Oracle® Audit Vault

Administrator's Guide

Release 10.2.3.1

E13841-02

March 2009

ORACLE®

Oracle Audit Vault Administrator's Guide, Release 10.2.3.1

E13841-02

Primary Authors: Patricia Huey, Rodney Ward

Contributors: Tammy Bednar, Janet Blowney, Raghavendran Hanumantharau, K. Karun , Donna Keesling, Valarie Moore, Janaki Narasinghanallur, Dongwon Park, Arkady Rabinov, Srividya Tat, Vipul Shah, Prahlada Varadan Thirumalai, Lok Sheung, Andrew Wang

Contents

Preface xv

Audience xv
Documentation Accessibility xv
Related Documents xvi
Conventions xvi

1 Introducing Oracle Audit Vault for Administrators

1.1 How Do Administrators Use Oracle Audit Vault? 1-1
1.2 General Steps for Administering Oracle Audit Vault 1-2
1.2.1 Step 1: Understand the Oracle Audit Vault Architecture 1-2
1.2.2 Step 2: Plan the Oracle Audit Vault Source Database and Collector Configuration . 1-2
1.2.3 Step 3: Configure Collectors to Collect Audit Data 1-2
1.2.4 Step 4: Monitor and Maintain the Audit Record Collection Process 1-2
1.3 Components of Oracle Audit Vault 1-3
1.3.1 Source Databases 1-3
1.3.2 Oracle Audit Vault Server 1-4
1.3.3 Audit Vault Collection Agent and Collectors 1-5
1.3.4 How the Oracle Audit Vault Components Work Together 1-7
1.4 Administrative Tools for Managing Oracle Audit Vault 1-9
1.5 Administrative Roles and Their Assigned Tasks 1-9
1.6 Planning the Source Database and Collector Configuration 1-10
1.6.1 About Planning the Source Database and Collector Configuration 1-10
1.6.2 Planning the Oracle Source Database and Collector Configuration 1-11
1.6.3 Planning the Microsoft SQL Server Source Database and Collector Configuration 1-12
1.6.4 Planning the Sybase ASE Source Database and Collector Configuration 1-13
1.6.5 Planning the IBM DB2 Source Database and Collector Configuration 1-13

2 Registering Source Databases and Collectors

2.1 General Steps for Adding Sources and Deploying Collectors 2-1
2.2 Checking and Setting Environment Variables 2-2
2.2.1 About Checking and Setting Linux and UNIX Environment Variables 2-2
2.2.2 Setting the Audit Vault Server Linux and UNIX Environment Variables 2-2
2.2.3 Setting the Collection Agent Linux and UNIX Environment Variables 2-4
2.2.4 Using the Collection Agent in a Microsoft Windows Environment 2-4
2.2.5 Setting the Oracle Source Database Linux and UNIX Environment Variables 2-4

2.3 Registering Oracle Database Sources and Collectors 2-5
2.3.1 Step 1: If Necessary, Create a Password File 2-5
2.3.2 Step 2: Create a User Account on the Oracle Source Database 2-5
2.3.3 Step 3: Verify That the Source Database Is Compatible with the Collectors 2-7
2.3.4 Step 4: Register the Oracle Source Database with Oracle Audit Vault 2-8
2.3.5 Step 5: Add the Oracle Collectors to Oracle Audit Vault 2-9
2.3.6 Step 6: Enable the Audit Vault Agent to Run the Oracle Database Collectors 2-11
2.4 Registering Microsoft SQL Server Database Sources and Collector 2-12
2.4.1 Step 1: Download the SQL Server 2005 Driver for JDBC 2-12
2.4.2 Step 2: Create a User Account on the Microsoft SQL Server Source Database 2-13
2.4.3 Step 3: Verify That the Source Database Is Compatible with the Collector 2-13
2.4.4 Step 4: Register the SQL Server Source Database with Oracle Audit Vault 2-14
2.4.5 Step 5: Add the MSSQLDB Collector to Oracle Audit Vault 2-15
2.4.6 Step 6: Enable the Audit Vault Agent to Run the MSSQLDB Collector 2-15
2.5 Registering Sybase ASE Database Sources and Collector 2-16
2.5.1 Step 1: Download the jConnect for JDBC Driver 2-16
2.5.2 Step 2: Create a User Account on the Sybase ASE Source Database 2-17
2.5.3 Step 3: Verify That the Source Database Is Compatible with the Collector 2-17
2.5.4 Step 4: Register the Sybase ASE Source Database with Oracle Audit Vault 2-17
2.5.5 Step 5: Add the SYBDB Collector to Oracle Audit Vault 2-18
2.5.6 Step 6: Enable the Audit Vault Agent to Run the SYBDB Collector 2-18
2.6 Registering IBM DB2 Database Sources and Collector 2-19
2.6.1 Step 1: Copy the DB2 Data Server Driver for JDBC and SQLJ to the Audit Vault Homes 2-19
2.6.2 Step 2: Designate a User Account on the IBM DB2 Source Database 2-20
2.6.3 Step 3: Verify That the Source Database Is Compatible with the Collector 2-20
2.6.4 Step 4: Register the IBM DB2 Source Database with Oracle Audit Vault 2-20
2.6.5 Step 5: Add the DB2DB Collector to Oracle Audit Vault 2-21
2.6.6 Step 6: Convert the Binary DB2 Audit File to an ASCII Text File 2-22
2.6.6.1 Step 7A: Complete the Preparation Steps 2-22
2.6.6.2 Step 7B: Run the Conversion Script 2-22
2.7 Starting the Collection Agents 2-24
2.7.1 Starting the Collection Agents from the Audit Vault Console 2-24
2.7.2 Starting the Collection Agents from a Shell 2-24
2.8 Starting the Collectors 2-25
2.8.1 Starting the Collectors from the Audit Vault Console 2-25
2.8.2 Starting the Collectors from the Audit Vault Server or Collection Agent Shell 2-26
2.9 Checking the Status of the Collectors 2-27
2.9.1 Checking the Status of Collectors from the Audit Vault Console 2-27
2.9.2 Checking the Status of Collectors from a Shell 2-27
2.10 Checking If the Collectors Are Collecting Audit Records 2-27

3 Managing Oracle Audit Vault

3.1 About Managing Oracle Audit Vault 3-1
3.2 Managing the Audit Vault Server 3-1
3.2.1 About Managing the Audit Vault Console 3-1
3.2.2 Checking the Audit Vault Console Status 3-2

3.2.3 Starting the Audit Vault Console ... 3-2
3.2.4 Stopping the Audit Vault Server Console ... 3-3
3.2.5 Globally Disabling and Enabling Alert Settings ... 3-3
3.2.6 Viewing Audit Event Categories ... 3-3
3.2.7 Viewing Operational Errors That Oracle Audit Vault Catches ... 3-5
3.3 Altering Collector Properties and Attributes ... 3-6
3.3.1 About Collector Properties and Attributes ... 3-6
3.3.2 Altering Collector Properties and Attributes Using the Audit Vault Console ... 3-6
3.3.3 Altering Collector Properties and Attributes Using a Shell ... 3-6
3.4 Managing the Oracle Audit Vault Data Warehouse ... 3-7
3.4.1 About Managing the Oracle Audit Vault Data Warehouse ... 3-7
3.4.2 Setting the Audit Vault Data Warehouse Refresh Schedule and Retention Period ... 3-8
3.4.2.1 About Setting the Refresh Schedule and Retention Period ... 3-8
3.4.2.2 Scheduling the Audit Data Refresh Settings Using the Audit Vault Console ... 3-9
3.4.2.3 Scheduling the Audit Data Refresh Settings Using a Shell ... 3-10
3.4.3 Manually Refreshing Audit Vault Data Warehouse Audit Data ... 3-11
3.4.3.1 About Manually Refreshing the Data Warehouse Data ... 3-11
3.4.3.2 Manually Refreshing the Data Warehouse Using the Audit Vault Console ... 3-11
3.4.3.3 Manually Refreshing the Data Warehouse Using a Shell ... 3-12
3.4.4 Loading Data to the Oracle Audit Vault Data Warehouse ... 3-12
3.4.4.1 About Loading Data into the Oracle Audit Vault Warehouse ... 3-12
3.4.4.2 Loading Data Warehouse Data Using the Audit Vault Console ... 3-12
3.4.4.3 Loading Data Warehouse Data Using a Shell ... 3-13
3.4.5 Purging Data from the Oracle Audit Vault Data Warehouse ... 3-14
3.4.5.1 About Purging the Oracle Audit Vault Data Warehouse ... 3-14
3.4.5.2 Purging Data Warehouse Data Using the Audit Vault Console ... 3-14
3.4.5.3 Purging Data Warehouse Data Using a Shell ... 3-14
3.5 Altering Source Database Attributes ... 3-15
3.5.1 About Source Database Attributes ... 3-15
3.5.2 Altering Source Database Attributes Using the Audit Vault Console ... 3-15
3.5.3 Altering Source Database Attributes Using a Shell ... 3-16
3.6 Removing Source Databases from Oracle Audit Vault ... 3-17
3.6.1 About Removing Source Databases from Oracle Audit Vault ... 3-17
3.6.2 Removing a Source Database Using the Audit Vault Console ... 3-17
3.6.3 Removing a Source Database Using a Shell ... 3-18

4 Administering the Oracle Audit Vault Repository

4.1 About the Administrative Tasks in This Chapter ... 4-1
4.2 Monitoring the Audit Vault Server SYSAUX Tablespace Space Usage ... 4-1
4.3 Monitoring Audit Vault Server Archive Log Disk Space Usage ... 4-2
4.4 Monitoring the Audit Vault Server Flash Recovery Area ... 4-2
4.5 Managing Oracle Audit Vault Backup and Recovery Operations ... 4-2
4.6 Using a Collection Agent to Listen to Oracle Database RAC Nodes ... 4-3
4.7 Configuring Collection Agent Connectivity for Oracle Database RAC ... 4-4
4.8 Purging the Oracle Source Database Audit Trail Records ... 4-4
4.8.1 General Steps for Purging the Oracle Database Audit Trail ... 4-4
4.8.2 Step 1: Prepare the Oracle Database Audit Trail for Purging ... 4-5

4.8.2.1 Step 1A: Download the DBMS_AUDIT_MGMT Package ... 4-5
4.8.2.2 Step 1B: Move the Database Audit Trail to a Different Tablespace ... 4-6
4.8.3 Step 2: Create a Job to Automatically Purge the Oracle Database Audit Trail ... 4-7
4.8.3.1 Step 2A: Ensure That the Collectors Are Enabled ... 4-7
4.8.3.2 Step 2B: Initialize the Audit Trail Cleanup Operation ... 4-7
4.8.3.3 Step 2C: Create the Purge Job ... 4-8
4.8.4 Step 3: Optionally, Set a Record Batch Size for the Purge Operations ... 4-9
4.8.5 Step 4: Perform Maintenance Tasks as Needed ... 4-10
4.8.5.1 Verifying That the Audit Trail Is Initialized for Cleanup ... 4-10
4.8.5.2 Enabling or Disabling an Audit Trail Purge Job ... 4-10
4.8.5.3 Setting the Default Audit Trail Purge Interval for Any Audit Trail Type ... 4-11
4.8.5.4 Setting the Default Audit Trail Purge Job Interval for a Specified Purge Job ... 4-12
4.8.5.5 Clearing the Database Audit Trail Records Batch Size ... 4-12
4.8.5.6 Canceling the Initialization Cleanup Settings ... 4-12
4.8.5.7 Deleting an Audit Trail Purge Job ... 4-13
4.8.5.8 Configuring Tracing Debug Levels for Purge Operations ... 4-13
4.8.5.9 Setting the Size of the Operating System Audit Trail ... 4-14
4.8.5.10 Setting the Age of the Operating System Audit Trail ... 4-15

5 Managing Oracle Audit Vault Security

5.1 About Managing Oracle Audit Vault Security ... 5-1
5.2 Managing Authentication Metadata Using Oracle Advanced Security ... 5-1
5.3 Using Oracle Database Vault with Oracle Audit Vault ... 5-2
5.4 Changing Oracle Audit Vault User Passwords on a Regular Basis ... 5-4
5.4.1 About Oracle Audit Vault User Passwords ... 5-4
5.4.2 Changing the AV_ADMIN User Password ... 5-5
5.4.3 Changing the AV_AGENT Password ... 5-5
5.4.4 Changing the Source User Password ... 5-6
5.4.5 Changing the AV_AUDITOR Password ... 5-7
5.4.6 Ensuring That All Changed User Name Passwords Work Correctly ... 5-7
5.5 Configuring HTTPS Communication for Oracle Audit Vault ... 5-8
5.5.1 About Configuring HTTPS Communication for Oracle Audit Vault ... 5-8
5.5.2 Step 1: Generate the Certificate Request ... 5-9
5.5.3 Step 2: Configure the Audit Vault Server and Agent HTTPS Communication ... 5-9

6 Audit Vault Configuration Assistant (AVCA) Reference

6.1 add_agent ... 6-2
6.2 create_credential ... 6-2
6.3 create_wallet ... 6-3
6.4 deploy_av ... 6-4
6.5 drop_agent ... 6-5
6.6 generate_csr ... 6-6
6.7 -help ... 6-6
6.8 import_cert ... 6-8
6.9 redeploy ... 6-9
6.10 remove_cert ... 6-9
6.11 secure_agent ... 6-10

6.12 secure_av 6-11
6.13 set_warehouse_retention 6-12
6.14 set_warehouse_schedule 6-13

7 Audit Vault Control (AVCTL) Reference

7.1 -help 7-2
7.2 load_warehouse 7-3
7.3 purge_warehouse 7-4
7.4 refresh_warehouse 7-5
7.5 show_agent_status 7-6
7.6 show_av_status 7-7
7.7 show_collector_status 7-7
7.8 show_oc4j_status 7-8
7.9 start_agent 7-9
7.10 start_av 7-9
7.11 start_collector 7-10
7.12 start_oc4j 7-11
7.13 stop_agent 7-12
7.14 stop_av 7-12
7.15 stop_collector 7-13
7.16 stop_oc4j 7-14

8 Audit Vault Oracle Database (AVORCLDB) Utility Commands

8.1 avorcldb 8-1
8.2 add_collector 8-2
8.3 add_source 8-4
8.4 alter_collector 8-5
8.5 alter_source 8-8
8.6 drop_collector 8-9
8.7 drop_source 8-9
8.8 -help 8-10
8.9 setup 8-11
8.10 verify 8-12

9 Audit Vault Microsoft SQL Server (AVMSSQLDB) Utility Commands

9.1 avmssqldb 9-1
9.2 add_collector 9-2
9.3 add_source 9-3
9.4 alter_collector 9-3
9.5 alter_source 9-5
9.6 drop_collector 9-6
9.7 drop_source 9-6
9.8 -help 9-7
9.9 setup 9-8
9.10 verify 9-9

10 Audit Vault Sybase ASE (AVSYBDB) Utility Commands

10.1 avsybdb 10-1
10.2 add_collector 10-2
10.3 add_source 10-3
10.4 alter_collector 10-3
10.5 alter_source 10-4
10.6 drop_collector 10-5
10.7 drop_source 10-6
10.8 -help 10-7
10.9 setup 10-7
10.10 verify 10-8

11 Audit Vault IBM DB2 (AVDB2DB) Utility Commands

11.1 avdb2db 11-1
11.2 add_collector 11-2
11.3 add_source 11-3
11.4 alter_collector 11-3
11.5 alter_source 11-4
11.6 drop_collector 11-5
11.7 drop_source 11-6
11.8 -help 11-7
11.9 setup 11-7
11.10 verify 11-8

12 REDO Collector Database Reference

12.1 About the Recommended Settings for the REDO Collector 12-1
12.2 Oracle9*i* Database Release 2 (9.2) Audit Source Parameter Recommendations 12-1
12.3 Oracle Database 10*g* Release 1 (10.1) Audit Source Parameter Recommendations 12-5
12.4 Oracle Database 10*g* Release 2 (10.2) Audit Source Parameter Recommendations 12-9
12.5 Oracle Database 11*g* Release 1 (11.1) Audit Source Parameter Recommendations 12-13

13 DBMS_AUDIT_MGMT Data Dictionary Views

13.1 DBA_AUDIT_MGMT_CONFIG_PARAMS 13-1
13.2 DBA_AUDIT_MGMT_LAST_ARCH_TS 13-2
13.3 DBA_AUDIT_MGMT_CLEANUP_JOBS 13-2
13.4 DBA_AUDIT_MGMT_CLEAN_EVENTS 13-3

14 DBMS_AUDIT_MGMT PL/SQL Package

14.1 About Using the DBMS_AUDIT_MGMT PL/SQL Package 14-1
14.2 DBMS_AUDIT_MGMT PL/SQL Package Security Model 14-2
14.3 DBMS_AUDIT_MGMT PL/SQL Package Constants 14-2
14.4 Summary of DBMS_AUDIT_MGMT PL/SQL Package Subprograms 14-3
14.4.1 CLEAN_AUDIT_TRAIL Procedure 14-4
14.4.2 CLEAR_AUDIT_TRAIL_PROPERTY Procedure 14-5
14.4.3 CLEAR_LAST_ARCHIVE_TIMESTAMP Procedure 14-6

14.4.4 CREATE_PURGE_JOB Procedure ... 14-7
14.4.5 DEINIT_CLEANUP Procedure ... 14-8
14.4.6 DROP_PURGE_JOB Procedure ... 14-9
14.4.7 GET_AUDIT_COMMIT_DELAY Function ... 14-9
14.4.8 INIT_CLEANUP Procedure ... 14-10
14.4.9 IS_CLEANUP_INITIALIZED Function ... 14-11
14.4.10 SET_AUDIT_TRAIL_LOCATION Procedure ... 14-11
14.4.11 SET_AUDIT_TRAIL_PROPERTY Procedure ... 14-12
14.4.12 SET_DEBUG_LEVEL Procedure ... 14-14
14.4.13 SET_LAST_ARCHIVE_TIMESTAMP Procedure ... 14-15
14.4.14 SET_PURGE_JOB_INTERVAL Procedure ... 14-16
14.4.15 SET_PURGE_JOB_STATUS Procedure ... 14-17

A Troubleshooting an Oracle Audit Vault System

A.1 Location of Audit Vault Server Log and Error Files ... A-1
A.2 Location of Audit Vault Collection Agent Log and Error Files ... A-2
A.3 Troubleshooting Tips ... A-4
A.3.1 Checking Trace Files for Detailed Information About Oracle Database Errors ... A-4
A.3.2 Troubleshooting Audit Vault Server ... A-5
A.3.3 Troubleshooting Audit Vault Collection Agent ... A-5
A.3.4 Troubleshooting the Audit Vault Collector ... A-7
A.3.5 Troubleshooting Oracle Audit Vault Console ... A-9
A.3.6 Troubleshooting Oracle Audit Vault in an Oracle Real Application Clusters Environment A-10

B Oracle Audit Vault Error Messages

B.1 Audit Vault Server Error Messages ... B-1
B.1.1 Generic Error Codes ... B-1
B.1.2 Source Database and Event Error Codes ... B-2
B.1.3 Collector Error Codes ... B-3
B.1.4 Attribute Definition Error Codes ... B-4
B.1.5 Alert Error Codes ... B-4
B.1.6 Server-Side Audit Service Error Messages ... B-5
B.1.7 Data Warehouse Error Messages ... B-5
B.1.8 Other Audit Vault Policy Error Messages ... B-6
B.2 Oracle Audit Vault Client Error Messages ... B-7
B.2.1 General Error Messages ... B-7
B.2.2 CSDK Error Messages ... B-7
B.2.3 OSAUD Collector Error Messages ... B-8
B.2.4 DBAUD Collector Error Messages ... B-11

Glossary

Index

List of Examples

2–1 Partially Successful Verify Operation of Source Compatibility with the Collectors 2-7
2–2 Successful Verify Operation of Source Compatibility with the REDO Collector..... 2-8
2–3 Adding the OSAUD Collector to Oracle Audit Vault for UNIX Platforms..... 2-10
2–4 Adding the OSAUD Collector to Oracle Audit Vault on Microsoft Windows..... 2-10
2–5 Adding the DBAUD Collector to Oracle Audit Vault..... 2-11
2–6 Adding the REDO Collector to Oracle Audit Vault 2-11

List of Figures

1–1 Overview of the Oracle Audit Vault Components 1-7
1–2 Detailed View of the Oracle Audit Vault Components..... 1-8
3–1 Audit Event Category Management Page..... 3-4
3–2 Audit Errors Page 3-5

List of Tables

1–1	Supported Source Database Products	1-3
1–2	Oracle Audit Vault Server Components	1-4
1–3	Oracle Audit Vault Collection Agent Components	1-5
1–4	Oracle Audit Vault Collector Types and Audit Trails	1-6
1–5	Oracle Audit Vault Administrator Roles and Their Assigned Tasks	1-10
1–6	Oracle Database Operating System Audit Settings for the OSAUD Collector	1-11
1–7	Oracle Database Audit Trail Settings for the DBAUD Collector	1-12
1–8	Oracle Database Redo Log Setting for the REDO Collector	1-12
1–9	Microsoft SQL Server Source Database Audit Settings for the MSSQLDB Collector	1-13
1–10	Sybase ASE Database Audit Setting for the SYBDB Collector	1-13
1–11	IBM DB2 Database Audit Setting for the DB2DB Collector	1-14
2–1	Audit Vault Server Environment Variable Settings	2-2
5–1	Roles and Privileges Granted to Audit Vault or Database Vault Administrators	5-3
5–2	Database Core Accounts Created and Privileges Use	5-3
5–3	Storage Location of Audit Vault and Source User Name Passwords	5-4
6–1	Audit Vault Configuration Assistant Commands	6-1
7–1	Audit Vault Control Commands	7-1
8–1	AVORCLDB Commands	8-1
8–2	DBAUD Collector Attributes	8-6
8–3	OSAUD Collector Attributes	8-7
8–4	REDO Collector Attributes	8-7
8–5	Source Attributes	8-8
9–1	AVMSSQLDB Commands	9-1
9–2	MSSQLDB Collector Attributes	9-4
9–3	Source Attributes	9-5
10–1	AVSYBDB Commands	10-1
10–2	SYBDB Collector Attributes	10-4
10–3	Source Attributes	10-5
11–1	AVDB2DB Commands	11-1
11–2	DB2DB Collector Attributes	11-4
11–3	Source Attributes	11-5
12–1	Hidden Initialization Parameters to Be Configured for the Database Source	12-1
12–2	Initialization Parameters to Be Configured for the Database Source	12-2
12–3	ARCHIVE_LAG_TARGET Recommended Setting	12-4
12–4	Hidden Initialization Parameters to Be Configured for the Database Source	12-5
12–5	Initialization Parameters to Be Configured for the Database Source	12-5
12–6	Hidden Initialization Parameters to Be Configured for the Database Source	12-9
12–7	Initialization Parameters to Be Configured for the Database Source	12-9
12–8	Hidden Initialization Parameters to Be Configured for the Database Source	12-13
12–9	Initialization Parameters to Be Configured for the Database Source	12-14
13–1	DBMS_AUDIT_MGMT Data Dictionary Views	13-1
14–1	DBMS_AUDIT_MGMT Constants - Types of Audit Trails	14-2
14–2	DBMS_AUDIT_MGMT Constants - Audit Trail Properties	14-3
14–3	DBMS_AUDIT_MGMT Constants - Purge Job Status	14-3
14–4	DBMS_AUDIT_MGMT Constants - Trace Level Values	14-3
14–5	DBMS_AUDIT_MGMT Package Subprograms	14-3
A–1	Name and Description of Audit Vault Server Log and Error Files	A-1
A–2	Name and Description of Audit Vault Collection Agent Log and Error Files	A-3

Preface

Oracle Audit Vault Administrator's Guide explains how Oracle Audit Vault administrators can perform administrative tasks on an Oracle Audit Vault system. This guide assumes that you have completed the installation tasks covered in *Oracle Audit Vault Server Installation Guide* and *Oracle Audit Vault Collection Agent Installation Guide*. This guide accompanies Beta Patch Release 10.2.3.0.1.

