than DBAs, were found with access to 'EXECUTE ANY PROCEDURE', 'GRANT ANY PRIVILEGE', 'GRANT ANY OBJECT PRIVILEGE', 'GRANT ANY ROLE', 'CREATE LIBRARY', 'AUDIT ANY', 'AUDIT SYS', 'IMP_FULL_DATABASE', 'DATA PUMP_IMP_FULL_DATABASE'. Recommendation: System privileges 'EXECUTE ANY PROCEDURE', 'GRANT ANY PRIVILEGE', 'GRANT ANY OBJECT PRIVILEGE', 'GRANT ANY ROLE', 'CREATE LIBRARY', 'AUDIT ANY', 'AUDIT SYS' should only be granted to users or roles with dba privileges. The pre-defined role 'EXP_FULL_DATABASE' has limited dba authority and can be granted the 'EXECUTE ANY PROCEDURE' privilege. The pre-defined role 'IMP_FULL_DATABASE' has limited dba authority and can be granted the 'EXECUTE ANY PROCEDURE', 'GRANT ANY PRIVILEGE' and 'AUDIT ANY' privileges. SELECT ANY DICTIONARY' may be granted to 'DBA', 'OEM_MONITOR', 'OLAP_DBA'.</td>
</tr>
<tr>
<td></td>
<td><strong>No Individual Users With 'Any Table' Privileges</strong></td>
<td>ORACLE: oracle - 10.10.9.128</td>
<td>Fail</td>
<td>Critical</td>
<td>ALTER/BACKUP/COMMENT/CREATE/DROP/FLASHBACK/INSERT/LOCK/SELECT/UNDER UPDATE ANY TABLE found granted to specific users.</td>
</tr>
<tr>
<td></td>
<td><strong>Only DBA Access To Any DBA_view</strong></td>
<td>ORACLE: oracle - 10.10.9.128</td>
<td>Fail</td>
<td>Critical</td>
<td>Some users or roles without DBA or OEM_MONITOR authority have access to 'DBA' tables or views: OLAP_XS_ADMIN, SELECT_CATALOG_ROLE, XDEADMIN. Recommendation: Access to some 'DBA_views' is not restricted. Privileges have been granted to users that do not have DBA authority. This compromises the security and integrity of your database. We recommend to restrict access to these objects.</td>
</tr>
<tr>
<td></td>
<td><strong>Only DBA Access To DBA_ROLES</strong></td>
<td>ORACLE: oracle - 10.10.9.128</td>
<td>Fail</td>
<td>Critical</td>
<td>Some users or roles without DBA authority have access to the table 'DBA_ROLES': OLAP_XS_ADMIN. Recommendation: Access to DBA_ROLES is not restricted. Privileges have been granted to users that do not have DBA authority. This compromises the security and integrity of your database. We recommend to restrict access to this object</td>
</tr>
</tbody>
</table>
## Controlling Inappropriate Privileges

<table>
<thead>
<tr>
<th>Priv</th>
<th>Test Name</th>
<th>Datasource</th>
<th>PF</th>
<th>Sev.</th>
<th>Reason</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Access To The UTL_FILE Package is restricted</strong></td>
<td>ORACLE: Oracle EE - Joe</td>
<td><strong>Fail</strong></td>
<td>Major</td>
<td>Found Exec UTL_FILE privilege granted to public.</td>
<td></td>
</tr>
<tr>
<td><strong>LOG_ARCHIVE_DUPLEX_DEST Set</strong></td>
<td>ORACLE: Oracle EE - Joe</td>
<td><strong>Fail</strong></td>
<td>Major</td>
<td>Parameter 'LOG_ARCHIVE_DUPLEX_DEST' is not set.</td>
<td></td>
</tr>
<tr>
<td><strong>MAX_ENABLED_ROLES is not greater than 30</strong></td>
<td>ORACLE: Oracle EE - Joe</td>
<td><strong>Fail</strong></td>
<td>Major</td>
<td>Parameter 'MAX_ENABLED_ROLES' with a value of '150' has been obsoleted for version 10.2.</td>
<td></td>
</tr>
<tr>
<td><strong>No 'Catalog' Role Assignments</strong></td>
<td>ORACLE: Oracle EE - Joe</td>
<td><strong>Fail</strong></td>
<td>Major</td>
<td>Some users or roles other than predefined dba or roles have been granted default roles: SH, OLAPSYS, PERFSTAT, IX.</td>
<td></td>
</tr>
<tr>
<td><strong>No Authority To Create Libraries</strong></td>
<td>ORACLE: Oracle EE - Joe</td>
<td><strong>Fail</strong></td>
<td>Major</td>
<td>Some users or roles without DBA or IMP_FULL_DATABASE authority have CREATE LIBRARY privileges: MDSYS, DMSYS, EXFSYS, ORDSYS, ORPLUGINS, XDB.</td>
<td></td>
</tr>
<tr>
<td><strong>No Roles With The Admin Option</strong></td>
<td>ORACLE: Oracle EE - Joe</td>
<td><strong>Fail</strong></td>
<td>Major</td>
<td>Found roles granted WITH ADMIN option.</td>
<td></td>
</tr>
</tbody>
</table>
Database Specific Tests