This preface contains:

- Audience
- Documentation Accessibility
- Related Documents
- Conventions

Audience

This document is intended for anyone who is responsible for administering an Oracle Audit Vault system.

Documentation Accessibility

Our goal is to make Oracle products, services, and supporting documentation accessible to all users, including users that are disabled. To that end, our documentation includes features that make information available to users of assistive technology. This documentation is available in HTML format, and contains markup to facilitate access by the disabled community. Accessibility standards will continue to evolve over time, and Oracle is actively engaged with other market-leading technology vendors to address technical obstacles so that our documentation can be accessible to all of our customers. For more information, visit the Oracle Accessibility Program Web site at `http://www.oracle.com/accessibility/`.

Accessibility of Code Examples in Documentation

Screen readers may not always correctly read the code examples in this document. The conventions for writing code require that closing braces should appear on an otherwise empty line; however, some screen readers may not always read a line of text that consists solely of a bracket or brace.

Accessibility of Links to External Web Sites in Documentation

This documentation may contain links to Web sites of other companies or organizations that Oracle does not own or control. Oracle neither evaluates nor makes any representations regarding the accessibility of these Web sites.

Deaf/Hard of Hearing Access to Oracle Support Services

To reach Oracle Support Services, use a telecommunications relay service (TRS) to call Oracle Support at 1.800.223.1711. An Oracle Support Services engineer will handle technical issues and provide customer support according to the Oracle service request process. Information about TRS is available at http://www.fcc.gov/cgb/consumerfacts/trs.html, and a list of phone numbers is available at http://www.fcc.gov/cgb/dro/trsphonebk.html.

Related Documents

For more information, see the following documents. See also the platform-specific Oracle Audit Vault Server installation guides.

- *Oracle Audit Vault Server Installation Guide for Linux x86*
- *Oracle Audit Vault Collection Agent Installation Guide*
- *Oracle Audit Vault Licensing Information*
- *Oracle Audit Vault Auditor's Guide*
- *Oracle Database Vault Administrator's Guide*
- *Oracle Database Security Guide*
- *Oracle Database Advanced Security Administrator's Guide*
- *Oracle Data Guard Concepts and Administration*
- *Oracle Database Administrator's Guide*
- *Oracle Database Concepts*

To download free release notes, installation documentation, updated versions of this guide, white papers, or other collateral, visit the Oracle Technology Network (OTN). You must register online before using OTN. Registration is free. You can register at

http://www.oracle.com/technology/membership/

If you already have a user name and password for OTN, then you can go directly to the documentation section of the OTN Web site at

http://www.oracle.com/technology/documentation/

For OTN information specific to Oracle Audit Vault, visit

http://www.oracle.com/technology/products/audit-vault/index.html

For the Oracle Audit Vault Discussion Forums, visit

http://forums.oracle.com/forums/forum.jspa?forumID=391

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
italic	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
`monospace`	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

1

Introducing Oracle Audit Vault for Administrators

This chapter contains:

- How Do Administrators Use Oracle Audit Vault?
- General Steps for Administering Oracle Audit Vault
- Components of Oracle Audit Vault
- Administrative Tools for Managing Oracle Audit Vault
- Administrative Roles and Their Assigned Tasks
- Planning the Source Database and Collector Configuration

1.1 How Do Administrators Use Oracle Audit Vault?

By the time you begin to use this guide, you will have installed Oracle Audit Vault, and the databases (called **source database**s, or **audit data source**s) from which you want to extract audit data are ready to audit. This guide explains how to configure the source databases so that Oracle Audit Vault can collect their audit data. After you have completed this configuration, then auditors can generate and customize reports that describe this audit data.

An Oracle Audit Vault administrator is responsible for the following tasks:

- Ensuring that the source databases have auditing enabled
- Understanding the type of auditing that each source database uses
- Selecting the correct Oracle Audit Vault component, called a **collector**, to connect to the source database, based on the type of auditing that database uses
- Configuring this collector to connect Oracle Audit Vault to the source database
- Configuring and scheduling Audit Vault Server processes
- Ensuring that the collectors are collecting audit data from the source database
- Managing the day-to-day activities of Oracle Audit Vault, such as disk space and backup and recovery operations
- Managing security for Oracle Audit Vault
- Monitoring Oracle Audit Vault to ensure that it is consistently collecting audit data

Oracle Database administrators are responsible for running the Oracle Database audit trail cleanup procedures on the source database, which purge audit trail records from the Oracle source database after these records are archived.

1.2 General Steps for Administering Oracle Audit Vault

To administer Oracle Audit Vault, follow these steps:

- Step 1: Understand the Oracle Audit Vault Architecture
- Step 2: Plan the Oracle Audit Vault Source Database and Collector Configuration
- Step 3: Configure Collectors to Collect Audit Data
- Step 4: Monitor and Maintain the Audit Record Collection Process

1.2.1 Step 1: Understand the Oracle Audit Vault Architecture

In this chapter, Section 1.3 describes the main components of Oracle Audit Vault, and explains how these components work together. Section 1.4 describes the various tools that you use to administer Oracle Audit Vault. Section 1.5 describes the predefined roles that are created during the Oracle Audit Vault installation process. Understanding how these pieces fit together provides the foundation you need to administer Oracle Audit Vault.

1.2.2 Step 2: Plan the Oracle Audit Vault Source Database and Collector Configuration

Section 1.6 provides guidelines for selecting the correct Oracle Audit Vault collector (that is, the module that collects audit data from your source databases) based on the type of database from which you are collecting audit data. You must understand the audit settings and audit trail used in your source databases before you can select the correct collector.

1.2.3 Step 3: Configure Collectors to Collect Audit Data

After you have decided which collectors to use for your source database, you are ready to configure them. Chapter 2 explains how to register (configure) collectors for the source databases.

To accomplish the configuration, you can use the command-line utilities described in Section 1.4.

After you complete this step, Oracle Audit Vault is collecting audit data, which the auditors on your site can access by using the reporting tools described in *Oracle Audit Vault Auditor's Guide*.

1.2.4 Step 4: Monitor and Maintain the Audit Record Collection Process

After you have completed the configuration, you should monitor the audit collection activities to ensure that they are working properly. These tasks include the following:

- **Perform common management tasks.** For example, you may need to check whether the collectors are running, fine-tune how data is collected in the Oracle Audit Vault data warehouse, or modify the attributes of a source database. See the following chapters:
  - Chapter 3 describes common management tasks.
  - Chapter 4 provides advice on managing an Oracle Audit Vault installation on an Oracle Real Application Clusters environment, and what to do if you are concerned that your audit data will fill the default tablespace and disk space.
  - Chapter 5 describes common security tasks and how Oracle Advanced Security and Oracle Database Vault enhance the security of an Oracle Audit Vault system.

  - Chapter 12 describes optimum initialization parameter settings for the REDO collector.

- **For Oracle Database administrators, periodically archive and purge the Oracle Database audit trail for the Oracle source database.** See the following:
  - Section 4.8 describes steps to follow for archiving and purging the Oracle Database audit trail.
  - Chapter 13 describes data dictionary views that you can query to ensure that your configuration settings are correct.
  - Chapter 14 describes the `DBMS_AUDIT_MGMT` package, which contains the calls you use to archive and purge the Oracle Database audit trail.
- **Troubleshoot problems that arise.** See the following:
  - Appendix A describes how to troubleshoot the Oracle Audit Vault system.
  - Appendix B explains how to resolve Oracle Audit Vault-specific error messages.

1.3 Components of Oracle Audit Vault

This section contains:

- Source Databases
- Oracle Audit Vault Server
- Audit Vault Collection Agent and Collectors
- How the Oracle Audit Vault Components Work Together

1.3.1 Source Databases

A source database is a database from which Oracle Audit Vault collects audit data. Oracle Audit Vault can collect this audit data from the internal audit trail tables and operating system audit trail files of a source database.

Table 1–1 lists the supported source database products.

Table 1–1 Supported Source Database Products

Database Product	Supported Versions
Oracle Database	Releases 9.2.*x*, 10.1.*x*, 10.2.*x*, and 11.*x* for the OSAUD and DBAUD collector types Enterprise Edition Releases 9.2.0.8, 10.2.0.3, 10.2.0.4, 11.1.0.6, and 11.1.0.7 and later for the REDO collector type
Microsoft SQL Server	SQL Server 2000 and SQL Server 2005 on Windows 2000 Server and Windows 2003 Server (32-bit) platforms
Sybase Adaptive Server Enterprise (ASE)	ASE 12.5.4 and ASE 15.0.2 on platforms based on Linux and UNIX, and on Microsoft Windows platforms
IBM DB2	IBM DB2 Version 8.2 and Version 9.5 on platforms based on Linux and UNIX, and on Microsoft Windows platforms. If you are using Version 8.2, ensure that you have installed Fixpack 16.

1.3.2 Oracle Audit Vault Server

The Oracle Audit Vault Server contains the tools necessary to configure Oracle Audit Vault to collect audit data from your source databases. The Audit Vault Server also stores in an Oracle database, and makes it available to reporting tools through a data warehouse.

The Audit Vault Server consists of:

- Audit Data Store
- Oracle Audit Vault Console
- The following services:
  - Audit data collection and storage management
  - Alert management
  - Collector management and monitoring
  - Report management
  - Audit settings management to establish your policy management
  - Published data warehouse that can be used with reporting tools such as Oracle Business Intelligence Publisher to create customized reports

  Configuration services help define information about the source databases that connect to Oracle Audit Vault. Oracle Audit Vault stores information (metadata) about the sources of audit data and policy information (database audit settings).

Table 1–2 describes the Oracle Audit Vault Server components. See also Figure 1–2 on page 1-8 to understand how these components work together.

Table 1–2 Oracle Audit Vault Server Components

Components	Description
Oracle Container for Java (OC4J)	Oracle Database container for Web applications. It hosts the following components: ■ **Audit Vault Console.** User interface for administrators to administer Oracle Audit Vault. Oracle Audit Vault auditors also can use this interface to generate reports, create alerts, and create Oracle Database audit policies. ■ **Oracle Enterprise Manager Database Control console.** User interface to manage the raw audit data store or audit repository database ■ **Management Framework.** Internal tool that sends management commands to the Audit Vault collection agent to start or stop collection agents and collectors, collect metrics, receive management commands from the Oracle Audit Vault command-line tools using HTTP protocol or HTTPS mutual certificate-based authentication. Section 1.4 lists the Oracle Audit Vault command-line tools. ■ **Audit Policy System.** Internal service that retrieves and provisions audit settings on the Oracle Database source. It also enables users to create and manage alerts raised by audit events from all source databases as they are stored in the audit event repository.
Database Client	Infrastructure to communicate to the audit repository, consisting of: ■ **Oracle Wallet.** Contains credentials to authenticate Oracle Audit Vault users ■ **Configuration files.** Files used by Oracle Audit Vault for networking, preferences, and so on.

Table 1–2 (Cont.) Oracle Audit Vault Server Components

Components	Description
Configuration and Management Tools	Utilities used to configure and manage Oracle Audit Vault, which are described in detail in Section 1.4. They let you define and configure information about what source databases are known to Oracle Audit Vault.
Logs	Informational and error messages for Oracle Audit Vault. See Section A.1 for more information.
Audit repository	Oracle database to consolidate and manage audit trail records, consisting of: ■ **Raw audit data store.** A partitioned table where audit records are inserted as rows ■ **Warehouse schema.** Open schema of normalized audit trail records. This is a published **data warehouse** that auditors can use with reporting tools such as Oracle Business Intelligence Publisher to create customized reports. ■ **Job scheduler.** Database jobs used to populate and manage the warehouse ■ **Alerts.** Queue that maintains auditor-created alerts

1.3.3 Audit Vault Collection Agent and Collectors

A **collector** retrieves the audit trail data from a source database and sends it to the Audit Vault Server. The **collection agent** manages the collectors. The collectors send both valid and invalid audit records, get configuration information, and send error records using Oracle Call Interface (OCI) and JDBC password-based authentication. If the collection agent is stopped, then the source database will still create an audit trail (assuming auditing is enabled). The next time you restart the collection agent, Oracle Audit Vault retrieves the audit data that had been accumulating since the agent was stopped.

Table 1–3 lists the components of the collection agent. To understand how the collection agent fits in with the Oracle Audit Vault process flow, see Figure 1–2 on page 1-8.

Table 1–3 Oracle Audit Vault Collection Agent Components

Component	Description
OC4J	Oracle container for Web applications. It hosts the following components: ■ **Audit Vault Collector Manager.** Receives management commands from the Audit Vault Server to start and stop collectors, collect and return metrics, and so on ■ **Audit Settings Manager.** Receives commands from Oracle Audit Vault to extract audit settings from an Oracle Database source
Database Server	Infrastructure to communicate to the audit repository, consisting of: ■ **Oracle Wallet.** Contains credentials to authenticate Audit Vault users ■ **Configuration Files.** Files used by Oracle Audit Vault for networking, preferences, and so on
Configuration and Management Tools	Utilities used to configure and manage Oracle Audit Vault. These are the `AVCA`, `AVCTL`, `AVORCLDB`, `AVMSSQLDB`, `AVSYBDB`, and `AVDB2DB` command-line utilities.
Logs	Informational and error messages for Oracle Audit Vault (see Section A.1)

Table 1–3 (Cont.) Oracle Audit Vault Collection Agent Components

Component	Description
Collectors	Table 1–4 shows the type of collectors deployed by the Oracle Audit Vault collection agents and the audit trail from which audit records are extracted and collected.

Table 1–4 lists the types of collectors.

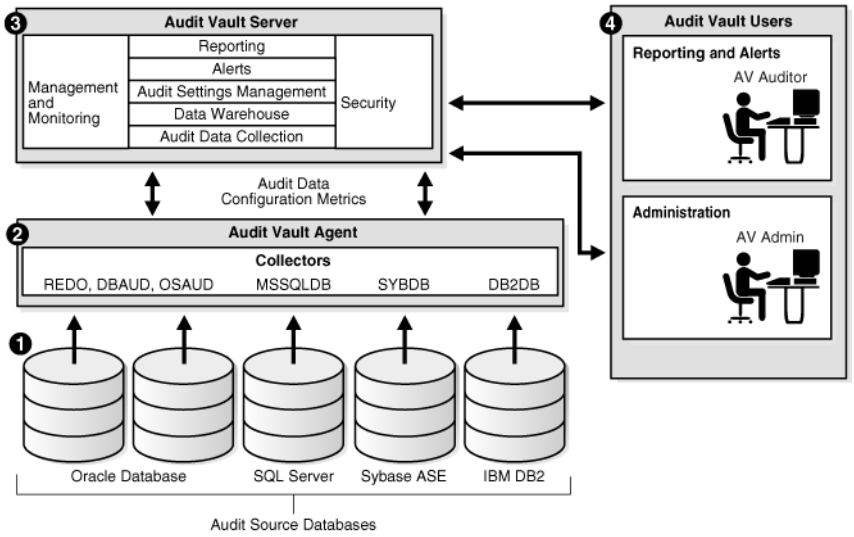
Table 1–4 Oracle Audit Vault Collector Types and Audit Trails

Audit Source	Collector Type	Audit Trail
Oracle Database	DBAUD	Collects from the following audit trails: ■ Oracle Database audit trail, where standard audit events are written to the `SYS.AUD$` dictionary table ■ Oracle Database fine-grained audit trail, where audit events are written to the `SYS.FGA_LOG$` dictionary table ■ Oracle Database Vault audit trail, where audit events are written to the `DVSYS.AUDIT_TRAIL$` dictionary table
Oracle Database	OSAUD	Collects from the following audit trails: ■ **On Linux and UNIX platforms:** The operating system logs (audit logs) (`SYS.AUD$`) (`.aud`) and XML (`.xml`) files), or syslog ■ **On Windows platforms:** The operating system Windows event log and operating system logs (audit logs) XML (`.xml`) files
Oracle Database	REDO	Collects from logical change records (LCRs) from the REDO logs. If you plan to use the REDO collector, you can define the data to audit by creating capture rules for the tables from which the REDO collector will capture audit information. See *Oracle Audit Vault Auditor's Guide* for more information.
Microsoft SQL Server	MSSQLDB	Collects from C2 audit logs, server-side trace logs, and Windows event logs
Sybase ASE	SYBDB	Collects from system audit tables (`sysaudits_01` through `sysaudits_08`) in the `sybsecurity` database
IBM DB2	DB2DB	Collects from ASCII text files extracted from the binary audit log (`db2audit.log`). These files are located in the `security` subdirectory of the DB2 database instance.

1.3.4 How the Oracle Audit Vault Components Work Together

Figure 1–1 provides a high-level overview of how the Oracle Audit Vault components work together.

Figure 1–1 Overview of the Oracle Audit Vault Components



The process flow works as follows:

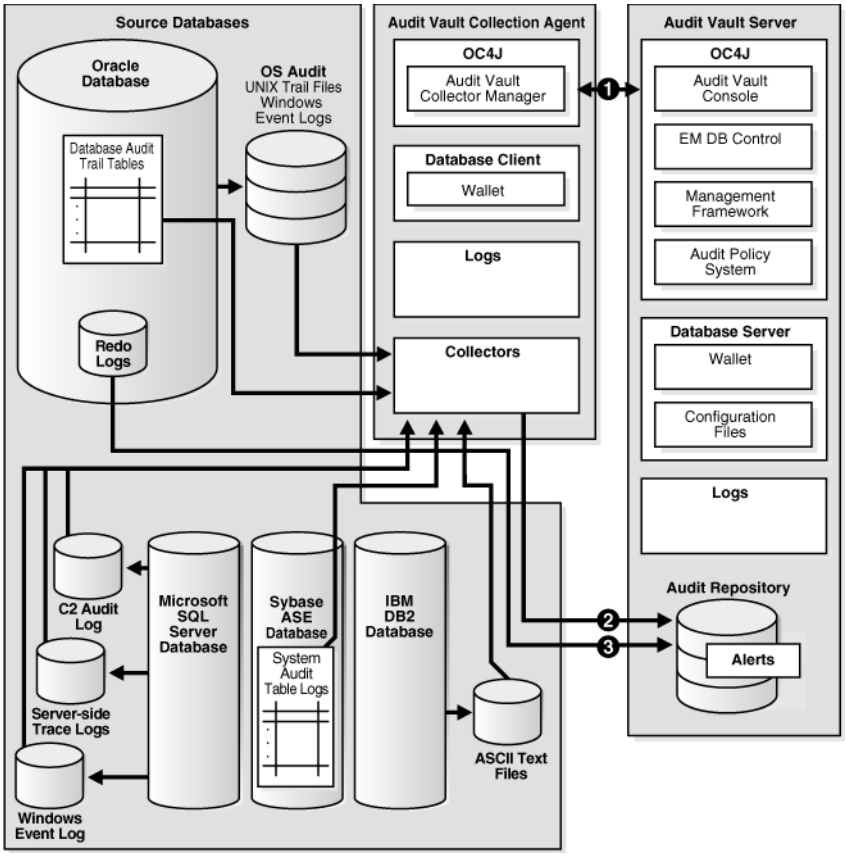
1. The source databases, Oracle Database, SQL Server, Sybase ASE, and IBM DB2, have all been configured to use their respective collectors:
   - Oracle Database uses the REDO, DBAUD, and OSAUD collectors.
   - SQL Server uses the MSSQLDB collector.
   - Sybase ASE uses the SYBDB collector.
   - IBM DB2 uses the DB2DB collector.

   As Figure 1–1 shows, you can configure multiple databases from different database product families using the same Audit Vault collection agent to connect to the same Audit Vault Server.
2. The collectors listed in Step 1 retrieve the audit data from their source databases and send this data to the Audit Vault Server.
3. The Audit Vault Server collects and stores this data it in the database, and then makes it available in the warehouse.

   The data warehouse organizes this data into a set of internal dimension tables. The Audit Vault Server stores other information as well, for both the auditor and the administrator.
4. Once the audit data is in the data warehouse dimension tables, an auditor can retrieve this data to generate and customize reports. Any settings that you, the administrator, create, such as security settings, are contained in this server. The Audit Vault Server stores all the tools that you need to configure the Audit Vault components and source databases.

Figure 1–2 shows a detailed view of the Oracle Audit Vault architecture.

Figure 1–2 Detailed View of the Oracle Audit Vault Components



The process flow works as follows:

1. The OC4J components in the Audit Vault Server and Audit Vault collection agent connect using HTTP or HTTPS.

   The OC4J is a container for Web applications that consists of the Audit Vault Console, the Oracle Enterprise Manager Database Control console, the Audit Vault internal tools (management framework), and the audit policy system used to retrieve and make available the audit settings. The HTTP (or HTTPS) connection is used for starting and stopping agents, managing metrics, and running commands related to policy retrieval.

   The Audit Vault Server contains its own database server, and an Oracle wallet containing the administrator's credentials. It also stores configuration information from utility settings (such as `AVCA`, `AVCTL`, and the command-line utilities used for the four database products) and log files that store operational information, such as broken database connections and missing files.

   In addition to its HTTP or HTTPS connection, each collector in the Oracle Audit Vault collection agent maintains an OCI and a JDBC connection to the Audit Vault Server using the credentials from the client wallet.

2. The collectors retrieve audit records from the source databases and send this data to the audit repository, which contains the Audit Vault data warehouse.

   The data warehouse organizes this data into a set of dimension tables. *Oracle Audit Vault Auditor's Guide* describes the data warehouse dimension tables in detail. In addition to the data warehouse, the audit repository contains auditor-created alert information.

3. Oracle Audit Vault receives data from the Oracle Database redo logs using a database link. The Oracle Database redo logs bypass the collectors.

1.4 Administrative Tools for Managing Oracle Audit Vault

You can use the following tools to administer Oracle Audit Vault:

- **Audit Vault Console.** This graphical user interface provides most of the functionality that you need to administer Oracle Audit Vault.
- **Audit Vault Configuration Assistant (AVCA) command-line utility.** Use `AVCA` to perform operations such as adding, deploying, and dropping agents, or managing wallets. See Chapter 6 for more information.
- **Audit Vault Control (AVCTL) command-line utility.** Use `AVCTL` to load, refresh, start, and stop Oracle Audit Vault collection agents and collectors. You also can load and purge data in the Oracle Audit Vault data warehouse with this utility. See Chapter 7 for more information.
- **Audit Vault Oracle Database (AVORCLDB) command-line utility.** Use `AVORCLDB` to configure Oracle Database source databases with Oracle Audit Vault. See Chapter 8 for more information.
- **Microsoft SQL Server Database (AVMSSQLDB) command-line utility.** Use `AVMSSQLDB` to configure SQL Server source databases with Oracle Audit Vault. See Chapter 9 for more information.
- **Sybase ASE Database (AVSYBDB) command-line utility.** Use `AVSYBDB` to configure Sybase ASE source databases with Oracle Audit Vault. See Chapter 10 for more information.
- **IBM DB2 Database (AVDB2DB) command-line utility.** Use `AVDB2DB` to configure IBM DB2 source databases with Oracle Audit Vault. See Chapter 11 for more information.

1.5 Administrative Roles and Their Assigned Tasks

A default Oracle Audit Vault installation provides a set of administrative roles that you can use to manage Oracle Audit Vault. These roles provide separation-of-duty tasks.

Table 1–5 describes the various Oracle Audit Vault administrator roles and the tasks permitted for each role. See also Table 5–1 on page 5-3 for a listing of the roles and privileges that are granted to these administrator roles.

Table 1–5 Oracle Audit Vault Administrator Roles and Their Assigned Tasks

Role	When Is the Role Granted?	Role Is Granted to Whom?	Description
AV_ADMIN	During Server installation	Audit Vault administrator	Accesses Oracle Audit Vault services to administer, configure, and manage a running Oracle Audit Vault system. A user who is granted this role configures and manages metadata for audit source databases, collection agents, collectors, the configuration of the source database with the collection agent, and the data warehouse. The installation process creates and grants a user account with this role. Only the user granted the AV_ADMIN role can grant the AV_ADMIN role to other Oracle Audit Vault administrators. You can consider the AV_ADMIN role a super-user account for Oracle Audit Vault, except that a user who has been granted this role cannot view, update, or delete audit data.
AV_AUDITOR	During Server installation	Audit Vault auditor	Accesses Oracle Audit Vault reporting and analysis services to monitor components, detect security risks, create and evaluate alert scenarios, create detail and summary reports of events across systems, and manage the reports. A user who is granted this role manages central audit settings and alerts. This user also uses the data warehouse services to further analyze the audit data to look for trends, intrusions, anomalies, and other items of interest. The installation process creates and grants a user account with this role. However, during installation, you optionally can bypass creating this user account. In that case, the roles and privileges normally granted to AV_AUDITOR are granted to AV_ADMIN instead. Typically, one user is granted an AV_ADMIN role and one user is optionally granted an AV_AUDITOR role as part of installing the Audit Vault Server.
AV_AGENT	During collection agent registration	Collection agent software component	Manages collection agents and collectors by starting and stopping them. Oracle Audit Vault creates this role for internal use only.
DV_ACCTMGR	During Audit Vault Server installation	Database Vault account manager	Manages database user accounts. Be aware that the inclusion of Oracle Database Vault in the Audit Vault Server prevents users SYS and SYSTEM from creating, altering, or dropping user accounts. See *Oracle Database Vault Administrator's Guide* for more information about how Oracle Database Vault affects user privileges. See also Section 5.3.
DV_OWNER	During Audit Vault Server installation	Database Vault owner	Manages Oracle Database Vault roles and configuration.

1.6 Planning the Source Database and Collector Configuration

This section contains:

- About Planning the Source Database and Collector Configuration
- Planning the Oracle Source Database and Collector Configuration
- Planning the Microsoft SQL Server Source Database and Collector Configuration
- Planning the Sybase ASE Source Database and Collector Configuration
- Planning the IBM DB2 Source Database and Collector Configuration

1.6.1 About Planning the Source Database and Collector Configuration

This section provides guidelines for selecting the correct Oracle Audit Vault collector for the source databases from which you want to extract audit data. In brief, for Oracle Database, the type of collector that you select depends on the type of auditing that you have enabled in the source database. The Microsoft SQL Server, Sybase ASE, and IBM DB2 databases each use one collector specific to each of these database products.

After you understand which collector to choose, you are ready to register the source database and collector with Oracle Audit Vault.

1.6.2 Planning the Oracle Source Database and Collector Configuration

To plan the Oracle Database source database and collector configuration:

1. Ensure that auditing has been enabled, and find the type of auditing that the Oracle source database uses.

   See *Oracle Audit Vault Auditor's Guide* for more information about the Oracle Database requirements.

2. Based on the audit trail setting, determine which collector to use.

   The type of auditing that has been enabled determines the collector you will choose. The types of collectors available are as follows:

   - **OSAUD collector.** Use this collector if the audit trail is being written to operating system files. Table 1–6 on page 1-11 lists the operating system audit trail settings that use the OSAUD collector.
   - **DBAUD collector.** Use this collector if the audit trail is being written to the database audit trail. Table 1–7 on page 1-12 lists of the database audit trail settings that use the DBAUD collector.
   - **REDO collector.** Use this collector if the database is collecting audit data from the redo logs. Table 1–8 on page 1-12 shows more information about redo logs.

3. Register the Oracle source database and the appropriate collector with Oracle Audit Vault, as described in Section 2.3.

The OSAUD operating system audit settings capture the following activities:

- `SELECT` statements
- Data definition language (DDL) and data manipulation language (DML) statements
- Succeeded and failed actions
- `SYS` operations (Set the `AUDIT_SYS_OPERATIONS` initialization parameter to `TRUE` to perform administrator auditing. SYS auditing collects SQL text information.)

Table 1–6 lists the Oracle Database operating system audit settings that use the OSAUD collector.

Table 1–6 Oracle Database Operating System Audit Settings for the OSAUD Collector

Audit Trail	Audit Trail Settings	Comments
Linux and UNIX-based platforms (`.aud`)	`OS`	None
Linux and UNIX-based platforms (`.xml`)	`XML, EXTENDED`	`EXTENDED` writes SQL text and SQL bind information to the audit trail.
Linux and UNIX-based platforms (syslog)	`OS`	More secure than audit records stored in operating system audit trail
Windows platform Windows event log	`OS`	None

Table 1–6 (Cont.) Oracle Database Operating System Audit Settings for the OSAUD

Audit Trail	Audit Trail Settings	Comments
Windows platform Operating system XML files (`.xml`)	`XML, EXTENDED`	`EXTENDED` writes SQL text and SQL bind information to the audit trail.

Table 1–7 lists the Oracle Database database audit trail settings, which must use the DBAUD collector.

Table 1–7 Oracle Database Audit Trail Settings for the DBAUD Collector

Audit Trail	Audit Trail Setting	Audited Operations	Comments
`SYS.AUD$`	`DB` or `DB, EXTENDED`	`SELECT`, DML, DDL, success and failure, SQL text, SQL bind	`EXTENDED` writes SQL text and SQL bind data to the audit trail.
`SYS.FGA_LOG$`	Not applicable. To enable fine-grained auditing, use the `DBMS_FGA` PL/SQL package.	Very specific user-defined audited conditions, such as the time a user modified a table column	None
`DVSYS.AUDIT_TRAIL$`	Not applicable	Oracle Database Vault audit activity specified by audit options on realms, command rules, and so on	None

Table 1–8 shows the redo log audit trail setting that uses the REDO collector.

Table 1–8 Oracle Database Redo Log Setting for the REDO Collector

Audit Trail	Audit Trail Setting	Audited Operations	Comments
Redo logs	Audit policy: capture rule	DML, DDL, before and after values	Tracks before and after changes to sensitive data columns.

1.6.3 Planning the Microsoft SQL Server Source Database and Collector Configuration

To plan the Microsoft SQL Server source database configuration:

1. Ensure that auditing has been enabled in the SQL Server source database.

   See the Microsoft SQL Server product documentation for more information.

2. Understand the audit trail settings used for SQL Server databases.

   Table 1–9 lists the SQL Server audit trail settings.

3. Configure the MDDSQLDB collector to collect audit data from the SQL Server database, as described in Section 2.4.

Table 1–9 describes the SQL Server audit trail.

Table 1–9 Microsoft SQL Server Source Database Audit Settings for the MSSQLDB Collector

Audit Trail - Audit Logs	Audit Trail Settings	Audited Operations	Comments
C2 audit logs	Configure SQL Server security properties through SQL Server Enterprise Manager.	Auditing compliant with C2 certification Records both failed and successful attempts to access statements and objects Uses all or nothing approach to auditing	Records all actions
Server-side trace logs	Run stored procedures to start and stop tracing, to configure and filter traces.	Records fine-grained security-related activity Can choose exactly which events to audit and what information about each event to record Trace configuration information is not persistent. It is deleted when you restart SQL Server.	Records specific activity Traces can be configured to record only specific activity. Results can be filtered to record only activity that matches a certain pattern, such as a SQL verb (for example, `SELECT`, `INSERT`, `UPDATE`, `DELETE`), or that involve a particular object (for example, a specific table).
Windows event log	Running by default.	Provides a standard, centralized way for applications (and the operating system) to record important software and hardware events.	None

1.6.4 Planning the Sybase ASE Source Database and Collector Configuration

To plan the Sybase ASE source database configuration:

1. Ensure that auditing has been enabled in the Sybase ASE source database.

   See the Sybase ASE product documentation for more information.

2. Understand the audit trail setting information used for Sybase ASE databases.

   Table 1–10 shows the Sybase ASE audit trail setting information.

3. Configure the SYBDB collector to collect audit data from the SQL Server database, as described in Section 2.5.

Table 1–10 describes the Sybase ASE audit trail.

Table 1–10 Sybase ASE Database Audit Setting for the SYBDB Collector

Audit Trail - Audit Logs	Audit Trail Setting	Audited Operation	Comments
System audit table logs	Run system procedures to set global audit options, and then to enable, disable, or restart auditing.	Records standard to fine-grained audit and security-related activity Can choose exactly what to audit Can choose to audit everything or just very specific events	Implement your best practices for Sybase ASE database auditing

1.6.5 Planning the IBM DB2 Source Database and Collector Configuration

To plan the IBM DB2 source database configuration:

1. Ensure that auditing has been enabled in the IBM DB2 source database.

   See the IBM DB2 product documentation for more information.

2. Understand the audit trail information used for IBM DB2 databases.

   Table 1–11 shows the IBM DB2 audit trail setting information.

3. Configure the DB2DB collector to collect audit data from the DB2 database, as described in Section 2.6.

Table 1–11 describes the IBM DB2 audit trail.

Table 1–11 IBM DB2 Database Audit Setting for the DB2DB Collector

Audit Trail - Audit Logs	Audit Trail Setting	Audited Operation	Comments
ASCII text files	Run the `DB2AUDIT` command to enable auditing, disable auditing, and set auditing operations.	**Audit (AUDIT).** Changes to audit records or when the audit log is accessed **Authorization Checking (CHECKING).** Authorization checking during attempts to access or manipulate DB2 database objects or functions **Security Maintenance (SECMAINT).** Grants or revokes to object or database privileges or to the `DBADM` privilege; also modification of the `SYSADM_GROUP`, `SYSCTRL_GROUP`, or `SYSMAINT_GROUP` configuration parameters **Object Maintenance (OBJMAINT).** Creating and dropping data objects **System Administration (SYSADMIN).** Operations requiring `SYSADM`, `SYSMAINT`, or `SYSCTRL` privileges **User Validation (VALIDATE).** Authentication of users or retrieval of system security information **Operation Context (CONTEXT).** Database operation context performed. Helps when interpreting the audit log file. See the IBM DB2 documentation for more information about how the operation context of a DB2 database is audited. In addition to these categories, you can audit successes, failures, or both.	Implement your best practices for IBM DB2 database auditing

2

Registering Source Databases and Collectors

This chapter contains:

- General Steps for Adding Sources and Deploying Collectors
- Checking and Setting Environment Variables
- Registering Oracle Database Sources and Collectors
- Registering Microsoft SQL Server Database Sources and Collector
- Registering Sybase ASE Database Sources and Collector
- Registering IBM DB2 Database Sources and Collector
- Starting the Collection Agents
- Starting the Collectors
- Checking the Status of the Collectors
- Checking If the Collectors Are Collecting Audit Records

2.1 General Steps for Adding Sources and Deploying Collectors

You must perform the following general tasks to add source databases to Oracle Audit Vault and then deploy collectors:

1. For Linux and UNIX platforms, check and set environment variables in the shells in which you will be interacting with the Audit Vault Server and the Audit Vault Collection Agent.

   See Section 2.2.

2. Add an Oracle source database and collectors using the `AVORCLDB` command-line utility.

   See Section 2.3.

3. To add a Microsoft SQL Server source database and collector, use the `AVMSSQLDB` command-line utility

   See Section 2.4.

4. To add a Sybase ASE source database and collector, use the `AVSYBDB` command-line utility

   See Section 2.5.

5. To add an IBM DB2 source database and collector, use the `AVDB2DB` command-line utility.

   See Section 2.6.

6. Start the collection agents and collectors using the `AVCTL` command-line utility.

   See Section 2.7 and Section 2.8.

7. Periodically ensure that the collectors are running and collecting audit data.

   See Section 2.9 and Section 2.10.

2.2 Checking and Setting Environment Variables

This section contains:

- About Checking and Setting Linux and UNIX Environment Variables
- Setting the Audit Vault Server Linux and UNIX Environment Variables
- Setting the Collection Agent Linux and UNIX Environment Variables
- Using the Collection Agent in a Microsoft Windows Environment
- Setting the Oracle Source Database Linux and UNIX Environment Variables

2.2.1 About Checking and Setting Linux and UNIX Environment Variables

For Linux and UNIX platforms, you must set environment variables before you begin the procedures in this chapter. You set these variables in the three shells that you will use to perform the configuration. *Keep these shells open throughout the configuration process.* You will need to access them periodically as you complete the configuration steps. If you reopen a shell, then you must reset its environment variables.

2.2.2 Setting the Audit Vault Server Linux and UNIX Environment Variables

You use the Audit Vault Server shell to interact with the Audit Vault Server. To set the environment variables for the Audit Vault Server, you can run either of two scripts, `coraenv` (for the C shell) or `oraenv` (for the Bourne, Bash, or Korn shell).

Table 2–1 describes how the `coraenv` and `oraenv` scripts set the environment variables.

Table 2–1 Audit Vault Server Environment Variable Settings

Environment Variable	Behavior
`ORACLE_HOME`	Sets to the Audit Vault Server home directory.
`ORACLE_SID`	Prompts for the Oracle system identifier (SID) for the Audit Vault Server. By default, this SID is `av`.
`PATH`	Appends `$ORACLE_HOME/bin` to your `PATH` environment variable.
`LD_LIBRARY_PATH`	Appends `$ORACLE_HOME/lib` to your `LD_LIBRARY_PATH` environment variable setting. Applies to Linux x86, Linux x86_64, and Solaris SPARC_64 installations only.
`SHLIB_PATH`	Appends `$ORACLE_HOME/lib` to your `SHLIB_PATH` environment variable setting. Applies to HP-UX installations only.
`LIBPATH`	Appends `$ORACLE_HOME/lib` to your `LIBPATH` environment variable setting. Applies to AIX installations only.

To set environment variables for the Audit Vault Server shell:

1. In the server where you installed the Oracle Audit Vault Server, open a shell.
2. Run one of the following scripts, which are located in the `/usr/local/bin` directory:
   - **C shell:** `coraenv`
   - **Bourne, Bash, or Korn shell:** `oraenv`
3. To test that the script was successful, try invoking the following command:

   ```
   $ avctl -help
   ```

   It should return help information for the `AVCTL` utility, and the only way it can do that is if the `ORACLE_HOME` and `PATH` environment variables are correctly set. If the scripts fail, then manually set the environment variables listed in Table 2–1.
4. If you plan to add Microsoft SQL Server, Sybase ASE, or IBM DB2 source databases to Oracle Audit Vault, then set the `LANG` and `NLS_LANG` environment variables.

   For example:

   - **C shell:**

     ```
     setenv LANG de_DE.UTF-8

     setenv NLS_LANG GERMAN_GERMANY.AL32UTF8
     ```

   - **Bourne, Bash, or Korn shell:**

     ```
     LANG=de_DE.UTF-8

     NLS_LANG=GERMAN_GERMANY.AL32UTF8
     ```

   See *Oracle Database Globalization Support Guide* for more information about the `NLS_LANG` environment variable, including supported character sets and languages.

   Oracle Audit Vault supports the following languages for the `LANG` environment variable:

`en`: English	`ja`: Japanese
`de`: German	`ko`: Korean
`es`: Spanish	`pt_BR`: Brazilian Portuguese
`fr`: French	`zh_CN`: Simplified Chinese
`it`: Italian	`zh_TW`: Traditional Chinese

   Optionally, you can set the `LANG` environment variable in the `.profile` or `.cshrc` file.

   You do not need to set this variable for the Oracle Database `AVORCLDB` utility. This utility automatically uses the `NLS_LANG` environment variable setting, which is set during installation. See *Oracle Database Globalization Support Guide* for more information about language support for Oracle Database.
5. Leave the Audit Vault Server shell open for the remaining procedures in this chapter.

2.2.3 Setting the Collection Agent Linux and UNIX Environment Variables

To set environment variables for the Audit Vault collection agent shell:

1. In the server where you installed the Audit Vault collection agent, open a shell.
2. Check and manually set the `ORACLE_HOME` environment variable to the Audit Vault collection agent home directory.
3. Check and set the `LD_LIBRARY_PATH` environment variable to include `$ORACLE_HOME/lib`.
4. Check and set the `PATH` environment variable to include `$ORACLE_HOME/bin`. Be sure that you append this information to the existing `PATH` information.
5. Ensure that the following environment variables are not set: `ORACLE_SID`, `TNS_ADMIN`, and `TWO_TASK`.
6. To test that you correctly set these environment variables, try invoking the following command:

   ```
   $ avctl -help
   ```

   It should return help information for the `AVCTL` utility, and the only way it can do that is if the `ORACLE_HOME` and `PATH` environment variables are correctly set.
7. If you plan to add Microsoft SQL Server, Sybase ASE, or IBM DB2 databases to Oracle Audit Vault, then set the `LANG` and `NLS_LANG` environment variables.

   See Step 4 under Section 2.2.2 for instructions.
8. Leave the Audit Vault collection agent shell open for the remaining procedures in this chapter.

2.2.4 Using the Collection Agent in a Microsoft Windows Environment

If you installed the collection agent on Microsoft Windows, do not set any environment variables. Instead, run any collection agent-specific commands (such as `avctl start_oc4j` or `avctl stop_oc4j`) from the collection agent home directory, which is *`ORACLE_HOME`*`\`*`agent_dir`*`\bin`.

2.2.5 Setting the Oracle Source Database Linux and UNIX Environment Variables

To set the environment variables for the source database, you can run the same scripts, `coraenv` or `oraenv`, that you used to set the Audit Vault Server environment variables. Table 2–1 on page 2-2 describes how these scripts set the environment variables, except that for the source database, they set the `ORACLE_SID` variable to `orcl`, unless you have given it a different name during installation.

To set environment variables for the source database:

1. In the server where you installed the Oracle source database, open a shell.
2. From the `/usr/local/bin` directory, run one of the following scripts:
   - **C shell:** `coraenv` script
   - **Bourne, Bash, or Korn shell:** `oraenv` script
3. Leave the Oracle source database shell open for the remaining procedures in this chapter.

2.3 Registering Oracle Database Sources and Collectors

This section contains:

- Step 1: If Necessary, Create a Password File
- Step 2: Create a User Account on the Oracle Source Database
- Step 3: Verify That the Source Database Is Compatible with the Collectors
- Step 4: Register the Oracle Source Database with Oracle Audit Vault
- Step 5: Add the Oracle Collectors to Oracle Audit Vault
- Step 6: Enable the Audit Vault Agent to Run the Oracle Database Collectors

2.3.1 Step 1: If Necessary, Create a Password File

If you use Oracle Database Vault to protect the Oracle source database, you must have a password file created. A connection to the source database using the `SYSDBA` or `SYSOPER` privilege succeeds only if the password file has been created. Some later versions of Oracle Database Vault enable operating system authentication by default. To create the password file in the source database, use the `orapwd` utility. See *Oracle Database Administrator's Guide* for more information about `orapwd`. To use `orapwd` to enable or disable `SYSDBA` connections, see *Oracle Audit Vault Server Installation Guide for Linux x86*.

For example:

```
$ orapwd file=$ORACLE_HOME/dbs/av_pwd password=sys_password
```

2.3.2 Step 2: Create a User Account on the Oracle Source Database

The collectors that you will configure later must use this user account to access audit data from the Oracle source database.

To create the user account:

1. Access the shell used by the Oracle source database.
2. Log in to SQL*Plus as a user who has been granted the `CREATE USER` privilege.

   If the source database is protected by Oracle Database Vault, log in as a user who has been granted the `DV_ACCTMGR` (Database Vault Account Manager) role.

   For example:

   ```
   $ sqlplus avadmindva
   Enter password: password
   Connected.
   ```

3. Create the Oracle source database user account.

   For example:

   ```
   SQL> CREATE USER srcuser_ora IDENTIFIED BY password;
   ```

4. Connect as user `SYS` with the `SYSDBA` privilege.

   ```
   SQL> CONNECT SYS/AS SYSDBA
   Enter password: password
   ```

5. Run the `zarsspriv.sql` script from either the Audit Vault Server or Audit Vault collection agent on Oracle soruce database.

This script grants the Oracle source database user account the privileges needed to enable the collectors to access audit data. By default, this script is located in the `$ORACLE_HOME/av/scripts/streams/source` directory in both the Audit Vault Server and the Audit Vault collection agent Oracle home directories.