FBI-USSS Advisory (Feb. 2009)
Disable harmful stored procedure calls including:
Disable access to the xp_cmdshell functions within MS SQL Server.

Conf. xp_cmdshell context is On
SYBASE: Sybase (ocean) Pass Major Parameter: 'xp_cmdshell context' is '1'.

Recommendation: The XP_CMDSHELL CONTEXT parameter is set to 1, as recommended
Create and Maintain a Secure Configuration Baseline

Designed to identify common vulnerability issues that may be encountered in database environments

<table>
<thead>
<tr>
<th>Environment Variable</th>
<th>File Pattern</th>
<th>1h</th>
<th>10m</th>
</tr>
</thead>
<tbody>
<tr>
<td>SORACLE_HOME/jre/<em>bin/</em></td>
<td>File Pattern</td>
<td>1h</td>
<td>10m</td>
</tr>
<tr>
<td>SORACLE_HOME/network/admin/*</td>
<td>File Pattern</td>
<td>10m</td>
<td>1h</td>
</tr>
<tr>
<td>SORACLE_HOME/oui/bin/*.*so</td>
<td>File Pattern</td>
<td>1h</td>
<td>10m</td>
</tr>
<tr>
<td>SORACLE_HOME/Apache/Apache/bin/*.*so</td>
<td>File Pattern</td>
<td>1h</td>
<td>10m</td>
</tr>
<tr>
<td>SORACLE_HOME/Apache/Apache/fcgi-bin/*</td>
<td>File Pattern</td>
<td>1h</td>
<td>10m</td>
</tr>
<tr>
<td>SORACLE_HOME/Apache/Jsdk/bin/servletrunner</td>
<td>File Pattern</td>
<td>1h</td>
<td>10m</td>
</tr>
<tr>
<td>SORACLE_HOME/Apache/open_ssl/bin/openssl*</td>
<td>File Pattern</td>
<td>1h</td>
<td>10m</td>
</tr>
<tr>
<td>SORACLE_HOME/assistants/dbma/mep.cfg</td>
<td>File Pattern</td>
<td>1h</td>
<td>10m</td>
</tr>
<tr>
<td>SORACLE_HOME/bin/*</td>
<td>File Pattern</td>
<td>1h</td>
<td>10m</td>
</tr>
<tr>
<td>SORACLE_HOME/ds/bin/*</td>
<td>File Pattern</td>
<td>1h</td>
<td>10m</td>
</tr>
<tr>
<td>SORACLE_HOME/javavn/admin/*.*so</td>
<td>File Pattern</td>
<td>1h</td>
<td>10m</td>
</tr>
<tr>
<td>SORACLE_HOME/jdk/bin/*</td>
<td>File Pattern</td>
<td>1h</td>
<td>10m</td>
</tr>
<tr>
<td>SORACLE_HOME/jdk/jre/bin/*</td>
<td>File Pattern</td>
<td>1h</td>
<td>10m</td>
</tr>
<tr>
<td>SORACLE_HOME/jre/<em>bin/</em></td>
<td>File Pattern</td>
<td>1h</td>
<td>10m</td>
</tr>
<tr>
<td>SORACLE_HOME/focmmon/nls/builder/builder</td>
<td>File Pattern</td>
<td>1h</td>
<td>10m</td>
</tr>
<tr>
<td>SORACLE_HOME/olap/cv/*</td>
<td>File Pattern</td>
<td>1h</td>
<td>10m</td>
</tr>
<tr>
<td>SORACLE_HOME/soap/bin/*</td>
<td>File Pattern</td>
<td>1h</td>
<td>10m</td>
</tr>
<tr>
<td>SORACLE_HOME/syndication/bin/*</td>
<td>File Pattern</td>
<td>1h</td>
<td>10m</td>
</tr>
<tr>
<td>SORACLE_HOME/sysman/admin/OMSRepository/Constraints.properties</td>
<td>File Pattern</td>
<td>1h</td>
<td>10m</td>
</tr>
<tr>
<td>SORACLE_HOME/sysman/config/*properties</td>
<td>File Pattern</td>
<td>1h</td>
<td>10m</td>
</tr>
<tr>
<td>SORACLE_HOME/dxd/admin/xml.properties</td>
<td>File Pattern</td>
<td>1h</td>
<td>10m</td>
</tr>
<tr>
<td>ORACLE_BASE</td>
<td>Environment Variable</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ORACLE_HOME</td>
<td>Environment Variable</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ORACLE_SID</td>
<td>Environment Variable</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TNS_ADMIN</td>
<td>Environment Variable</td>
<td>10m</td>
<td></td>
</tr>
</tbody>
</table>