Use the following syntax:

```
zarsspriv.sql srcusr mode
```

In this specification:

- *srcusr*: Enter the name of the user account that you just created.
- *mode*: Specify one of the following modes. Enter the modes in uppercase letters.
  - `SETUP`: For the OSAUD and DBAUD collectors, and for policy management
  - `REDO_COLL`: For the REDO log collector; includes all privileges that are granted using the argument mode `SETUP`.

For example, to specify the `SETUP` mode for user `srcuser_ora`:

```
SQL> @/oracle/product/10.2.3/av/scripts/streams/source/zarsspriv.sql
Enter value for 1: srcuser_ora
Enter value for 2: SETUP

Granting privileges to SRCUSER_ORA ... Done.
```

6. Connect as the source user that you created in Step 3, and then check that the privileges were granted.

```
SQL> CONNECT srcuser_ora
Enter password: password
Connected.

SQL> SELECT * FROM SESSION_PRIVS;
SQL> SELECT * FROM SESSION_ROLES;
```

The output for each `SELECT` statement should list the privileges and roles that are listed in the `zarsspriv.sql` file, such as the `CREATE SESSION` privilege and the `RESOURCE` role.

7. If the source database has Oracle Database Vault installed, log in as a user who has been granted the `DV_OWNER` (Database Vault Owner) role, and then add the source user to the Oracle Data Dictionary realm.

For example:

```
SQL> CONNECT dbvowner
Enter password: password
Connected.

SQL> EXEC DBMS_MACADM.ADD_AUTH_TO_REALM('Oracle Data Dictionary', 'SRCUSER_
ORA', null, dbms_macutl.g_realm_auth_participant);
SQL> COMMIT;
```

8. If the source database has Oracle Database Vault installed, grant the Oracle source database user account the `DV_SECANALYST` role.

The DV_SECANALYST role enables the user to run Oracle Database Vault reports and monitor Oracle Database Vault. This role also enables the Oracle source database user to collect Database Vault audit trail data from the source database.

For example:

```
SQL> GRANT DV_SECANALYST TO srcuser_ora;
```

9. Exit SQL*Plus.
10. Leave this shell open.

2.3.3 Step 3: Verify That the Source Database Is Compatible with the Collectors

Now you are ready to verify that the Oracle source database is compatible with the collector type in the Audit Vault collection agent home.

To verify the Oracle source database compatibility:

1. Access either the shell used for the Audit Vault Server or the collection agent.

   If you want to use the collection agent location, and if you installed the collection agent on Microsoft Windows, go to the *ORACLE_HOME**agent_dir*\bin directory.

2. Run the following command and note the host, port, and service settings:

   ```
   $ lsnrctl status
   ```

3. Run the `avorcldb verify` command, using the values that the LSNRCTL utility returned.

   You must specify the host name, port number, and service name. Typically, for Oracle Database, the host is the fully qualified domain name or the IP address of the server on which the Oracle source database is running, and the port number is 1521.

   For example, assume that the host is `hrdb.example.com`, the port number is `1521`, the service name is `orcl`, and the user account is `srcuser_ora`:

   ```
   $ avorcldb verify -src hrdb.example.com:1521:orcl -colltype ALL
   Enter Source user name: srcuser_ora
   Enter Source password: password
   ```

   See Section 8.10 for detailed information about the `avorcldb verify` command.

4. Do not close this shell.

The AVORCLDB utility checks if an Audit Vault collector can be run against the source database configuration.

Example 2–1 shows what happens if the Oracle source database is not properly configured. In this case, you must set the initialization parameters listed in the output before you can use the REDO log collector.

Example 2–1 Partially Successful Verify Operation of Source Compatibility with the Collectors

```
$ avorcldb verify -src hrdb.example.com:1521:orcl -colltype ALL
Enter Source user name: srcuser_ora
Enter Source password: password

source hrdb.EXAMPLE.COM verified for OS File Audit Collector
source hrdb.EXAMPLE.COM verified for Aud$/FGA_LOG$ Audit Collector
Source database must be in ARCHIVELOG mode to use REDO Log collector
```

```
Incorrect database compatibility 9.2.0; recommended value is 10.2.0.0.0
Parameter _JOB_QUEUE_INTERVAL not set; recommended value range [1 - ANY_VALUE]
Parameter JOB_QUEUE_PROCESSES = 0 not in recommended value range [4 - ANY_VALUE]
Parameter AQ_TM_PROCESSES = 0 is not in required value range [4 - ANY_VALUE]
Parameter UNDO_RETENTION = 900 not in recommended value range [3600 - ANY_VALUE]
Parameter GLOBAL_NAMES = false not set to recommended value true
Please set the above init.ora parameters to recommended values
```

After you correct the problems (in this case, setting all those missing or incorrect initialization parameters), rerun the `avorcldb verify` command to ensure that the result is as you want it. Example 2–2 shows what happens after this source database has been properly configured. See also Chapter 12, "REDO Collector Database Reference."

Example 2–2 Successful Verify Operation of Source Compatibility with the REDO Collector

```
$ avorcldb verify -src hrdb.example.com:1521:orcl -colltype REDO
Enter Source user name: srcuser_ora
Enter Source password: password

source hrdb.EXAMPLE.COM verified for REDO Log Audit Collector collector
```

2.3.4 Step 4: Register the Oracle Source Database with Oracle Audit Vault

To register the Oracle source database with Oracle Audit Vault:

1. Access the shell used for the Audit Vault Server.
2. Run the `avorcldb add_source` command.

   For example:

   ```
   $ avorcldb add_source -src hrdb.example.com:1521:orcl
                         -desc 'HR Database'
                         -agentname agent1
   Enter Source user name: srcuser_ora
   Enter Source password: password

   Adding source...
   Source added successfully.
   source successfully added to Audit Vault

   remember the following information for use in avctl
   Source name (srcname): HRDB.EXAMPLE.COM
   Storing user credentials in wallet...
   Create credential oracle.security.client.connect_string3
   done.
   Mapping Source to Agent...
   ```

   In this example:

   - `-src`: Enter the source database connection information: host name, port number, and service name, separated by a colon. If you are unsure of this information, run the `lsnrctl status` command on the computer where you installed the source database.
   - `-desc`: Optionally, enter a brief description for the source database.
   - `-agentname`: Optionally, create a name for the collector agent to be associated with this source database. However, you must specify an agent name if auditors plan to configure policy management using the Audit Vault Console.

- `Source user name` and `password`: Enter the user account information that you created in Step 2: Create a User Account on the Oracle Source Database.

See Section 8.3 for detailed information about the `avorcldb add_source` command.

3. Note the return value from the output.

   You will need this value, which represents the global database name, for subsequent steps in this section. In this example, the return value is `HRDB.EXAMPLE.COM`.

4. Do not close this shell.

2.3.5 Step 5: Add the Oracle Collectors to Oracle Audit Vault

You can add one or more collectors to Oracle Audit Vault, depending on your needs. The available collector types are listed in Table 1–4 on page 1-6.

To add a collector to Oracle Audit Vault:

1. If you plan to use the OSAUD collector, access the shell used for the Oracle source database.
2. Log in to SQL*Plus as `SYS` with the `SYSDBA` privilege.

```
$ sqlplus sys/as sysdba
Enter password: password
Connected.
```

3. Set the maximum operating system file size to a setting equal to or less than 204800.

   If the operating system file grows larger than 2 GB, then the OSAUD collector ignores all audit records created past this size. Use the following SQL statement to set the maximum size to 102400 KB, which translates as 2 GB.

```
BEGIN
  DBMS_AUDIT_MGMT.SET_AUDIT_TRAIL_PROPERTY(
   AUDIT_TRAIL_TYPE            =>  DBMS_AUDIT_MGMT.AUDIT_TRAIL_OS,
   AUDIT_TRAIL_PROPERTY        =>  DBMS_AUDIT_MGMT.OS_FILE_MAX_SIZE,
   AUDIT_TRAIL_PROPERTY_VALUE  =>  204800);
END;
/
```

   Afterwards, when the operating system exceeds 2 GB, then Oracle Database stops appending audit records to the current file, and then creates a new file to resume the audit data collection.

   For reference information about the `DBMS_AUDIT_MGMT` PL/SQL package, see Chapter 14.

4. Access the shell used for the Audit Vault Server.
5. Run the `avorcldb add_collector` command to add the collectors you want.

   For example:

```
avorcldb add_collector -srcname HRDB.EXAMPLE.COM
                       -agentname agent1
                       -colltype OSAUD
                       -orclhome /u01/app/oracle/product/10.2.0/db_1
```

   In this example:

- `-srcname`: Create a name for this source database, which Oracle Audit Vault will refer to when collecting audit data. Remember that the source name is case-sensitive.
- `-agentname`: Enter the name for the agent that you created in Step 4: Register the Oracle Source Database with Oracle Audit Vault.
- `-colltype`: Enter `OSAUD`, `DBAUD`, or `REDO`. If you plan to specify REDO, you must include the `-av` argument, which specifies the connection information for Oracle Audit Vault used for the database link from the source database to Oracle Audit Vault. See Section 8.2 more information about the `-av` argument.
- `-orclhome`: Enter the Oracle source database home directory. For Microsoft Windows installations of Oracle Database, enter the path using forward slashes, or if you want to use back slashes, enclose the path in double quotation marks.

See Section 8.2 for detailed information about the `avorcldb add_collector` command.

6. Optionally, modify the attributes associated with the collector.

   The collector has a set of default attributes. You can modify these by using the `avorcldb alter_collector` command. See Section 8.4.

7. Do not close this shell.

Example 2–3 shows how to add the OSAUD collector to Oracle Audit Vault for UNIX platforms. You must include the `-orclhome` *`orclhome`* parameter to specify the location of the source database as an absolute path, if `u01/app` is the Oracle base directory.

Example 2–3 Adding the OSAUD Collector to Oracle Audit Vault for UNIX Platforms

```
$ avorcldb add_collector -srcname hrdb.example.com
                         -agentname agent1
                         -colltype OSAUD
                         -orclhome /u01/app/oracle/product/10.2.0/db_1

source HRDB.EXAMPLE.COM verified for OS File Audit Collector collector
Adding collector...
Collector added successfully.
collector successfully added to Audit Vault

remember the following information for use in avctl
Collector name (collname): OSAUD_Collector
```

Example 2–4 shows how to add the OSAUD collector to Oracle Audit Vault on Microsoft Windows for the event log and XML audit trail. You must include the `-orclhome` *`orclhome`* parameter to specify the location of the source database. Use slashes (`/`) instead of backslashes (`\`) for the Microsoft Windows path. If you want to use backslashes, enclose the path in double quotation marks. For example:

```
-orclhome "c:\oracle\product\10.2.0\db_1"
```

Example 2–4 Adding the OSAUD Collector to Oracle Audit Vault on Microsoft Windows

```
$ avorcldb add_collector -srcname HRDB.EXAMPLE.COM
                         -agentname agent1
                         -colltype OSAUD
                         -orclhome c:/oracle/product/10.2.0/db_1
```

```
source HRDB.EXAMPLE.COM verified for Windows Event Log Audit Collector collector
Adding collector...
Collector added sucessfully.
collector successfully added to Audit Vault

remember the following information for use in avctl
Collector name (collname): OSAUD_Collector
```

Example 2–5 shows how to add the DBAUD collector to Oracle Audit Vault.

Example 2–5 Adding the DBAUD Collector to Oracle Audit Vault

```
$ avorcldb add_collector -srcname HRDB.EXAMPLE.COM
                         -agentname agent1 -colltype DBAUD

source HRDB.EXAMPLE.COM verified for Aud$/FGA_LOG$ Audit Collector collector
Adding collector...
Collector added successfully.
collector successfully added to Audit Vault

remember the following information for use in avctl
Collector name (collname): DBAUD_Collector
```

Example 2–6 shows how to add the REDO collector to Oracle Audit Vault. Note that you must supply the `-av` argument for this collector type.

Example 2–6 Adding the REDO Collector to Oracle Audit Vault

```
$ avorcldb add_collector -srcname HRDB.EXAMPLE.COM
                         -agentname agent1
                         -colltype REDO
                         -av hrdb.example.com:1521:orcl

source HRDB.EXAMPLE.COM verified for REDO Log Audit Collector collector
Adding collector...
Collector added successfully.
collector successfully added to Audit Vault

remember the following information for use in avctl
Collector name (collname): REDO_Collector
initializing REDO Collector
setting up APPLY process on Audit Vault server
setting up CAPTURE process on source database
```

Note: If the REDO collector does not initialize, the `APPLY` process on the Audit Vault Server and `CAPTURE` process on the source database cannot start. This problem happens if the source user account does not have the correct privileges. Ensure that you ran the `zarsspriv.sql` script, described in Section 2.3.2.

2.3.6 Step 6: Enable the Audit Vault Agent to Run the Oracle Database Collectors

You now are ready to add the collection agent credentials to the Oracle source database. This process adds the source user credentials to the wallet, creates a database alias in the wallet for the source user, and verifies the connection to the source using the wallet. This way, the Audit Vault collection agent can run the Oracle Database collectors. You must complete this step so that the collectors can start properly.

To enable to Audit Vault agent to run the Oracle Database collectors:

1. Access the shell used for the Audit Vault collection agent.

   If you have closed this shell, see the following sections:

   - Section 2.2.3 describes how to set environment variables for the collection agent.
   - If you installed the collection agent on Microsoft Windows, do not set any environment variables. Instead, go to the *ORACLE_HOME**agent_dir*\bin directory.

2. Use the `avorcldb setup` command to add the collection agent credentials.

   For example:

```
$ avorcldb setup -srcname hrdb.example.com

Enter Source user name: srcuser_ora
Enter Source password: password

adding credentials for user srcuser_ora for connection [SRCDB1]
Storing user credentials in wallet...
Create credential oracle.security.client.connect_string3
done.
updated tnsnames.ora with alias [SRCDB1] to source database
verifying SRCDB1 connection using wallet
```

   In this example:

   - `-srcname`: Enter the name of the source database that you plan to use.
   - `Source user name` and `password`: Enter the source database user name and password that you created in Step 2: Create a User Account on the Oracle Source Database.

   See Section 8.9 for detailed information about the `avorcldb setup` command.

3. Do not close this shell.

This step completes the registration for the Oracle source database and its collectors. Next, you must start the collection agents and collectors. See Section 2.7 and Section 2.8 for more information.

2.4 Registering Microsoft SQL Server Database Sources and Collector

This section contains:

- Step 1: Download the SQL Server 2005 Driver for JDBC
- Step 2: Create a User Account on the Microsoft SQL Server Source Database
- Step 3: Verify That the Source Database Is Compatible with the Collector
- Step 4: Register the SQL Server Source Database with Oracle Audit Vault
- Step 5: Add the MSSQLDB Collector to Oracle Audit Vault
- Step 6: Enable the Audit Vault Agent to Run the MSSQLDB Collector

2.4.1 Step 1: Download the SQL Server 2005 Driver for JDBC

Ensure that you have downloaded the SQL Server 2005 Driver for JDBC (`sqljdbc.jar`) to the `$ORACLE_HOME/jlib` directories in both the Audit Vault Server and Audit Vault Agent homes. This driver provides high performance native

access to Microsoft SQL Server 2000 and 2005 database data sources. Ensure that this jar file is present in the Oracle Audit Vault OC4J before starting the agent OC4J. The MSSQLDB collector uses this driver to collect audit data from Microsoft SQL Server databases.

See Also:

- *Oracle Audit Vault Server Installation Guide for Linux x86* for information about downloading and copying JDBC driver files for Microsoft SQL Server
- *Oracle Audit Vault Collection Agent Installation Guide* for information about downloading and copying JDBC driver files for Microsoft SQL Server
- *Oracle Audit Vault Collection Agent Installation Guide* to ensure that the `sqljdbc.jar` file is present in the Oracle Audit Vault OC4J before starting the agent OC4J

2.4.2 Step 2: Create a User Account on the Microsoft SQL Server Source Database

The collector that you will configure later must use this user account to access audit data from the Microsoft SQL Server source database. After you create the user account, the privileges that you assign to this user depend on whether the source database is Microsoft SQL Server 2000 or 2005.

To create the user account:

1. Log in to the Microsoft SQL Server source database.
2. Create a user account.

   For example, to create a user account named `srcuser_mss`:

   ```
   EXEC sp_addlogin srcuser_mss, password
   ```

For a Microsoft SQL Server 2005 database, grant this user the `alter_trace` privilege.

1. Log in as the `SYSADMIN` user.
2. Run the following command to grant the alter trace privilege to the user.

   For example:

   ```
   GRANT ALTER TRACE TO srcuser_mss
   ```

For a Microsoft SQL Server 2000 database, grant the user the `SYSADMIN` fixed server role.

1. Click **Security**.
2. Click **Logins**.
3. Right-click the login you created (for example, `srcuser_mss`).
4. Click **Properties**.
5. On the left pane, click **Server Roles**.
6. Select the **sysadmin option** setting, and then click **OK**.

2.4.3 Step 3: Verify That the Source Database Is Compatible with the Collector

Next, you are ready to verify that the Microsoft SQL Server source database is compatible with the collector type in the Audit Vault collection agent home.

To verify the source database compatibility:

1. Access either the shell used for the Audit Vault Server or the collection agent.

   If you want to use the collection agent location, and if you installed the collection agent on Microsoft Windows, go to the *ORACLE_HOME**agent_dir*\bin directory.

2. Run the `avmssqldb verify` command.

   You must specify the host name and port number. Typically, for Microsoft SQL Server, the host is the fully qualified domain name or the IP address of the server on which the SQL Server source database is running, and the port number is 1433.

   For example, assume that the host is `hrdb.example.com` and the port number is `1433`, and the user account is `srcuser_mss`:

   ```
   $ avmssqldb verify -src hrdb.example.com:1433
   Enter a username : srcuser_mss
   Enter a password: password

   ***** Source Verified *****
   ```

   See Section 9.10 for detailed information about the `avmssqldb verify -src` command.

3. Do not close this shell.

2.4.4 Step 4: Register the SQL Server Source Database with Oracle Audit Vault

To register the SQL Server source database with Oracle Audit Vault:

1. Access the shell for the Audit Vault Server.

2. Run the `avmssqldb add_source` command.

   For example:

   ```
   $ avmssqldb add_source -src hrdb.example.com:1433 -srcname mssqldb4 -desc 'HR
   Database'
   Enter a username :srcuser_mss
   Enter a password : password

   ***** Source Verified *****
   ***** Source Added Successfully *****
   ```

   In this example:

   - `-src`: Enter the fully qualified domain name (or IP address) and port number for the source database that you specified in Step 3: Verify That the Source Database Is Compatible with the Collector.
   - `-srcname`: Create a name for the source database. Oracle Audit Vault refers to this name when it collects audit data.
   - `-desc`: Optionally, enter a brief description for the source database.
   - `username` and `password`: Enter the user name and password that you created in Step 2: Create a User Account on the Microsoft SQL Server Source Database.

   See Section 9.3 for detailed information about the `avmssqldb add_source` command.

3. Do not close this shell.

2.4.5 Step 5: Add the MSSQLDB Collector to Oracle Audit Vault

Now you are ready to add the MSSQLDB collector to Oracle Audit Vault. By default, the MSSQLDB collector collects audit records from all audit trails that have been enabled in the source database: C2 audit logs, server-side trace logs, and the Windows Event log.

To add the MSSQLDB collector to Oracle Audit Vault:

1. Access the shell used for the Audit Vault Server.
2. Run the `avmssqldb add_collector` command.

   For example:

   ```
   $ avmssqldb add_collector -srcname mssqldb4 -agentname agent1
   Enter a username :srcuser_mss
   Enter a password : password

   ***** Collector Added Successfully*****
   ```

   In this example:

   - `-srcname`: Enter the name of the SQL Server source database that you verified in Step 3: Verify That the Source Database Is Compatible with the Collector.
   - `-agentname`: Create a name for the agent.

   See Section 9.2 for detailed information about the `avmssqldb add_collector` command.
3. Optionally, modify the attributes associated with the MSSQLDB collector.

   The MSSQLDB collector has a set of default attributes. You can modify these by using the `avssqldb alter_collector` command. See Section 9.4.
4. Do not close this shell.

2.4.6 Step 6: Enable the Audit Vault Agent to Run the MSSQLDB Collector

Next, you must add the collection agent credentials to the Microsoft SQL Server source database. This process adds the source user credentials to the wallet, creates a database alias in the wallet for the source database user, and verifies the connection to the source database using the wallet. This way, the Oracle Audit Vault collection agent can run the MSSQLDB collector. You must complete this step so that the collectors can start properly.

To enable the Oracle Audit Vault agent to run the MSSQLDB collector:

1. Access the shell used for the Audit Vault collection agent.

   If you installed the collection agent on Microsoft Windows, go to the *ORACLE_HOME**agent_dir*\bin directory.
2. Run the `avmssqldb setup` command.

   For example:

   ```
   $ avmssqldb setup -srcname mssqldb4
   Enter a username :srcuser_mss
   Enter a password : password

   ***** Credentials Successfully added *****
   ```

In this example:

- `-srcname`: Enter the source database name that you specified in Step 3: Verify That the Source Database Is Compatible with the Collector.
- `username` and `password`: Enter the user name and password that you created in Step 2: Create a User Account on the Microsoft SQL Server Source Database.

See Section 8.9 for detailed information about the `avmssqldb setup` command.

3. Do not close this shell.

This step completes the registration for the Microsoft SQL Server source database and its collector. Next, you must start the collection agent and collector. See Section 2.7 and Section 2.8 for more information.

2.5 Registering Sybase ASE Database Sources and Collector

This section contains:

- Step 1: Download the jConnect for JDBC Driver
- Step 2: Create a User Account on the Sybase ASE Source Database
- Step 3: Verify That the Source Database Is Compatible with the Collector
- Step 4: Register the Sybase ASE Source Database with Oracle Audit Vault
- Step 5: Add the SYBDB Collector to Oracle Audit Vault
- Step 6: Enable the Audit Vault Agent to Run the SYBDB Collector

2.5.1 Step 1: Download the jConnect for JDBC Driver

Ensure that you have downloaded the jConnect for JDBC driver JDBC (`jconn3.jar`) to the `$ORACLE_HOME/jlib` directories in both the Audit Vault Server and Audit Vault Agent homes. This driver provides high performance native access to Sybase ASE database data sources. Ensure that this jar file is present in the Oracle Audit Vault OC4J before starting the agent OC4J. The SYBDB collector uses this driver to collect audit data from Sybase ASE databases.

See Also:

- *Oracle Audit Vault Server Installation Guide for Linux x86* for information about downloading and copying JDBC driver files for Sybase ASE
- *Oracle Audit Vault Collection Agent Installation Guide* for information about downloading and copying JDBC driver files for Sybase ASE
- *Oracle Audit Vault Collection Agent Installation Guide* to ensure that the `sqljdbc.jar` file is present in the Oracle Audit Vault OC4J before starting the agent OC4J

2.5.2 Step 2: Create a User Account on the Sybase ASE Source Database

The collector that you will configure later must use this user account to access audit data from the Sybase ASE source database.

To create the user account:

1. Log in to the Sybase ASE source database.
2. Create a user account.

   For example:

   ```
   sp_addlogin srcuser_syb, password
   ```

3. Add this user to the Sybase ASE source database.

   ```
   sp_adduser srcuser_syb
   ```

4. Grant the `SSO_role` privilege to the source user.

   ```
   grant role sso_role to srcusr_syb
   ```

2.5.3 Step 3: Verify That the Source Database Is Compatible with the Collector

Now you are ready to verify that the Sybase ASE source database is compatible with the collector type in the Audit Vault collection agent home:

To verify the Sybase ASE source database compatibility:

1. Access either the shell used for the Audit Vault Server or the collection agent.

   If you want to use the collection agent location, and if you installed the collection agent on Microsoft Windows, go to the *ORACLE_HOME*`\`*agent_dir*`\bin` directory.

2. Run the `avsybdb verify` command.

   You must specify the host name and port number. Typically, for Sybase ASE, the host is the fully qualified domain name or IP address of the server on which the Sybase ASE source database is running, and the port number is 5000.

   For example, assume that the host is `hrdb.example.com` and the port number is `5000`, and the user account is `srcuser_syb`:

   ```
   $ avsybdb verify -src hrdb.example.com:5000
   Enter a username :srcuser_syb
   Enter a password : password

   ***** Source Verified *****
   ```

   See Section 10.10 for detailed information about the `avsybdb verify` command.

3. Do not close this shell.

2.5.4 Step 4: Register the Sybase ASE Source Database with Oracle Audit Vault

To register the Sybase ASE source database with Oracle Audit Vault:

1. Access the shell used for the Audit Vault Server.
2. Run the `avsybdb add_source` command.

   For example:

   ```
   $ avsybdb add_source -src hrdb.example.com:5000 -srcname  sybdb4
   ```

```
Enter a username :srcuser_syb
Enter a password : password

***** Source Verified *****
***** Source Added Successfully *****
```

In this example:

- `-src`: Enter the fully qualified domain name (or IP address) and port number for the source database that you verified in Step 3: Verify That the Source Database Is Compatible with the Collector.
- `-srcname`: Create a name for this source database. Oracle Audit Vault refers to this name when it collects audit data.
- `username` and `password`: Enter the user name and password that you created in Step 2: Create a User Account on the Sybase ASE Source Database.

See Section 10.3 for detailed information about the `avsybdb add_source` command.

3. Do not close this shell.

2.5.5 Step 5: Add the SYBDB Collector to Oracle Audit Vault

To add the SYBDB collector to Oracle Audit Vault:

1. Access the shell used for the Audit Vault Server.
2. Run the `avsybdb add_collector` command.

   For example:

```
$ avsybdb add_collector -srcname sybdb4 -agentname agent1
Enter a username :srcuser_syb
Enter a password : password

***** Collector Added Successfully*****
```

   In this example:

   - `-srcname`: Create a name for the source database. Oracle Audit Vault refers to this name when collecting audit data.
   - `-agentname`: Create a name for the agent.
   - `username` and `password`: Enter the user name and password that you created in Step 2: Create a User Account on the Sybase ASE Source Database.

   See Section 10.2 for detailed information about the `avsybdb add_collector` command.

3. Optionally, modify the attributes associated with the collector.

   The collector has a set of default attributes. You can modify these by using the `avsybdb alter_collector` command. See Section 10.4.

4. Do not close this shell.

2.5.6 Step 6: Enable the Audit Vault Agent to Run the SYBDB Collector

You now are ready to configure the collection agent credentials to the Sybase ASE source database. This process adds the source user credentials to the wallet, creates a database alias in the wallet for the source user, and verifies the connection to the

source using the wallet. This way, the Oracle Audit Vault collection agent can run the SYBDB collector. You must complete this step so that the collectors can start properly.

To enable the Oracle Audit Vault collection agent to run the SYBDB collector:

1. Access the shell used for the Oracle Audit Vault collection agent.

   If you installed the collection agent on Microsoft Windows, go to the *ORACLE_HOME**agent_dir*\bin directory.

2. Run the `avsybdb setup` command.

   For example:

   ```
   $ avsybdb setup -srcname sybdb4
   Enter a username :srcuser_syb
   Enter a password : password

   ***** Credentials Successfully added *****
   ```

   In this example:

   - `-srcname`: Enter the source database name that you created in Step 5: Add the SYBDB Collector to Oracle Audit Vault.
   - `username` and `password`: Enter the user name and password that you created in Step 2: Create a User Account on the Sybase ASE Source Database.

   See Section 10.9 for detailed information about the `avsybdb setup` command.

3. Do not close this shell.

This step completes the registration for the Sybase ASE source database and its collector. Next, you must start the collection agent and collector. See Section 2.7 and Section 2.8 for more information.

2.6 Registering IBM DB2 Database Sources and Collector

This section contains:

- Step 1: Copy the DB2 Data Server Driver for JDBC and SQLJ to the Audit Vault Homes
- Step 2: Designate a User Account on the IBM DB2 Source Database
- Step 3: Verify That the Source Database Is Compatible with the Collector
- Step 4: Register the IBM DB2 Source Database with Oracle Audit Vault
- Step 5: Add the DB2DB Collector to Oracle Audit Vault
- Step 6: Convert the Binary DB2 Audit File to an ASCII Text File

2.6.1 Step 1: Copy the DB2 Data Server Driver for JDBC and SQLJ to the Audit Vault Homes

Copy the IBM Data Server Driver for JDBC and SQLJ (`db2jcc.jar`) to the `$ORACLE_HOME/jlib` directories in both the Audit Vault Server and Audit Vault Agent homes. Oracle Audit Vault requires version 3.50 or later of the driver. This version of the `db2jcc.jar` file is available in either IBM DB2 UDB version 9.5 or IBM DB2 Connect version 9.5 or later.

This driver provides high performance native access to IBM DB2 database data sources. The DB2 collector uses this driver to collect audit data from IBM DB2

databases, so the driver must be present in Oracle Audit Vault OC4J before you can start the agent OC4J.

You can verify the version of this jar file that is currently installed as follows:

1. Ensure that the directory path to the `db2jcc.jar` file is included in the `CLASSPATH` environment variable setting.
2. Run the following command:

```
java com.ibm.db2.jcc.DB2Jcc -version
```

2.6.2 Step 2: Designate a User Account on the IBM DB2 Source Database

Designate an IBM DB2 user account to be used for the `AVDB2DB` utility, which you will use later to configure collectors for your DB2 database. This user must have privileges to run the IBM DB2 `SYSPROC.ENV_GET_PROD_INFO` procedure.

Note: If you are using IBM DB2 Version 8.2, ensure that you have installed Fixpack 16. Otherwise, the `SYSPROC.ENV_GET_PROD_INFO` procedure is not available.

2.6.3 Step 3: Verify That the Source Database Is Compatible with the Collector

Now you are ready to verify that the IBM DB2 source database is compatible with the collector type in the Audit Vault collection agent home:

To verify the IBM DB2 source database compatibility:

1. Access either the shell used for the Audit Vault Server or the collection agent.

   If you want to use the collection agent location, and if you installed the collection agent on Microsoft Windows, go to the *`ORACLE_HOME`*`\`*`agent_dir`*`\bin` directory.

2. Run the `avdb2db verify` command.

   You must specify the host name and port number. Typically, for IBM DB2, the host is the fully qualified domain name or IP address of the server on which the IBM DB2 source database is running, and the port number is 50000.

   For example, assume that the host is `hrdb.example.com`, the port number is 50000, the source database is `sales_db`, and the user account is `srcuser_db2`:

```
$ avdb2db verify -src hrdb.example.com:50000:sales_db
Enter a username : srcuser_db2
Enter a password : password

***** Source Verified *****
```

   See Section 11.10 for detailed information about the `avdb2db verify` command.

3. Do not close this shell.

2.6.4 Step 4: Register the IBM DB2 Source Database with Oracle Audit Vault

To register the IBM DB2 source database with Oracle Audit Vault:

1. Access the shell used for the Audit Vault Server.
2. Run the `avdb2db add_source` command.

For example:

```
$ avdb2db add_source -src hrdb.example.com:50000 -srcname db2db4
Enter a username : srcuser_db2
Enter a password : password

***** Source Verified *****
***** Source Added Successfully *****
```

In this example:

- `-src`: Enter the fully qualified domain name (or IP address) and port number for the source database that you verified in Step 3: Verify That the Source Database Is Compatible with the Collector.
- `-srcname`: Create a name for this source database. Oracle Audit Vault refers to this name when it collects audit data.
- `username` and `password`: Enter the user name and password that you designated in Step 2: Designate a User Account on the IBM DB2 Source Database.

See Section 11.3 for detailed information about the `avdb2db add_source` command.

3. Do not close this shell.

2.6.5 Step 5: Add the DB2DB Collector to Oracle Audit Vault

To add the DB2DB collector to Oracle Audit Vault:

1. Access the shell used for the Audit Vault Server.
2. Run the `avdb2db add_collector` command.

   For example:

   ```
   $ avdb2db add_collector -srcname db2db4 -agentname agent1
   Enter a username :srcuser_db2
   Enter a password : password

   ***** Collector Added Successfully*****
   ```

   In this example:

   - `-srcname`: Create a name for the source database. Oracle Audit Vault refers to this name when collecting audit data.
   - `-agentname`: Create a name for the agent.
   - `username` and `password`: Enter the user name and password that you designated in Step 2: Designate a User Account on the IBM DB2 Source Database.

   See Section 11.2 for detailed information about the `avdb2db add_collector` command.

3. Modify the `SINGLE_FILEPATH` attribute of the `avdb2db alter_collector` command to point to the location of the DB2 audit directory. This is the directory where the DB2 collector will collect audit data. You must specify an absolute path, not a relative path.

   For example:

   ```
   $ avdb2db alter_collector -srcname db2db4 -collname DB2Collector
   ```

```
SINGLE_FILEPATH=DB2_HOME/sqlib/security/auditdata

***** Collector Altered Successfully *****
```

See Section 11.4 for more information about the `avdb2db alter_collector` command.

4. Do not close this shell.

2.6.6 Step 6: Convert the Binary DB2 Audit File to an ASCII Text File

IBM DB2 creates its audit files in a binary file format that is separate from the DB2 database. You must convert the binary file to an ASCII file before each time that Oracle Audit Vault collects audit data from a DB2 database. Ideally, schedule the script to run periodically. If the script finds older text files that have already been collected by the DB2DB collector, the script deletes them. It creates a new, timestamped ASCII text file each time you run it.

- Step 7A: Complete the Preparation Steps
- Step 7B: Run the Conversion Script

2.6.6.1 Step 7A: Complete the Preparation Steps

Follow these steps:

1. Identify a user who has privileges to run the `db2audit` command.

   This user will extract the binary files to the trace files.

2. Access the shell used by the Oracle Audit Vault collection agent.
3. Log in as the Oracle Audit Vault agent software owner.
4. Grant the user you identified in Step 1 execute privileges to run the conversion script from the Oracle Audit Vault directory.

   Alternatively, you can copy the appropriate conversion script located in the `$ORACLE_HOME/bin` directory to a location where this user can run them. These scripts are as follows:

   - **DB2 release 8.2 databases:** `DB282ExtractionUtil` (for Microsoft Windows, this file is called `DB282ExtractionUtil.bat`.)
   - **DB2 9.5 release databases:** DB295ExtractionUtil (for Microsoft Windows, this file is called `DB295ExtractionUtil.bat`.)

5. Grant the user you identified in Step 1 read permission for the `$ORACLE_HOME/av/log` directory and its contents.

   This user needs read permission for this directory as part of the process of generating the trace files that are extracted by the extraction utility.

2.6.6.2 Step 7B: Run the Conversion Script

Follow these steps:

1. In the server where you installed the IBM DB2 database, open a shell as the `SYSADM` DB2 user.
2. Set the following variables:

   - `ORACLE_HOME`

- DB2AUDIT_HOME (this directory points to the main directory that contains the db2audit command)

3. Ensure that the Oracle Audit Vault owner of the agent process has read permissions for the trace files that will be generated by the extraction utility.
4. Log in as the DB2 user that you identified in Step 1 in Section 2.6.6.1.
5. Make a note of the directory that you identified in Step 3 in Section 2.6.5.

   You will need to provide this directory path when you run the conversion script.
6. Run one of the following scripts, depending on the version of DB2 that you have installed:

   - **DB2 release 8.2 databases:** Run the script as follows:

     ```
     DB282ExtractionUtil default_DB2_audit_directory
     ```

     Enter the full directory path to the location of the DB2 audit directory. Typically, this directory is in the following locations:

     – **UNIX:** *DB2_HOME*/sqlib/security/auditdata

     – **Microsoft Windows:** *DB2HOME*\instance\security\auditdata

     Ensure that this path is the same as the path that you specfied for the avdb2db alter_collector SINGLE_FILEPATH attribute in Step 3 in Section 2.6.5.

     This script creates the ASCII text file in the auditdata directory, using the following format, which indicates the time the file was created:

     ```
     db2audit.instance.log.0.YYYYDDMMHHMMSS.out
     ```

   - **DB2 release 9.5 databases:** Run the script as follows:

     ```
     DB295ExtractionUtil default_DB2_audit_directory output_directory
     ```

     In this specification:

     – *default_DB2_audit_directory* is the same as the directory that is used for DB2 release 8.2.

     – *output_directory* is a directory specified by the avdb2db alter_collector SINGLE_FILEPATH attribute. See Section 11–2 in Section 11.4 for more information. This file is created in using the db2audit.instance.log.0.YYYYDDMMHHMMSS.out format.

     These two directory paths can be the same, or optionally, you can specify different directories for each location.

To schedule the script to run automatically, follow these guidelines:

- **Microsoft Windows.** Use the Windows Scheduler. Provide the archive directory path, extraction path (for release 9.5 databases only), and source database name in the scheduled task.
- **Linux.** Use the crontab UNIX utility. Provide the same information that you would provide using the parameters described previously when you normally run the script.

This step completes the registration for the IBM DB2 source database and its collector. Next, you must start the collection agent and collector. See Section 2.7 and Section 2.8 for more information.

2.7 Starting the Collection Agents

This section contains:

- Starting the Collection Agents from the Audit Vault Console
- Starting the Collection Agents from a Shell

2.7.1 Starting the Collection Agents from the Audit Vault Console

To start the collection agents from the Audit Vault Console:

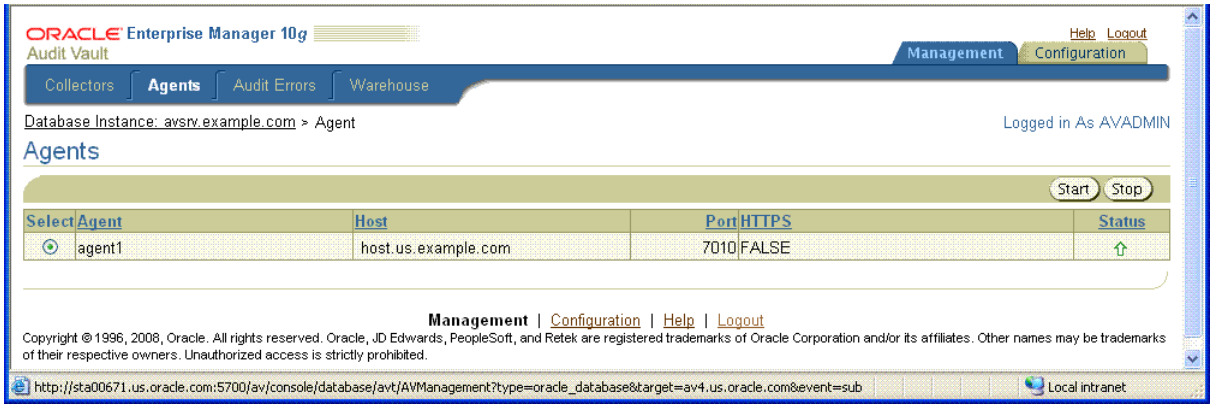
1. Start the Audit Vault Console.

   See Section 3.2.3. You must ensure that OC4J and the Audit Vault Console are running, and then you must log in to the Audit Vault Console.

2. In the Audit Vault Console, select the **Management** tab, and then select the **Agents** subpage.

   The Agents page appears with a table containing the following columns.

   - **Agent**: Name of the collection agent
   - **Host**: The host name where the collection agent is installed
   - **Port**: The port number of the host system where the collection agent is installed
   - **HTTPS**: Whether the collection agent is communicating with the Audit Vault Server using a secure communication channel (HTTPS)
   - **Status**: The current running status of the collection agent: a green up arrow indicates that the collection agent is running; a red down arrow indicates that the collection agent is not running, or error indicates that the collection agent is in an error state



3. Select the agent that you want to start, and then click **Start**.

2.7.2 Starting the Collection Agents from a Shell

To start the collection agents from a shell:

1. Start the Audit Vault Console.

   See Section 3.2.3. You must ensure that OC4J and the Audit Vault Console are running, but do not log in to the Audit Vault Console.

2. Access the shell used for the Audit Vault Server.

If you have closed this shell, reset its environment variables. See Section 2.2.2.

3. Run the `avctl show_agent_status` command to ensure that the collection agent is started.

   For example:

```
$ avctl show_agent_status -agentname agent1

AVCTL started
Getting agent metrics...
--------------------------------
Agent is not running
--------------------------------
Metrics retrieved successfully
--------------------------------
```

4. If the collection agent is not started, run the `avctl start_agent` command.

   For example:

```
$ avctl start_agent -agentname agent1

AVCTL started
Executing task start_agent
Starting Agent...
Agent started successfully.
```

2.8 Starting the Collectors

This section contains:

- Starting the Collectors from the Audit Vault Console
- Starting the Collectors from the Audit Vault Server or Collection Agent Shell

2.8.1 Starting the Collectors from the Audit Vault Console

To start the collectors from the Audit Vault Console:

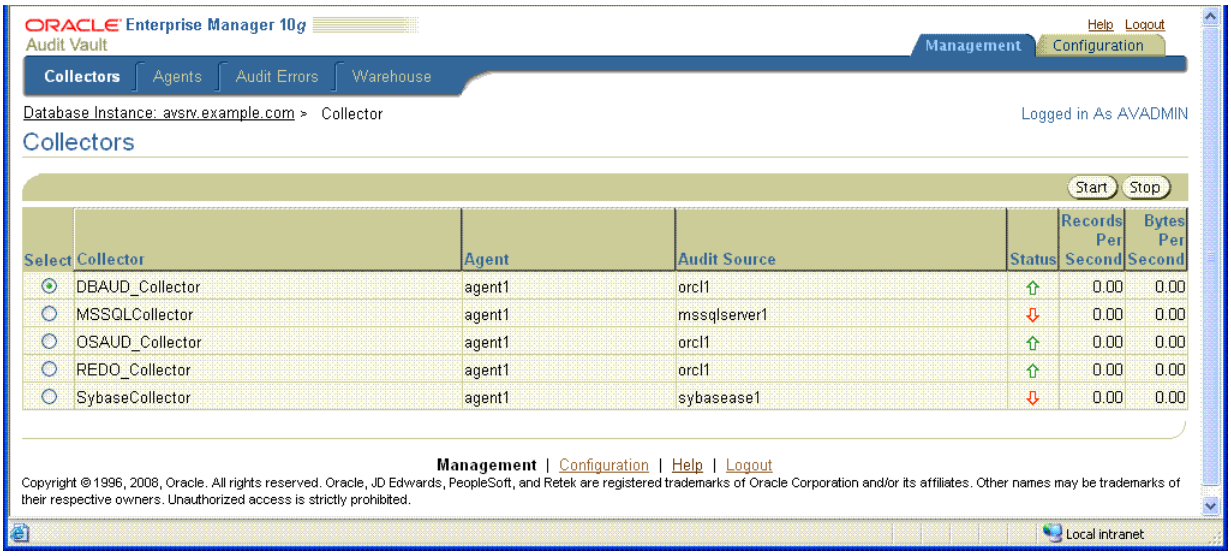
1. Log in to the Audit Vault Console as a user who has been granted the `AV_ADMIN` role.

   See Section 3.2.3 for login instructions.

2. Click the **Management** tab, then **Collectors** to display the **Collectors** page.

   The Collectors page appears with a table containing the following columns.

   - **Collector**: Name of the collector
   - **Agent**: The name of the collection agent for this collector
   - **Audit Source**: The name of the audit data source
   - **Status**: The current running status of the collector: a green up arrow indicates that the collector is running, a red down arrow indicates that the collector is not running, an error indicates that the collector is in an error state
   - **Records Per Second**: The number of records per second being collected for the current time period
   - **Bytes Per Second**: The number of bytes per second in audit records being collected for the current time period



3. Select the collector that you want to start.

   This page also indicates whether the collector is running. A green up arrow indicates the collector is running; a red down arrow indicates it is not running.

4. Click **Start**.

2.8.2 Starting the Collectors from the Audit Vault Server or Collection Agent Shell

To start the collectors from a shell:

1. Access the shell used for the Audit Vault collection agent.

   If you have closed this shell, see the following sections:

   - Section 2.2.3 describes how to set environment variables for the collection agent.
   - If you installed the collection agent on Microsoft Windows, do not set any environment variables. Instead, go to the *ORACLE_HOME**agent_dir*\bin directory.

2. Ensure that the agent OC4J is running.

   ```
   $ avctl show_oc4j_status
   ```

3. If the agent OC4J is not running, run the `avctl start_oc4j` command.

   ```
   $ avctl start_oc4j
   ```

4. Access the shell used for the Audit Vault Server.

   Section 2.2.2 describes how to set environment variables for the Audit Vault Server.

5. Run the `avctl start_collector` command.

   For example:

   ```
   $ avctl start_collector -collname OSAUD_Collector
                           -srcname ORCLSRC1.EXAMPLE.COM
   AVCTL started
   Executing task start_collector
   Starting Collector...
   ```

```
Collector started successfully.
```

If the startup is successful, Oracle Audit Vault moves the collector to a RUNNING state.

See Section 7.11 for more information about the avctl start_collector command.

2.9 Checking the Status of the Collectors

This section contains:

- Checking the Status of Collectors from the Audit Vault Console
- Checking the Status of Collectors from a Shell

2.9.1 Checking the Status of Collectors from the Audit Vault Console

1. Log in to the Audit Vault Console as a user who has been granted the AV_ADMIN role.

   See Section 3.2.3 for login instructions.