Select statements:
- select * from dba_db_links
- select * from sys.link$
Control Data Leakage

Should my customer service rep view 99 records in an hour?

<table>
<thead>
<tr>
<th>DB User Name</th>
<th>Sql</th>
<th>Records</th>
</tr>
</thead>
<tbody>
<tr>
<td>STEVE</td>
<td>select * from ar.creditcard where i&gt;? and i&lt;? 4</td>
<td></td>
</tr>
<tr>
<td>HARRY</td>
<td>select * from ar.creditcard where i&gt;?</td>
<td>4</td>
</tr>
<tr>
<td>JOE</td>
<td>select * from ar.creditcard where i?&lt;</td>
<td>99</td>
</tr>
</tbody>
</table>

Is this normal behavior?

What exactly did Joe see?

Assess & Harden
Find & Classifying Sensitive Data

- Discover databases
- Discover sensitive data
- Policy-based actions
  - Alerts
  - Add to group of sensitive objects
Identifying All Instances of Sensitive Data

- Discover where sensitive data is located
  - Social Security Numbers are stored with the following format: `xxx-xx-xxxx`
  - Look for any Table, View, and Synonym with data in this format

- Regular Expressions (regex) are a powerful tool to help find any sensitive data
  - This regex `[0-9]{3}-[0-9]{2}-[0-9]{4}` will find data in this format: `xx-xx-xxxx`

- Once data is found, multiple actions can be initiated for automation and time saving
Integrating with SIEM Systems

Policy violation in DAM system

Events in IBM SIEM

Monitor & Enforce
Protecting Data Enterprise-wide is a Key Element of Information Governance

- Understanding the “what & where” of enterprise data
- Knowing who’s accessing your data when, how and why
- Monitoring and reporting on data access for audit purposes
- Protecting the data across the enterprise, both internal and external threats
What the Analysts are saying

Enterprise Database Auditing and Real-Time Protection

“Database Auditing and Real-Time Protection have become necessities for all … to meet regulatory compliance requirements and mitigate risk against various types of threats to enterprise databases.”

DAM Technology

“Auditors and Security Teams should evaluate DAM technologies to identify applicability in their enterprises to enhance overall risk management and achieve data governance goals.”
“Breaches Go Undiscovered for Weeks or Months in 75% of the Cases” – Can your company take this risk?

[Diagram showing time span of breach events by percent of breaches]
Summary & Conclusions

- Basic database security, traditional log management, network scanners, SIEM and DLP are insufficient to secure high-value databases
  - Ineffective against privileged users or end-users violating corporate policies
  - Native logging/auditing require database changes & impact performance
  - No knowledge about DBMS commands, vulnerabilities & structures
  - No real-time monitoring to immediately detect or block unauthorized access
  - Inability to detect fraud at application layer (SAP, PeopleSoft, etc.)
  - No data discovery, leakage detection, file integrity monitoring, …
  - No data masking to de-identify data in test/dev environments
  - Requires unique policies for each DBMS platform
DAM Provides a Simple Means of Centralizing and Automating Controls