2. Select the **Management** tab, and then select the **Collectors** tab.

3. In the Collectors page, check the list of collectors.

   If the collector is running, its Status is set to an up arrow. If it is not, it is set to a red arrow pointing downward.

2.9.2 Checking the Status of Collectors from a Shell

To check the status of collectors from a shell:

1. Access the shell used for the Audit Vault Server.

   If you have closed this shell, open a new one and reset its environment variables. See Section 2.2.2.

2. Run the avctl show_collector_status command.

   For example:

```
$ avctl show_collector_status -collname OSAUD_Collector
                              -srcname ORCLSRC1.EXAMPLE.COM
AVCTL started
Getting collector metrics...
--------------------------------
Collector is running
Records per second  =  0.00
Bytes per second  =  0.00
--------------------------------
```

   See Section 7.7 for detailed information about the avctl show_collector_status command.

2.10 Checking If the Collectors Are Collecting Audit Records

To ensure that audit records are being collected, inspect the contents of the log files in the Audit Vault collection agent $ORACLE_HOME/av/log directory. The non-Oracle Database log files have the format *sourcedatabasename_collectorname*-%g.log. The %g is a generation number that starts from 0 (zero)

and increases once the file size reaches the 100 MB limit. The log file names for command-line utilities are as follows:

- **Oracle Database AVORCLDB utility:** *DBAUD-and-OSAUD-collector-name_source-name_source-id*.log
- **Microsoft SQL Server AVMSSQLDB utility**: MSSQLDB-%g.log
- **Sybase ASE AVSYBDB:** SYBDB-%g.log
- **IBM DB2 AVDB2DB utility:** AVDB2DB-%g.log

The log file keeps a running record of its audit record collection operations and will indicate when collection has occurred, or if a problem was encountered in the collection operation. See Appendix A for more information about the log files, and troubleshooting collector setup and collector startup operations.

3

Managing Oracle Audit Vault

This chapter contains:

- About Managing Oracle Audit Vault
- Managing the Audit Vault Server
- Altering Collector Properties and Attributes
- Managing the Oracle Audit Vault Data Warehouse
- Altering Source Database Attributes
- Removing Source Databases from Oracle Audit Vault

3.1 About Managing Oracle Audit Vault

This chapter describes common management activities that you need to perform after you have completed the configuration tasks in Chapter 2. You can use the Audit Vault Console or the command-line tools described in this chapter to manage Oracle Audit Vault.

3.2 Managing the Audit Vault Server

This section contains:

- About Managing the Audit Vault Console
- Checking the Audit Vault Console Status
- Starting the Audit Vault Console
- Stopping the Audit Vault Server Console
- Globally Disabling and Enabling Alert Settings
- Viewing Audit Event Categories
- Viewing Operational Errors That Oracle Audit Vault Catches

3.2.1 About Managing the Audit Vault Console

The Audit Vault Console is a graphical user interface that you can use to perform commonly used Oracle Audit Vault administration tasks. If you prefer to use a command-line interface, you can use equivalent commands in the `AVCA` and `AVCTL` utilities.

3.2.2 Checking the Audit Vault Console Status

To check the status of the Audit Vault Console:

1. Open a shell for the Audit Vault Server.
2. Follow the instructions in Section 2.2.2 to set the environment variables for the Audit Vault Server.
3. Run the following command:

```
$ avctl show_av_status
```

3.2.3 Starting the Audit Vault Console

To start the Audit Vault Console:

1. Access the shell used for the Oracle Audit Vault collection agent.

   If you have closed this shell, see the following sections:

   - Section 2.2.3 describes how to set environment variables for the collection agent.
   - If you installed the collection agent on Microsoft Windows, do not set any environment variables. Instead, go to the *ORACLE_HOME**agent_dir*\bin directory.

2. Ensure that the agent OC4J is running.

   Run the following AVCTL command in the Oracle Audit Vault Agent home (*ORACLE_HOME*/*agent_dir*/bin) to check its status.

```
$ avctl show_oc4j_status
```

3. If the agent OC4J is not running, run the `avctl start_oc4j` command.

```
$ avctl start_oc4j
```

4. Access the shell used for the Audit Vault Server.

   If you have closed this shell, reset its environment variables. See Section 2.2.2.

5. Ensure that the Audit Vault Console is running.

```
$ avctl show_av_status
```

   If the `avctl show_status` command indicates that the Audit Vault Console is not running, enter the following command:

```
$ avctl start_av
```

At this stage, you can log in to the Audit Vault Console.

1. From a Web browser, enter the following URL:

   http://*host*:*port*/av

   In this specification:

   - *host*: The host computer on which you installed the Audit Vault Server.
   - *port*: The port number reserved for the Audit Vault Server.

   If you are unsure of the host and port number values, then enter the `avctl show_av_status` command, which displays this information.

2. In the Login page, enter the following information:

- **User Name**: Enter the name of a user who has been granted the `AV_ADMIN` role.
- **Password**: Enter the user's password.
- **Connect As**: From the list, select **AV_ADMIN**.

3. Click **Login**.

3.2.4 Stopping the Audit Vault Server Console

To stop the Audit Vault Server console:

1. In a shell for the Audit Vault Server, set its environment variables.

   See Section 2.2.2 for more information.

2. Run the following command:

```
$ avctl stop_av
```

3.2.5 Globally Disabling and Enabling Alert Settings

If you need to perform maintenance tasks or other similar activities that do not require alert settings to be active, you can globally enable or disable the alert settings that Oracle Audit Vault auditors create. Do not disable alerts unless you are directed to do so by Oracle Support Services or encounter a problem with the alerts table. By default, alerts are enabled.

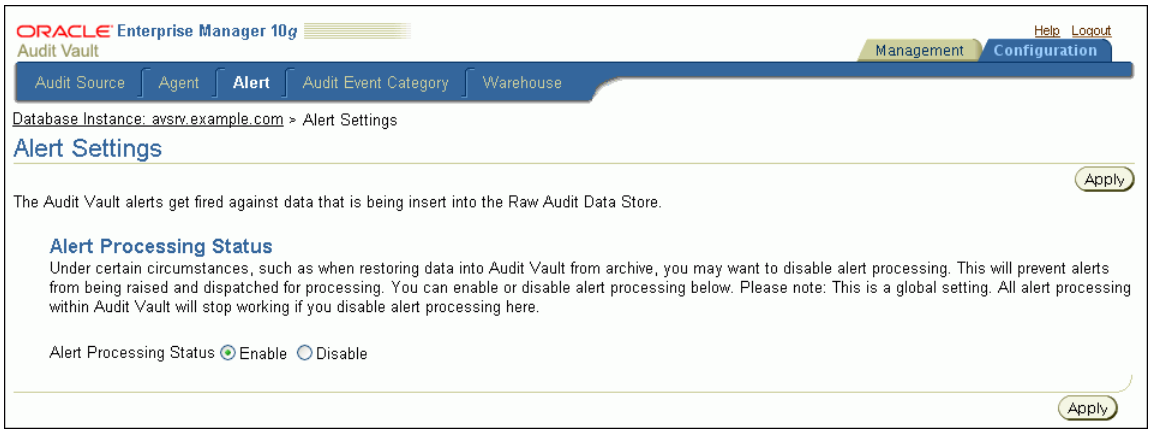
To globally disable and enable alerts:

1. Log in to the Audit Vault Console as a user who has been granted the `AV_ADMIN` role.

   See Section 3.2.3 for login instructions.

2. Select the **Configuration** tab, and then select the **Alert** subpage.

   The Alert Settings page appears.



3. At the Alert Processing Status label, select either **Disable** or **Enable**.
4. Click **Apply**.

3.2.6 Viewing Audit Event Categories

Audit event category management consists of viewing the Oracle Audit Vault audit event categories, their attributes, and their audited events. An audit event category

defines how various types of events are organized. For example, invalid records are placed in the Invalid Record event category. See *Oracle Audit Vault Auditor's Guide* for more information about audit event categories.

1. Log in to the Audit Vault Console as a user who has been granted the `AV_ADMIN` role.

   See Section 3.2.3 for login instructions.

2. Select the **Configuration** tab, and then select the **Audit Event Category** subpage.

   The Audit Event Category Management page appears.

3. Select an audit event category, and then click **View** to find detailed information about that category.

   The View Audit Event Category page appears.

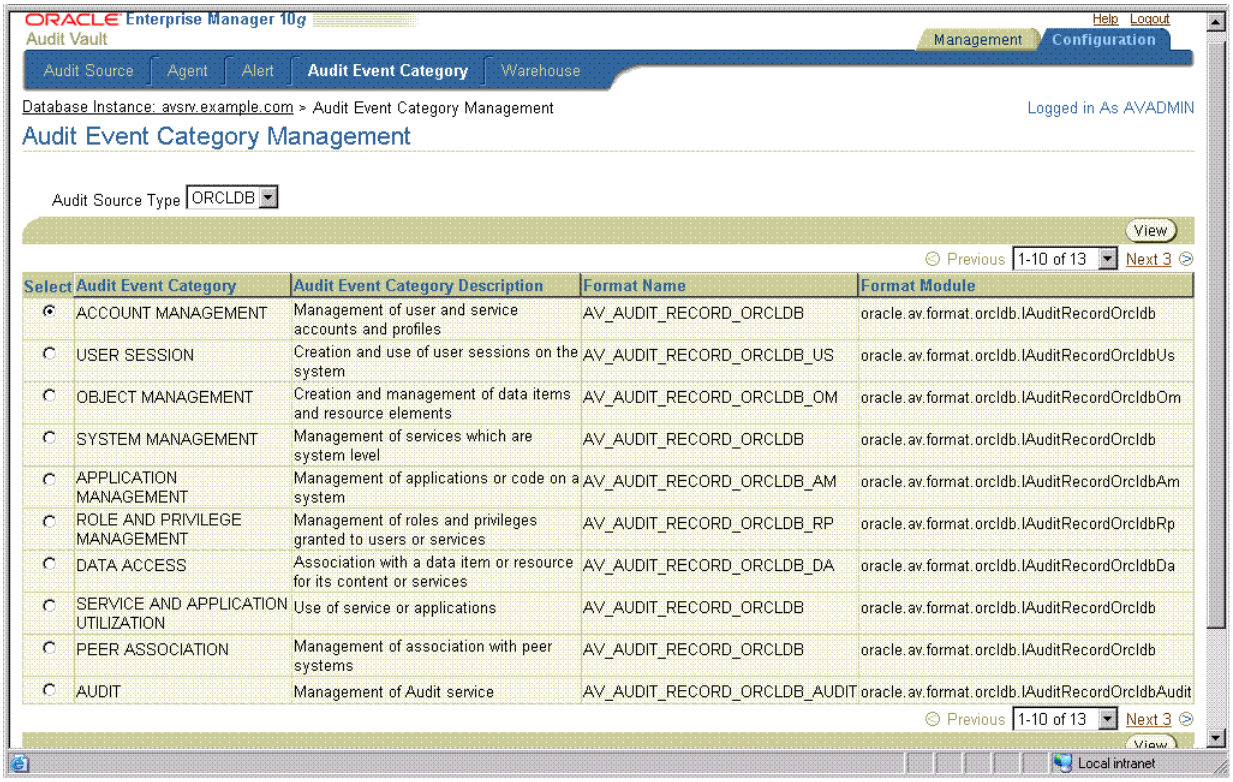
4. From the **Audit Source Type** list, select from the available source types: **ORCLDB**, **MSSQLDB**, **SYBDB**, and **DB2DB**.

5. Select the **Attributes** or **Audit Events** subpages to view detailed information about these categories.

6. Click **OK** when you complete viewing the audit event information for the category you selected.

Figure 3–1 shows the Audit Event Category Management page.

Figure 3–1 Audit Event Category Management Page



On the **Audit Event Category Management** page, audit event categories appear in a table with the following columns:

- Audit Event Category

- Audit Event Category Description
- Format Name
- Format Module

3.2.7 Viewing Operational Errors That Oracle Audit Vault Catches

You can use the Audit Vault Console to view operational errors that Oracle Audit Vault catches, such as broken database connections and missing files.

To view errors using Oracle Audit Vault:

1. Log in to the Audit Vault Console as a user who has been granted the `AV_ADMIN` role.

   See Section 3.2.3 for login instructions.

2. Select the **Management** tab, and then select the **Audit Errors** subpage.

   The Audit Errors page appears.

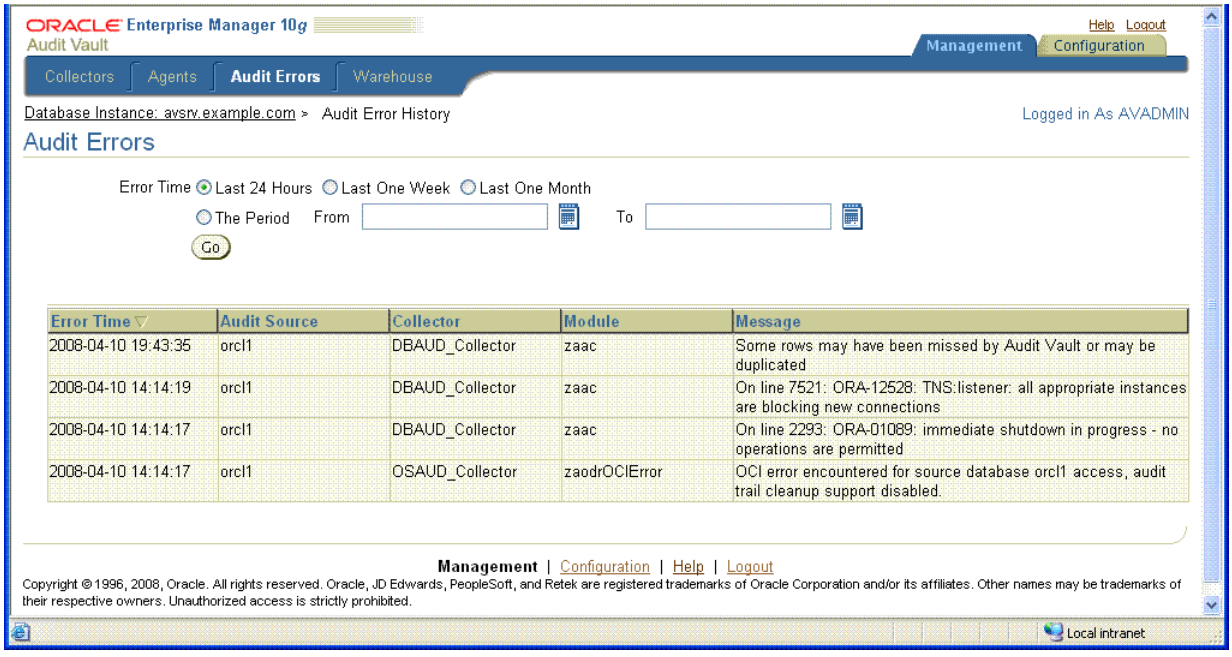
3. After the Error Time label, specify a time range of errors to view.

   Select from the **Last 24 Hours**, **Last One Week**, or **Last One Month** options to view errors from those times, or select **The Period** and then enter a start date in the **From** field and end date in the **To** field to specify a different time range.

4. Click **Go**.

Figure 3–2 shows the Audit Errors page with audit errors from the last 24 hours.

Figure 3–2 Audit Errors Page



The **Audit Errors** page displays error information as a table with the following column headings:

- **Error Time**: Local time when the audit error was generated
- **Audit Source**: The audit source database on which the audit error originated

- **Collector**: The collector on which the audit error originated
- **Module**: The module name involved in the audit error
- **Message**: The content of the audit error message

3.3 Altering Collector Properties and Attributes

This section contains:

- About Collector Properties and Attributes
- Altering Collector Properties and Attributes Using the Audit Vault Console
- Altering Collector Properties and Attributes Using a Shell

3.3.1 About Collector Properties and Attributes

After you add a collector to a database source, Oracle Audit Vault creates the collector with a set of default properties that are internal to Oracle Audit Vault. They have no effect on the source database. These properties control aspects such as the frequency of audit data collection from the source database, the name of the source database, and so on.

3.3.2 Altering Collector Properties and Attributes Using the Audit Vault Console

To alter collector properties and attributes using the Audit Vault Console:

1. Log in to the Audit Vault Console as a user who has been granted the `AV_ADMIN` role.

   See Section 3.2.3 for login instructions.

2. Select the **Configuration** tab, and then select the **Audit Source** subpage.

   The Source Configuration Management page appears.

3. Select the **Collector** subpage.

   The Collector Configuration Management page appears, which displays the current settings for the available collectors.

4. Select the collector that you want to modify, and then click the **Edit** button.

   The Edit Collector page appears.

5. Under Attributes, modify the attributes for the collectors by editing the values in the Value column.

   For more information about these attributes, see the following sections:

   - Section 8.4 for the Oracle Database collector attributes
   - Section 9.4 for the SQL Server collector attributes
   - Section 10.4 for the Sybase ASE collector attributes
   - Section 11.4 for the IBM DB2 collector attributes

6. Click **OK**.

3.3.3 Altering Collector Properties and Attributes Using a Shell

To alter collector properties using a shell:

1. In a shell for the Audit Vault Server, ensure that you have set its environment variables.

   See Section 2.2.2 for more information.

2. Run the `alter_collector` command for each collector type, as shown in the following examples:

   For Oracle Database:

   ```
   $ avorcldb alter_collector -srcname hrdb.example.com -collname DBAUD_Collector
   AUDAUDIT_DELAY_TIME=60
   ```

   See Section 8.4 for more information about the `avorcldb alter_collector` command.

   For Microsoft SQL Server:

   ```
   $ avmssqldb alter_collector -srcname mssqldb4 -collname MSSQLCollector NO_OF_
   RECORDS=1500 DESCRIPTION="MSSQLDB collector 45" SERVERSIDE_
   FILEPATH="c:\SQLAuditFile*
   ```

   See Section 9.4 for more information about the `avmssqldb alter_collector` command.

   For Sybase ASE:

   ```
   $ avsybdb alter_collector -srcname sybdb4 -collname SybaseCollector
   NO_OF_RECORDS=1500 DESCRIPTION="Sybase collector 45"
   ```

   See Section 10.4 for more information about the `avsybdb alter_collector` command.

   For IBM DB2:

   ```
   $ avdb2db alter_collector -srcname db2db4 -collname DB2Collector
   NO_OF_RECORDS=1500 DESCRIPTION="IBM DB2 collector 95"
   ```

   See Section 11.4 for more information about the `avdb2db alter_collector` command.

3.4 Managing the Oracle Audit Vault Data Warehouse

This section contains:

- About Managing the Oracle Audit Vault Data Warehouse
- Setting the Audit Vault Data Warehouse Refresh Schedule and Retention Period
- Manually Refreshing Audit Vault Data Warehouse Audit Data
- Loading Data to the Oracle Audit Vault Data Warehouse
- Purging Data from the Oracle Audit Vault Data Warehouse

3.4.1 About Managing the Oracle Audit Vault Data Warehouse

The collectors collect audit data from their source database and send it to the Oracle Audit Vault repository. The repository stores the data in an internal format. The repository also contains a data warehouse. A database job periodically refreshes the data warehouse with the latest audit records. Oracle Audit Vault provides predefined reports that display the data in the warehouse to the auditor.

You can perform the following activities with the Oracle Audit Vault data warehouse:

- **Set the Audit Vault data warehouse refresh schedule.** This schedule determines how frequently the data warehouse is refreshed with current data collected by the collectors.
- **Set a retention period for the data that has been refreshed.** The data warehouse then contains the most recent data for that length of time after each refresh.
- **Load older data from the raw audit data store into the data warehouse tables.** You can load older data into the data warehouse so that it can be available for analysis in the Oracle Audit Vault reports. However, you cannot load data from outside sources—just data that has been previously collected by the collectors but is too old to be loaded into the data warehouse as part of a normal refresh.
- **Purge audit data.** If you load older audit data into the warehouse, you can purge it from the data warehouse. Oracle Audit Vault still maintains this data in the Audit Vault repository but does not make is available for analysis in the warehouse.

3.4.2 Setting the Audit Vault Data Warehouse Refresh Schedule and Retention Period

This section contains:

- About Setting the Refresh Schedule and Retention Period
- Scheduling the Audit Data Refresh Settings Using the Audit Vault Console
- Scheduling the Audit Data Refresh Settings Using a Shell

3.4.2.1 About Setting the Refresh Schedule and Retention Period

The refresh schedule moves data from the **raw audit data store** (that is, the internal format) into the data warehouse, so that it can be made available for the Oracle Audit Vault reports. The data warehouse is implemented as a sliding window over the audit data that has been collected. Each refresh of the data warehouse moves this window forward in time so that it always contains the latest audit records. The size of the window specifies how far back in time the window extends.

By default, Oracle Audit Vault refreshes the data warehouse once every 24 hours. You can set a retention period that determines the size of a sliding window of time for the data warehouse to hold this audit data.

The refresh schedule and retention period work together as follows: Suppose you have configured two source databases with Oracle Audit Vault. One database has 4 years of audit data accumulated and the other has 3 years of audit data. You want to retain only *exactly* the last year of data after each refresh. To accomplish this, you must do the following:

1. Schedule the refresh to start on a given day. For example, assuming that today is August 8, 2008, you set it for today.
2. Specify a frequency of once a day for the refresh to occur.
3. Set the retention period to 1 year. This retention period refers to the year before and leading up to the date that you specified in Step 1.

When the first refresh occurs, Oracle Audit Vault loads into the data warehouse the audit data that began 1 year ago, starting on August 8, 2007, to the current date, August 8, 2008. When the next refresh occurs on August 9, 2008, only the new audit data is retrieved. The retention period shifts forward: now this period is from August 9, 2007, to August 9, 2008. Oracle Audit Vault then discards the audit data from August 8, 2007, because now it is older than the retention period. This way, you always have the most recent year of audit data, right up to the current date.

There are two ways that you can create a refresh schedule:

- **Create the schedule once, directly in Oracle Audit Vault.** The schedule settings remain in place until the next time you modify these settings.
- **Create one or more predefined schedules by using the DBMS_SCHEDULER PL/SQL package.** You can create this schedule in SQL*Plus (or another SQL tool such as SQL Developer). Afterward, you use Oracle Audit Vault to select the schedule that you want to use. For more information about the `DBMS_SCHEDULER` package, see *Oracle Database PL/SQL Packages and Types Reference*.

You can create a schedule and retention period from either the Audit Vault Console or at a shell by using the `AVCA` utility.