- Discovering and applying controls to all sensitive data
- Controlling who accesses and modifies what data, from where, and when
- Managing exposure to misuse of credentials, privileges, etc.
- Ensuring sensitive data stores are appropriately configured
- Standardizing, automating and streamlining the review and remediation of policy violations, as well compliance validation activities
- Without compromising separation of duties or performance

Implement Defense-in-Depth for your database infrastructure

“…security techniques available within the database can and should be augmented with security systems sitting outside the database…”

- Network security systems
- Database activity monitoring
- Host security
- etc.
IBM Acquired Guardium November 30, 2009

- Joining IBM's Information Management business

- Why Guardium? Unique ability to:
  - Safeguard critical enterprise information
  - Reduce operational costs by automating compliance processes
  - Simplify governance with centralized policies for heterogeneous infrastructures
  - Continuously monitor access and changes to high-value databases

- Trusted information lies at the center of today’s business transformations
  - Guardium enables organizations to maintain trusted information infrastructures
  - Business analytics and trusted information drive smarter business outcomes
  - This supports IBM’s vision of creating a Smarter Planet: Smarter energy, smarter healthcare, smarter cities, smarter finance, smarter IT, and more
How Guardium Fits with IBM’s Security Framework

- Tivoli Identity Manager, Access Manager, zSecure, …
- **Guardium Database Monitoring**, Optim TDM & DP, SIEM, …
- Rational AppScan, Ounce Suite, WebSphere, DataPower, …
- Server Protection, Network Intrusion Prevention System (IPS, …
Guardium Evolution

V3:
- Adds support for SQL Server and Sybase
- Enhances log storage mechanism to intelligent structured storage (patent-pending)
- Adds SQL parsing
- Adds alerting & reporting

V4:
- Enhances network monitoring with host-based agent for local access monitoring (S-TAP, patent-pending)
- Adds behavioral security assessment & access map, real-time policy-based access control
- Database firewall
- SOX & PCI Accelerators

V5:
- Adds support for IBM Informix
- Application user monitoring for Oracle EBS, PeopleSoft, JDE + custom apps
- Encrypted traffic monitoring: Kerberos, SQL Server, SSL, TLS
- Monitoring for bind values, Full SQL, Stored Procedures
- Data Privacy Accelerator

V6:
- Enhances S-TAP with universal local-access monitoring (Bequtech, etc.)
- Authentication for LDAP, RSA SecureID
- Application user monitoring for Siebel & SAP
- Change Audit System (CAS)
- Data discovery & classification
- Data extrusion policies
- Incident Manager
- Unstructured data monitoring
- External data connector
- Direct archiving to IBM TSM & EMC Centera

V7:
- Adds vulnerability & configuration assessment with enhanced behavioral assessment
- S-GATE Granular Access Control (blocking)
- Oracle ASO & SQL Server 2008 encrypted traffic monitoring
- SIEM data feed
- XML & xQuery monitoring
- Archiving of workflow results
- Classification of unstructured data
- Subscription-based content
- Adds DB2 on z/OS
- Adds Teradata support
- Adds MySQL support

Forrester Recognizes Guardium as "A Leader across the board" with the highest overall scores for Architecture, Current Offering & Corporate Strategy

Forms strategic partnerships with Neon Enterprise Software (mainframe support) and NetApp (encryption)

Guardium founded to develop the industry’s first solution for enterprise data security

Named “Hot Pick” by Information Security Magazine

Cisco makes strategic investment

162M records breached in 2007

Named to Red Herring 100
Guardium: Chosen by Leading Organizations Worldwide for Database Protection and Compliance

- 5 of the top 5 global banks
- 2 of the top 3 global retailers
- 3 of the top 5 global insurers
- 2 of the world’s favorite beverage brands
- The most recognized name in PCs
- 25 of the world’s leading telcos

- Top government agencies
- Top 3 auto maker
- #1 dedicated security company
- Leading energy suppliers
- Major health care providers
- Media & entertainment brands
Thought Leadership

- Check out *Implementing Database Security and Auditing*
  - Definitive 413-page text for security, risk management & database professionals
  - Specific tips for DB2, Oracle, SQL Server, MySQL and Sybase
  - Written by database security expert, IBM GOLD Consultant & Guardium CTO, Ron Ben Natan, Ph.D.