3.4.2.2 Scheduling the Audit Data Refresh Settings Using the Audit Vault Console

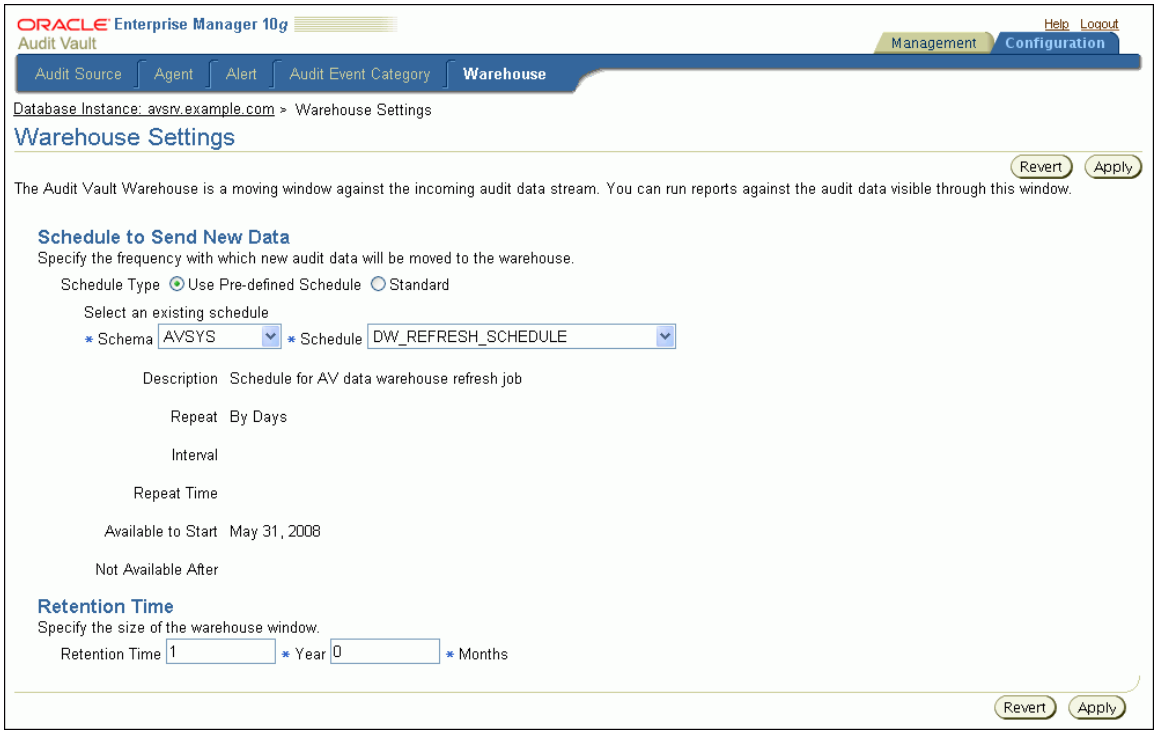
To create the refresh schedule and retention period using the Audit Vault Console:

1. Log in to the Audit Vault Console as a user who has been granted the `AV_ADMIN` role.

   See Section 3.2.3 for login instructions.

2. Select the **Management** tab, and then select the **Configuration** subpage.

   The Warehouse Settings page appears.



3. Either select an existing schedule or create a new one.

   To select an existing schedule:

   a. Under Schedule to Send New Data, select **Use Pre-defined Schedule**.

   b. From the **Schema** list, select the name of the schema in which the schedule was created.

   c. From the **Schedule** list, select the name of the schedule.

Information about the schedule appears: a brief description, repeat times (frequency of to repeating the schedule), interval, repeat time (the time to repeat the schedule), and start and end dates. If settings have been omitted (for example, an interval time), then these labels are blank.

To create a new, standard schedule:

a. Under Schedule to Send New Data, select **Standard**.

b. Enter the following information:

Frequency Type: From the list, select a frequency type, such as **By Hours**.

Interval (*frequency type*): Enter the frequency for the type of frequency that you selected. For example, `1` for once every hour.

Start Date: Specify the date on which the refresh occurs. If you select a date that is earlier than today's date, then the refresh today.

Start Time: Enter the time at which the refresh occurs.

4. Set the retention window, that is, the period of time during which the data sent to the Oracle Audit Vault data warehouse remains in storage.

For example, suppose that you scheduled Oracle Audit Vault to refresh the raw audit data store every 2 hours, starting on August 19, 2008 at 2 a.m., and you want to keep this data in storage for the next year and a half. To do so, you would enter `1` in the **Year** field and `6` in the **Months** field.

5. Click **Apply**.

3.4.2.3 Scheduling the Audit Data Refresh Settings Using a Shell

To create the refresh schedule and retention period using a shell:

1. In a shell for the Audit Vault Server, ensure that you have set its environment variables.

See Section 2.2.2 for more information.

2. Run the `avca set_warehouse_schedule` command to either specify an existing schedule or to create a new one.

For example, to select an existing schedule named `daily_refresh`:

```
$ avca set_warehouse_schedule -schedulename 'daily_refresh'
```

To create a new schedule:

```
$ avca set_warehouse_schedule -startdate 01-JUL-06 -rptintrv
'FREQ=DAILY;BYHOUR=0'
```

In this example:

- `startdate` specifies the date for the first refresh to begin.
- `rptintrv` specifies the intervals for the refreshes, in this case, once a day.

See Section 6.14 for more information about the `avca set_warehouse_schedule` command.

3. Run the `avca set_warehouse_retention` command to set the retention period.

For example, to specify a period of 1 year and 6 months, enter the following command:

```
$ avca set_warehouse_retention -intrv +01-06
```

See Section 6.13 for more information about the `avca set_warehouse_retention` command.

3.4.3 Manually Refreshing Audit Vault Data Warehouse Audit Data

This section contains:

- About Manually Refreshing the Data Warehouse Data
- Manually Refreshing the Data Warehouse Using the Audit Vault Console
- Manually Refreshing the Data Warehouse Using a Shell

3.4.3.1 About Manually Refreshing the Data Warehouse Data

You can refresh the Oracle Audit Vault data warehouse repository with data from the raw audit data store. As with a scheduled refresh, Oracle Audit Vault collects the raw audit data from its source databases and places it into the Audit Vault data warehouse.

3.4.3.2 Manually Refreshing the Data Warehouse Using the Audit Vault Console

When you manually refresh the data in the Oracle Audit Vault data warehouse, you also can check the history of when refresh operations occurred.

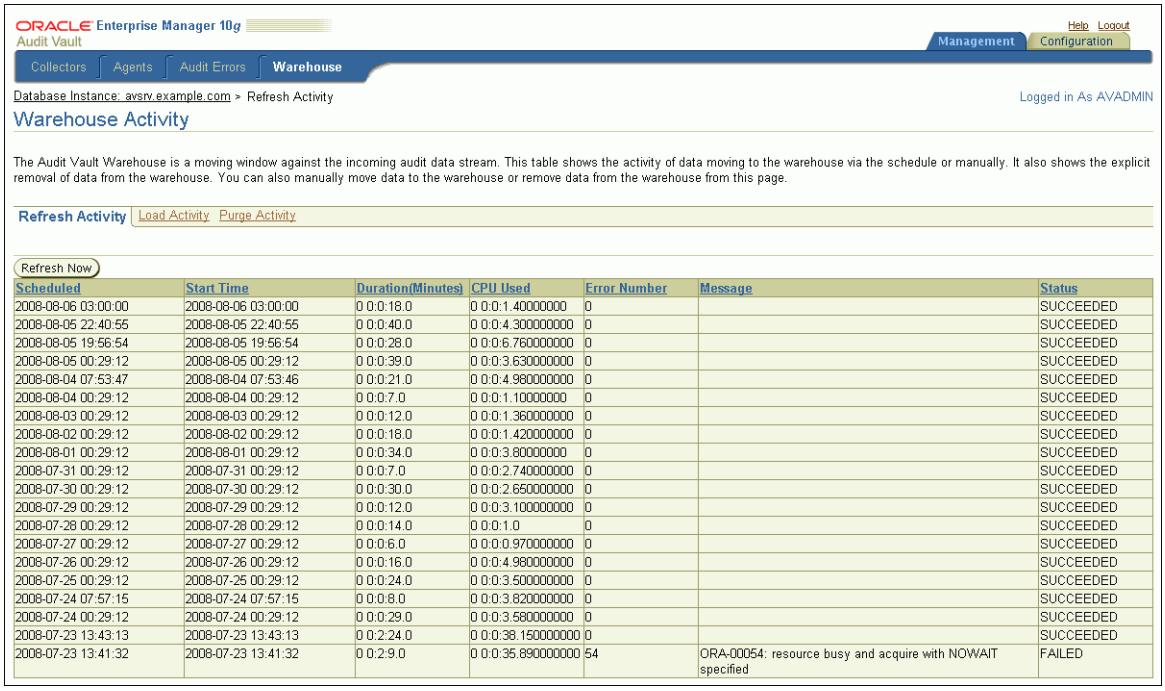
To manually refresh the data warehouse using the Audit Vault Console:

1. Log in to the Audit Vault Console as a user who has been granted the `AV_ADMIN` role.

   See Section 3.2.3 for login instructions.

2. Select the **Management** tab, and then select the **Warehouse** subpage.

   The Warehouse Activity page appears.



Scheduled	Start Time	Duration(Minutes)	CPU Used	Error Number	Message	Status
2008-08-06 03:00:00	2008-08-06 03:00:00	0 0:0:18.0	0 0:0:1.40000000	0		SUCCEEDED
2008-08-05 22:40:55	2008-08-05 22:40:55	0 0:0:40.0	0 0:0:4.300000000	0		SUCCEEDED
2008-08-05 19:56:54	2008-08-05 19:56:54	0 0:0:28.0	0 0:0:6.760000000	0		SUCCEEDED
2008-08-05 00:29:12	2008-08-05 00:29:12	0 0:0:39.0	0 0:0:3.630000000	0		SUCCEEDED
2008-08-04 07:53:47	2008-08-04 07:53:46	0 0:0:21.0	0 0:0:4.980000000	0		SUCCEEDED
2008-08-04 00:29:12	2008-08-04 00:29:12	0 0:0:7.0	0 0:0:1.10000000	0		SUCCEEDED
2008-08-03 00:29:12	2008-08-03 00:29:12	0 0:0:12.0	0 0:0:1.360000000	0		SUCCEEDED
2008-08-02 00:29:12	2008-08-02 00:29:12	0 0:0:18.0	0 0:0:1.420000000	0		SUCCEEDED
2008-08-01 00:29:12	2008-08-01 00:29:12	0 0:0:34.0	0 0:0:3.80000000	0		SUCCEEDED
2008-07-31 00:29:12	2008-07-31 00:29:12	0 0:0:7.0	0 0:0:2.740000000	0		SUCCEEDED
2008-07-30 00:29:12	2008-07-30 00:29:12	0 0:0:30.0	0 0:0:2.650000000	0		SUCCEEDED
2008-07-29 00:29:12	2008-07-29 00:29:12	0 0:0:12.0	0 0:0:3.100000000	0		SUCCEEDED
2008-07-28 00:29:12	2008-07-28 00:29:12	0 0:0:14.0	0 0:0:1.0	0		SUCCEEDED
2008-07-27 00:29:12	2008-07-27 00:29:12	0 0:0:6.0	0 0:0:0.970000000	0		SUCCEEDED
2008-07-26 00:29:12	2008-07-26 00:29:12	0 0:0:16.0	0 0:0:4.980000000	0		SUCCEEDED
2008-07-25 00:29:12	2008-07-25 00:29:12	0 0:0:24.0	0 0:0:3.500000000	0		SUCCEEDED
2008-07-24 07:57:15	2008-07-24 07:57:15	0 0:0:8.0	0 0:0:3.820000000	0		SUCCEEDED
2008-07-24 00:29:12	2008-07-24 00:29:12	0 0:0:29.0	0 0:0:3.580000000	0		SUCCEEDED
2008-07-23 13:43:13	2008-07-23 13:43:13	0 0:2:24.0	0 0:0:38.150000000	0		SUCCEEDED
2008-07-23 13:41:32	2008-07-23 13:41:32	0 0:2:9.0	0 0:0:35.890000000	54	ORA-00054: resource busy and acquire with NOWAIT specified	FAILED

The Warehouse Activity page shows the following information:

- **Scheduled**: The scheduled time to perform a refresh operation
- **Start Time**: The time when a refresh operation started
- **Duration (Minutes)**: The total time required to complete a refresh operation
- **CPU Used**: The amount of CPU time used to complete a refresh operation
- **Error Number**: The Oracle ORA- error number, if any, resulting from a refresh operation
- **Message**: The error messages, if any, resulting from a refresh operation
- **Status**: The current status of a refresh operation: FAILED or SUCCEEDED

3. Click the **Refresh Now** button.

3.4.3.3 Manually Refreshing the Data Warehouse Using a Shell

To manually refresh the data warehouse using a shell:

1. In a shell for the Audit Vault Server, set its environment variables.

   See Section 2.2.2 for more information.

2. Run the `avctl refresh_warehouse` command.

   For example:

```
$ avctl refresh_warehouse -wait

AVCTL started
Refreshing warehouse...
Waiting for refresh to complete...
done.
```

   The `-wait` parameter delays refreshing the raw data store until the current refresh job (if one is occurring) completes. See Section 7.4 for more information about the `avctl refresh_warehouse` command.

3.4.4 Loading Data to the Oracle Audit Vault Data Warehouse

This section contains:

- About Loading Data into the Oracle Audit Vault Warehouse
- Loading Data Warehouse Data Using the Audit Vault Console
- Loading Data Warehouse Data Using a Shell

3.4.4.1 About Loading Data into the Oracle Audit Vault Warehouse

You can load data that is older than the retention period from the **raw audit data store** into the Oracle Audit Vault data warehouse tables. After you load this data, it is available to auditors to generate reports or perform analysis.

To find the current retention period setting, view the Warehouse Settings page of the Audit Vault Console (see Section 3.4.2); to find the last time the data was refreshed, view the Warehouse Activity page (Section 3.4.3).

3.4.4.2 Loading Data Warehouse Data Using the Audit Vault Console

To load the data warehouse data using the Audit Vault Console:

1. Log in to the Audit Vault Console as a user who has been granted the `AV_ADMIN` role.

See Section 3.2.3 for login instructions.

2. Optionally, disable the alert settings.

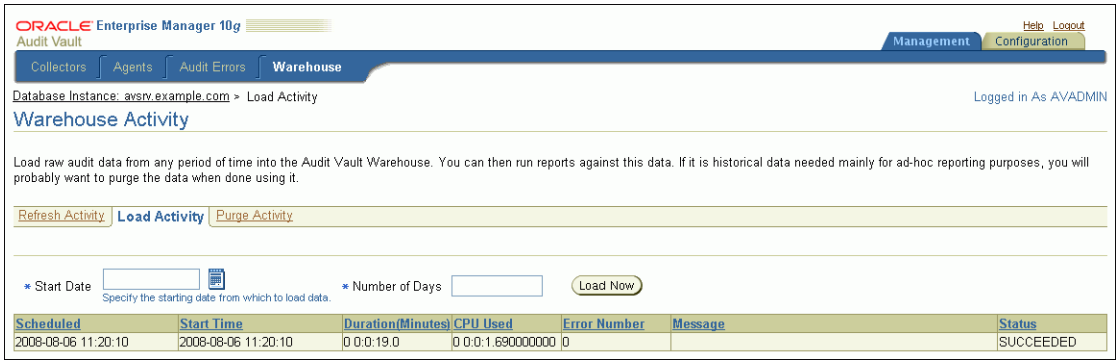
   See Section 3.2.5 for more information.

3. Select the **Management** tab, and then select the **Warehouse** subpage.

   The Warehouse Activity page appears.

4. Select the **Load Activity** subpage.

   The Load Activity page appears.



5. In the **Start Date** field, enter the beginning date of the data that you want to load. For example, suppose the source database contains audit data that is 10 years old, and you want to load the last 5 years worth of audit data into the Oracle Audit Vault data warehouse. Assuming that today's date is August 8, 2008, you would specify August 8, 2003 as the start date.

6. In the **Number of Days** field, enter the number of days, starting from the start date, through which you want to load data.

7. Click the **Load Now** button.

   Oracle Audit Vault schedules the data load operation, which is listed on this page the next time you access it.

8. Reenable the alert settings if you had disabled them.

   See Section 3.2.5 for more information.

3.4.4.3 Loading Data Warehouse Data Using a Shell

To load the data warehouse data using a shell:

1. Optionally, disable the alert settings.

   See Section 3.2.5 for more information.

2. In a shell for the Audit Vault Server, ensure that you have set its environment variables.

   See Section 2.2.2 for more information.

3. Run the `avctl load_warehouse` command.

   For example, to load 10 days of audit data that was recorded starting on August 8, 2003, enter the following command:

```
$ avctl load_warehouse -startdate 08-AUG-03 -numofdays 10
```

See Section 7.2 for more information about the `avctl load_warehouse` command.

4. Reenable the alert settings if you had disabled them.

   See Section 3.2.5 for more information.

3.4.5 Purging Data from the Oracle Audit Vault Data Warehouse

This section contains:

- About Purging the Oracle Audit Vault Data Warehouse
- Purging Data Warehouse Data Using the Audit Vault Console
- Purging Data Warehouse Data Using a Shell

3.4.5.1 About Purging the Oracle Audit Vault Data Warehouse

When you no longer need the audit data that you have loaded into Audit Vault Server, you can remove it from the Oracle Audit Vault data warehouse. If in the future you decide that you need to run reports against this purged data, follow the instructions in Section 3.4.4 to reload the necessary data into the data warehouse. You can only remove data that is older than the retention period. You can find and reset the retention period from the Audit Vault Console Warehouse Settings page (see Section 3.4.2).

3.4.5.2 Purging Data Warehouse Data Using the Audit Vault Console

To purge the data warehouse data using the Audit Vault Console:

1. Log in to the Audit Vault Console as a user who has been granted the `AV_ADMIN` role.

   See Section 3.2.3 for login instructions.

2. Select the **Management** tab, and then select the **Warehouse** subpage.

   The Warehouse Activity page appears.

3. Select the Purge Activity page.

   The Purge Activity subpage appears.

4. In the **Start Date** field, enter the beginning date of the data that you want to purge.

5. In the **Number of Days** field, enter the number of days, starting from the start date, through which you want to purge data.

6. Click the **Purge Now** button.

   Oracle Audit Vault schedules the data purge operation, which is listed on this page the next time you access it.

3.4.5.3 Purging Data Warehouse Data Using a Shell

To purge the data warehouse data using a shell:

1. In a shell for the Audit Vault Server, ensure that you have set its environment variables.

   See Section 2.2.2 for more information.

2. Run the `avctl purge_warehouse` command.

For example, to purge 10 days of audit data that was recorded starting on January 1, 2004, and to specify that the operation wait until the previous purge job completes, enter the following command:

```
$ avctl purge_warehouse -startdate 01-JAN-04 -numofdays 10 -wait
```

See Section 7.3 for more information about the `avctl purge_warehouse` command.

3.5 Altering Source Database Attributes

This section contains:

- About Source Database Attributes
- Altering Source Database Attributes Using the Audit Vault Console
- Altering Source Database Attributes Using a Shell

3.5.1 About Source Database Attributes

After you register a source database, Oracle Audit Vault creates a set of properties that reflect general aspects of the source database itself, such as its port number and IP address. These properties are internal to Oracle Audit Vault and have no effect on the source database.

3.5.2 Altering Source Database Attributes Using the Audit Vault Console

To alter the source database attributes using the Audit Vault Console:

1. Log in to the Audit Vault Console as a user who has been granted the `AV_ADMIN` role.

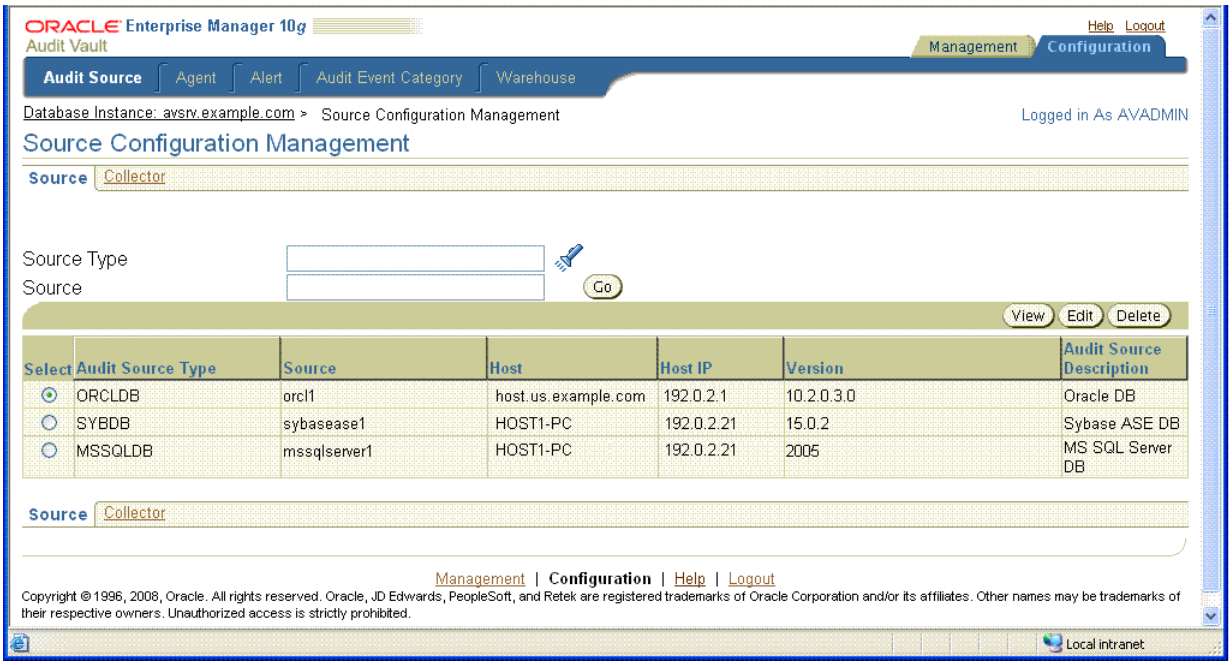
   See Section 3.2.3 for login instructions.

2. Select the **Configuration** tab, and then select the **Audit Source** subpage.

   The Source Configuration Management page appears.

3. Select the **Source** subpage.

   The Source Configuration Management page displays the current settings for the available collectors.



4. Select the source database that you want to modify, and then click the **Edit** button.

   The Edit Source page appears.