- Check out *HOWTO Secure and Audit Oracle 10g and 11g*
  - Definitive 454-page text for security, risk management & db professionals
  - Written by database security expert & Guardium CTO, Ron Ben Natan, Ph.D.

- Check out the *Database Security TechCenter*
  - Latest news, tips & reports

- See "Resources" section for case studies, ROI examples, white papers, reviews, Webcasts

- **Next Webcast:** “HOWTO Secure Your SAP Data and Eliminate Last-Minute Audit Scrambles”, October 28, 11am ET
Previously Recorded Webcasts

HOWTO Secure Mainframe PII Data & Pass Compliance Audits Faster
10 Database Activities You Should Monitor to Prevent Database Attacks, Featuring Gartner
The Hacker's Roadmap: HOWTO Safeguard Against Constantly Evolving Threats
HOWTO Secure Oracle 10g and 11g: Hardening the Database
Databases at Risk – and HOWTO Address Them
Creating a Database Security Plan – Why Basic Database Security is No Longer Sufficient
Data Discovery & Classification for Heterogeneous Database Environments
Product Demo: Top Scenarios for Real-Time Database Security & Monitoring
Protecting Against Database Attacks and Insider Threats: Top 5 Scenarios
Best Practices for Data Privacy & Protection
Enforce Database Change Controls for SOX, PCI and SAS70
Best Practices for Database Security & Compliance with Noel Yuhanna, Forrester
Guardium and ArcSight: Application Monitoring & Fraud Detection
S-GATE: Data-Level Access Control for Heterogeneous DBMS Environments
How Dell Simplified Database Security for SOX, PCI, SAS70
Thank You
IBM Security Portfolio
What Are the Challenges?

- No separation of duties -- DBAs run the process & DBAs (or hackers) can easily modify logs
- Performance impact of native logging on the DBMS
- Limited scope & granularity of log data (often ignore SELECTs)
- Not real-time
- No preventive controls
- Another data store to secure and manage ($$$)
- Inconsistent policies across applications, DBMS platforms, compliance initiatives, business units
- Can’t identify end-user fraud for connection-pooled applications that use generic service accounts (SAP, PeopleSoft, etc.)
- Lack of DBMS expertise on security teams
- Significant labor cost to clean & review data, maintain processes
- Poor audit trail of audit processes
- Manual remediation is error prone and costly
Top Database Control Weaknesses: A Client Perspective

#1: Separation of duties is not maintained when dealing with sensitive data

#2: Account credentials are shared between individuals

#3: Not all databases are known and all instances of sensitive data are not cataloged

#4: Access and changes to all sensitive data that violate policy are not logged and investigated

#5: All privileged accounts are not known and managed to minimize exposure

#6: The privileges of all user and group/roles is not known and managed to minimize exposure

#7: Business users do not regularly review data transaction violation logs and take remedial action

#8: Databases not monitored to ensure patching levels and configuration changes are to standard

#9: Password policies are not monitored and enforced

#10: Production data exposed in development, test or training systems
# Top Database Threats: A Client Perspective

1. Unauthorized actions by insiders with special privileges (e.g. alter production systems)
2. Unauthorized data modifications (e.g. update salary table)
3. Unauthorized non-employee activity (contract DBA download sensitive data)
4. External data exposure (e.g. PCI data breach)
5. Unauthorized database structure change (e.g. drop order entry table)
6. Unauthorized activity by external users (e.g. execute xp_cmdshell)
7. Excessive privilege abuse (e.g. read PII)
8. Unauthorized account manipulation (e.g. grant unauthorized user accounts)
9. Denial of Service Attack: (e.g. bring down database through excessive requests)
10. Unauthorized data modifications in outsourced vendor environment (e.g. alter A/P info)
Top Questions an Auditor Should Ask

#1: How do you monitor and control access and changes to sensitive data?

#2: Does your process provide full separation of duties?

#3 Can you detect all transactions by privileged users?

#4: Can you detect and block all policy violations in real-time?

#5: Can you implement consistent policies across the enterprise?

#6: Can you demonstrate that all policy violations are reviewed and remediated?

#7: Can you demonstrate that you have ID’d and applied controls to all sensitive data?

#8: Can you prove controls to limit exposure to SW and config vulnerabilities are implemented?

#9: Can you monitor end users of enterprise applications to detect fraudulent transactions?

#10: Do you have a secure repository encompassing audit data and oversight activities?