5. Under Properties, optionally modify the description of the source database.

6. Under Attributes, modify the attributes for the source database by editing the values in the **Value** column.

   For more information about these attributes, see the following sections:

   - Section 8.5 for the Oracle Database source database attributes
   - Section 9.5 for the SQL Server source database attributes
   - Section 10.5 for the Sybase ASE source database attributes
   - Section 11.5 for the IBM DB2 source database attributes

7. Click **OK**.

3.5.3 Altering Source Database Attributes Using a Shell

To alter source database attributes using a shell:

1. In a shell for the Audit Vault Server, ensure that you have set its environment variables.

   See Section 2.2.2 for more information.

2. Run the `alter_source` command for each source database type, as shown in the following examples.

   For Oracle Database:

   ```
   $ avorcldb alter_source -srcname hrdb.example.com PORT=1522
   ```

   See Section 8.5 for more information about the `avorcldb alter_source` command.

   For Microsoft SQL Server:

```
$ avmssqldb alter_source -srcname mssqldb4 DESCRIPTION="HR Database"
```

See Section 9.5 for more information about the `avmssqldb alter_source` command.

For Sybase ASE:

```
$ avsybdb alter_source -srcname sybdb4 DESCRIPTION="HR Database"
```

See Section 10.5 for more information about the `avsybdb alter_source` command.

For IBM DB2:

```
$ avdb2db alter_source -srcname db2db4 DESCRIPTION="HR Database"
```

See Section 11.5 for more information about the `avdb2db alter_source` command.

3.6 Removing Source Databases from Oracle Audit Vault

This section contains:

- About Removing Source Databases from Oracle Audit Vault
- Removing a Source Database Using the Audit Vault Console
- Removing a Source Database Using a Shell

3.6.1 About Removing Source Databases from Oracle Audit Vault

If you no longer need to have a source database registered with Oracle Audit Vault, you can use either the Audit Vault Console or the command-line utilities to remove the source database from Oracle Audit Vault. After you have removed the source database, its audit data still resides in the data warehouse within its retention period. To purge this audit data, see Section 3.4.5. You can check the length of the retention period in the Audit Vault Console; see Section 3.4.2.

Remember that after you have removed a source database, its identity data remains in Oracle Audit Vault so that there will be a record of source databases that have been dropped. Therefore, you cannot add a new source database with the name of a dropped source database. Remove the source database only if you no longer want to collect its data or if it has moved to a new host computer.

3.6.2 Removing a Source Database Using the Audit Vault Console

To remove a source database from Oracle Audit Vault using the Audit Vault Console:

1. Log in to the Audit Vault Console as a user who has been granted the `AV_ADMIN` role.

   See Section 3.2.3 for login instructions.

2. Select the **Configuration** tab, and then select the **Audit Source** subpage.

   The Source Configuration Management subpage appears.

3. From the list of source databases, select the database that you want to remove, and then click **Delete**.

   You can search for a source database by entering data in the **Source Type** and **Source** fields.

4. Click **Yes** in the Confirmation window.

3.6.3 Removing a Source Database Using a Shell

To remove a source database from Oracle Audit Vault using a shell:

1. In a shell for the Audit Vault Server, ensure that you have set its environment variables.

   See Section 2.2.2 for more information.

2. Run the `drop_source` command for the source database, as shown in the following examples:

   For Oracle Database:

   ```
   $ avorcldb drop_source -srcname orcldb.example.com
   ```

   See Section 8.7 for more information about the `avorcldb drop_source` command.

   For Microsoft SQL Server:

   ```
   $ avmssqldb drop_source -srcname mssqldb4
   ```

   See Section 9.7 for more information about the `avmssqldb drop_source` command.

   For Sybase ASE:

   ```
   $ avsybdb drop_source -srcname sybdb4
   ```

   See Section 10.7 for more information about the `avsybdb drop_source` command.

   For IBM DB2:

   ```
   $ avdb2db drop_source -srcname db2db4
   ```

   See Section 11.7 for more information about the `avdb2db drop_source` command.

4

Administering the Oracle Audit Vault Repository

This chapter contains:

- About the Administrative Tasks in This Chapter
- Monitoring the Audit Vault Server SYSAUX Tablespace Space Usage
- Monitoring Audit Vault Server Archive Log Disk Space Usage
- Monitoring the Audit Vault Server Flash Recovery Area
- Managing Oracle Audit Vault Backup and Recovery Operations
- Using a Collection Agent to Listen to Oracle Database RAC Nodes
- Configuring Collection Agent Connectivity for Oracle Database RAC
- Purging the Oracle Source Database Audit Trail Records

4.1 About the Administrative Tasks in This Chapter

This chapter describes important administrative tasks to perform on the Oracle Audit Vault system. These tasks are especially important if your audit data collectors are collecting high volumes of audit records and rapidly filling the default tablespace and disk space.

4.2 Monitoring the Audit Vault Server SYSAUX Tablespace Space Usage

The Oracle Audit Vault Server database contains the `SYSAUX` tablespace, which by default has one data file. The `SYSAUX` tablespace is a locally managed tablespace with automatic segment space management.

You should monitor the space usage for the `SYSAUX` tablespace and create additional data files for storage as needed. Remember that if you use the procedures in Section 4.8 to clean up the audit trail, the `SYSAUX` tablespace by default will store the audit trail.

See *Oracle Database Administrator's Guide* for more information about the `ALTER TABLESPACE` SQL statement, which you can use to add more storage data files. For information about optimizing a tablespace, see *Oracle Database Performance Tuning Guide*.

4.3 Monitoring Audit Vault Server Archive Log Disk Space Usage

By default, ARCHIVELOG mode is enabled in the Audit Vault Server database. The ARCHIVELOG mode copies filled online redo logs to disk. This enables you to back up the database while it is open and being accessed by users, and to recover the database to any desired point in time. You should monitor the disk space usage for the redo logs.

See *Oracle Database Administrator's Guide* for more information about changing the LOG_ARCHIVE_DEST_*n* location to relocate these archive log files to larger disks. For information about backing up the archive logs, see *Oracle Database Backup and Recovery Advanced User's Guide*.

4.4 Monitoring the Audit Vault Server Flash Recovery Area

By default, the Audit Vault Server database has the following initialization parameter settings:

- The DB_RECOVERY_FILE_DEST_SIZE initialization parameter is set to 2 GB.
- The DB_RECOVERY_FILE_DEST initialization parameter is set to the default flash recovery area, typically the *ORACLE_HOME/flash_recovery_area* directory.

Ensure that the size of the flash recovery area is large enough to hold a copy of all data files, all incremental backups, online redo logs, archived redo logs not yet backed up on tape, control files, and control file auto backups. This space can fill up quickly, depending on the number of collectors configured, the scope of the audit record collection being administered, and the backup and archive plans that you have in place.

You can use Oracle Enterprise Manager Database Control to monitor the available space in the flash recovery area. Monitor the percent space that is usable in the Usable Flash Recovery Area field under the High Availability section on the Home page. Check the alert log in the Database Console for messages. When the used space in the flash recovery area reaches 85 percent, a warning message is sent to the alert log. When the used space in the flash recovery area reaches 97 percent, a critical warning message is sent to the alert log.

You can manage space in the flash recovery area by adjusting the retention policy for data files to keep fewer copies or reduce the number of days these files stay in the recovery window. Alternatively, increase the value of the DB_RECOVERY_FILE_DEST_SIZE initialization parameter to accommodate these files and to set the DB_RECOVERY_FILE_DEST initialization parameter to a value where more disk space is available. See *Oracle Database Administrator's Guide* and *Oracle Database Backup and Recovery Basics* for more information.

4.5 Managing Oracle Audit Vault Backup and Recovery Operations

When you back up Oracle Audit Vault, you must back up the database, the Audit Vault Server home, and the Audit Vault collection agent home.

> **See Also:** *Oracle Database Backup and Recovery Basics* for more information about backing up a database.

Backing Up the Database

After cleanly shutting down the instance following the analysis of the database, you should perform a full backup of the database. Complete the following steps:

1. Log in to Oracle Recovery Manager (RMAN):

```
rman "target / nocatalog"
```

2. Issue the following RMAN commands:

```
BACKUP DATABASE FORMAT 'some_backup_directory%U' TAG before_upgrade;
BACKUP CURRENT CONTROLFILE TO 'save_controlfile_location';
```

Backing Up Audit Vault Server Home and Audit Vault Collection Agent Home

Back up or copy the Audit Vault Server home and the Audit Vault collection agent home to separate directories.

4.6 Using a Collection Agent to Listen to Oracle Database RAC Nodes

In an Oracle Real Application Clusters (Oracle RAC) environment, after you have configured the Audit Vault collection agent, the node on which the collection agent was installed has its listener set to listen only to that node. Thus, only that node can be specified to which to connect. However, you can configure the listener to listen to the other nodes.

For the OSAUD and DBAUD collectors, you must update the `tnsnames.ora` file during installation of the Audit Vault collection agents.

After you configure the collection agent, the `tnsnames.ora` file located in `$ORACLE_HOME/network/admin` has an alias similar to the following:

```
AV =
  (DESCRIPTION =
     (ADDRESS = (PROTOCOL = TCP)(HOST = node01) (PORT = 1521))
     (CONNECT_DATA =
       (SERVICE_NAME = avsrv.example.com)))
```

For high availability, you may need to edit the Audit Vault collection agent home `tnsnames.ora` file after you have configured the collection agent, and then add the host and port of the other listeners.

For example:

```
AV =
  (DESCRIPTION =
     (ADDRESS = (PROTOCOL = TCP)(HOST = node01)(PORT = 1521))
     (ADDRESS = (PROTOCOL = TCP)(HOST = node02)(PORT = 1521))
     (ADDRESS = (PROTOCOL = TCP)(HOST = node03)(PORT = 1521))
     (ADDRESS = (PROTOCOL = TCP)(HOST = node04)(PORT = 1521))
     (LOAD_BALANCE = yes)
     (CONNECT_DATA =
       (SERVER = DEDICATED)
         (SERVICE_NAME = avsrv.example.com)
     )
   )
```

For the REDO collector, you must log in using the source user account at the source database and then re-create the database link for `avsrv.example.com`. The new database link can either have a list of host and port numbers or point to a `tnsnames` entry with the list of host and port numbers.

4.7 Configuring Collection Agent Connectivity for Oracle Database RAC

When you add an Oracle source database to Oracle Audit Vault, you must provide the *host*:*port*:*service* information for the source database being added. This information is used for the following tasks from the collection agent:

- **REDO collector:** Starting and stopping the capture process on the source
- **DBAUD collector:** Retrieving rows from `AUD$` and `FGA_LOG$` tables
- **Policy management:** Retrieving source dictionary information

Typically, when the Oracle Database instance on the host goes down or if the host computer goes down, the connectivity to the source database from the Oracle Audit Vault collection agent is broken. Any attempt to perform these tasks is unsuccessful because this connection is not available:

You can do any or all of the following operations to make the connection between the source and the Audit Vault collection agent more highly available.

- **In the Audit Vault collection agent home, update the tnsnames.ora file to include additional host or port information for the service.** This file is located in the `$ORACLE_HOME/network/admin` directory. You can add options for load balancing and failure in the connect string. For additional information, see *Oracle Database Net Services Administrator's Guide*.
- **Configure a listener on the Oracle RAC nodes to support connecting to remote nodes and configuring the Oracle Database to communicate with remote listeners.** If the Oracle Database instance goes down, then the listener on the host can create connections on a different Oracle RAC node. For additional information, see *Oracle Database Net Services Administrator's Guide*.
- **Provide host information using the virtual IP address of the node instead of the physical IP address.** If the host computer goes down, then all traffic to the host is redirected to a different node.

4.8 Purging the Oracle Source Database Audit Trail Records

This section contains:

- General Steps for Purging the Oracle Database Audit Trail
- Step 1: Prepare the Oracle Database Audit Trail for Purging
- Step 2: Create a Job to Automatically Purge the Oracle Database Audit Trail
- Step 3: Optionally, Set a Record Batch Size for the Purge Operations
- Step 4: Perform Maintenance Tasks as Needed

4.8.1 General Steps for Purging the Oracle Database Audit Trail

An Oracle Database administrator (not necessarily an Oracle Audit Vault administrator) is responsible for purging audit data from the Oracle source database.

Follow these general steps to purge the Oracle Database audit trail records from an Oracle source database:

1. If necessary, tune online and archive redo log sizes to accommodate the additional records generated during the audit table purge process.

   For more information about tuning log files, see *Oracle Database Performance Tuning Guide* and *Oracle Database Administrator's Guide*.

2. Complete the preparatory steps described in Section 4.8.2.

   You must download and install the DBMS_AUDIT_MGMT PL/SQL package, which is available as a patch set from the Oracle*MetaLink* Web site. After you install this package, you must move the database audit trail to a different tablespace before you can purge the audit trail.

3. Configure an automatic purge job by following the steps in Section 4.8.3.

4. After you configure the purge time for the automatic purge job and before the purge occurs, optionally configure the audit records for batch deletions. For very large audit trails, deleting the records in batches helps to speed the purge process. See Section 4.8.4.

5. Perform maintenance tasks as needed, as described in Section 4.8.5.

Note: Oracle Database audits all deletions from the audit trail, without exception.

See Also:

- Chapter 14 for information about the DBMS_AUDIT_TRAIL PL/SQL package
- Chapter 13 for information about data dictionary views that you can use while completing these steps

4.8.2 Step 1: Prepare the Oracle Database Audit Trail for Purging

This section contains:

- Step 1A: Download the DBMS_AUDIT_MGMT Package
- Step 1B: Move the Database Audit Trail to a Different Tablespace

4.8.2.1 Step 1A: Download the DBMS_AUDIT_MGMT Package

The DBMS_AUDIT_MGMT PL/SQL package enables you to perform the following tasks with the Oracle Database audit trail:

- Move the database audit trail from the SYSTEM tablespace to a different tablespace, such as the SYSAUX tablespace.
- Set the size and age of the operating system audit trail file before creating a new operating system audit trail file.
- Purge the audit trail records, either by manually purging the records or by creating a purge job.

The DBMS_AUDIT_MGMT PL/SQL package is available in a patch set. Check Oracle*MetaLink* and the *Oracle Audit Vault Release Notes* for information about the specific Oracle Database versions you can use with this package.

The Oracle*MetaLink* Web site is at

`https://metalink.oracle.com`

If you do not have a current Oracle Support Services contract, then you can access the same information at the following Web site:

`http://www.oracle.com/technology/support/metalink/content.html`

See the following sections for information about using the DBMS_AUDIT_MGMT package:

- Section 4.8 for general procedures for using the DBMS_AUDIT_MGMT package
- Chapter 14 for reference information on the DBMS_AUDIT_MGMT package
- Chapter 13 for information about the data dictionary views that accompany the DBMS_AUDIT_MGMT package

4.8.2.2 Step 1B: Move the Database Audit Trail to a Different Tablespace

The SYSTEM tablespace stores the database audit trail AUD$ and FGA_LOG$ tables. When you initialize the purge process, by default Oracle Database moves the AUD$ and FGA_LOG$ tables to the SYSAUX tablespace. If you prefer to store these tables in a different tablespace, follow the procedures in this section.

Be aware that moving the database audit trail tables to a different tablespace can take a while, so you may want to do this during a time when database activity is slow.

To move the database audit trail from SYSTEM to a different tablespace:

1. Log in to SQL*Plus as an administrator who has the EXECUTE privilege on the DBMS_AUDIT_MGMT PL/SQL package.

   For more information about the DBMS_AUDIT_MGMT PL/SQL package, see Chapter 14.

2. Check the tablespace to which you want to move the database audit trail tables.

   You may need to optimize and allocate more space to this tablespace, including the SYSAUX auxiliary tablespace. For more information, see *Oracle Database Performance Tuning Guide*.

3. Run the DBMS_AUDIT_MGMT.SET_AUDIT_TRAIL_LOCATION PL/SQL procedure to specify the name of the destination tablespace and move it to that tablespace.

   For example:

```
BEGIN
 DBMS_AUDIT_MGMT.SET_AUDIT_TRAIL_LOCATION(
  AUDIT_TRAIL_TYPE            => DBMS_AUDIT_MGMT.AUDIT_TRAIL_AUD_STD,
  AUDIT_TRAIL_LOCATION_VALUE  => 'AUD_AUX');
END;
```

   In this example:

   - AUDIT_TRAIL_TYPE: Refers to the database audit trail type. Enter one of the following values:
     - DBMS_AUDIT_MGMT.AUDIT_TRAIL_AUD_STD: Refers to the standard audit trail table, AUD$.
     - DBMS_AUDIT_MGMT.AUDIT_TRAIL_FGA_STD: Refers to the fine-grained audit trail table, FGA_LOG$.
     - DBMS_AUDIT_MGMT.AUDIT_TRAIL_DB_STD: Refers to both standard and fine-grained audit trail tables.
   - AUDIT_TRAIL_LOCATION_VALUE: Specifies the destination tablespace. This example specifies a tablespace named AUD_AUX.

4.8.3 Step 2: Create a Job to Automatically Purge the Oracle Database Audit Trail

The automatic purge job deletes all audit records that were created before the last recorded timestamp. Be aware that purging the audit trail, particularly a large one, can take a while to complete. Consider scheduling the purge job so that it runs during a time when the database is not busy.

To set up an automatic purge job:

- Step 2A: Ensure That the Collectors Are Enabled
- Step 2B: Initialize the Audit Trail Cleanup Operation
- Step 2C: Create the Purge Job

4.8.3.1 Step 2A: Ensure That the Collectors Are Enabled

Ensure that the Oracle Audit Vault collectors are recording timestamps and archiving the audit trail records. See Section 2.9 to check the status of the collectors. To find the last recorded timestamp, query the `LAST_ARCHIVE_TS` column of the `DBA_AUDIT_MGMT_LAST_ARCH_TS` data dictionary view, described in Section 13.2. If the collectors are disabled, then this view shows the last recorded timestamp that occurred before the collector was disabled.

4.8.3.2 Step 2B: Initialize the Audit Trail Cleanup Operation

Before you can purge the audit trail, you must initialize the audit trail cleanup operation. For the database audit trail, if you have not moved the database audit trail tables (`SYS.AUD$` and `SYS.FGA_LOG$`) from the `SYSTEM` tablespace to another tablespace, this process moves these tables to the `SYSAUX` tablespace or to the tablespace that you specified in Section 4.8.2.2. Be aware that moving these tables takes a while, so you may want to schedule the initialization process during time when the database is not busy.

To initialize the audit trail cleanup operation:

1. Log in to SQL*Plus as an administrative user who has the `EXECUTE` privilege on the `DBMS_AUDIT_MGMT` PL/SQL package.
2. Initialize the audit trail cleanup operation by running the `DBMS_AUDIT_MGMT.INIT_CLEANUP` procedure.

   For example:

```
BEGIN
 DBMS_AUDIT_MGMT.INIT_CLEANUP(
  AUDIT_TRAIL_TYPE            => DBMS_AUDIT_MGMT.AUDIT_TRAIL_DB_AUD,
  DEFAULT_CLEANUP_INTERVAL    => 12 );
END
```

   In this example:

   - `AUDIT_TRAIL_TYPE`: Enter one of the following values:
     - `DBMS_AUDIT_MGMT.AUDIT_TRAIL_AUD_STD`: Standard audit trail table, `AUD$`
     - `DBMS_AUDIT_MGMT.AUDIT_TRAIL_FGA_STD`: Fine-grained audit trail table, `FGA_LOG$`
     - `DBMS_AUDIT_MGMT.AUDIT_TRAIL_DB_STD`: Both standard and fine-grained audit trail tables

  - DBMS_AUDIT_MGMT.AUDIT_TRAIL_OS: Operating system audit trail files with the .aud extension. (This setting does not apply to Windows event log entries.)
  - DBMS_AUDIT_MGMT.AUDIT_TRAIL_XML: XML operating system audit trail files with the .xml extension
  - DBMS_AUDIT_MGMT.AUDIT_TRAIL_FILES: Both operating system and XML audit trail files
  - DBMS_AUDIT_MGMT.AUDIT_TRAIL_ALL: All audit trail records, that is, both database audit trail and operating system audit trail types
- DEFAULT_CLEANUP_INTERVAL: Specify the desired default hourly purge interval (for example, 12 for every 12 hours). The DBMS_AUDIT_MGMT procedures use this value to determine how to purge audit records. The timing begins when you run the DBMS_AUDIT_MGMT.INIT_CLEANUP procedure. To update this value later, set the DBMS_AUDIT_MGMT.CLEAN_UP_INTERVAL property of the DBMS_AUDIT_MGMT.SET_AUDIT_TRAIL_PROPERTY procedure.

4.8.3.3 Step 2C: Create the Purge Job

Create the purge job by running the DBMS_AUDIT_MGMT.CREATE_PURGE_JOB PL/SQL procedure.

For example:

```
BEGIN
  DBMS_AUDIT_MGMT.CREATE_PURGE_JOB (
   AUDIT_TRAIL_TYPE            => DBMS_AUDIT_MGMT.AUDIT_TRAIL_DB_AUD,
   AUDIT_TRAIL_PURGE_INTERVAL  => 12,
   AUDIT_TRAIL_PURGE_NAME      => 'Standard_Audit_Trail_PJ',
   USE_LAST_ARCH_TIMESTAMP     => TRUE );
END;
```

In this example:

- AUDIT_TRAIL_TYPE: Enter one of the following values:
  - DBMS_AUDIT_MGMT.AUDIT_TRAIL_DB_AUD: Standard audit trail table, AUD$
  - DBMS_AUDIT_MGMT.AUDIT_TRAIL_FGA_STD: Fine-grained audit trail table, FGA_LOG$
  - DBMS_AUDIT_MGMT.AUDIT_TRAIL_DB_STD: Both standard and fine-grained audit trail tables
  - DBMS_AUDIT_MGMT.AUDIT_TRAIL_OS: Operating system audit trail files with the .aud extension. (This setting does not apply to Windows event log entries.)
  - DBMS_AUDIT_MGMT.AUDIT_TRAIL_XML: XML operating system audit trail files with the .xml extension
  - DBMS_AUDIT_MGMT.AUDIT_TRAIL_FILES: Both operating system and XML audit trail files
  - DBMS_AUDIT_MGMT.AUDIT_TRAIL_ALL: All audit trail records, that is, both database audit trail and operating system audit trail types
- AUDIT_TRAIL_PURGE_INTERVAL: Specify the hourly interval for this purge job to run. The timing begins when you run the DBMS_AUDIT_MGMT.CREATE_

PURGE_JOB procedure, in this case, 12 hours after you run this procedure. Later on, if you want to update this value, run the DBMS_AUDIT_MGMT.SET_PURGE_JOB_INTERVAL procedure.

- USE_LAST_ARCH_TIMESTAMP: Enter either of the following settings:
  - TRUE: Deletes audit records created before the last archive timestamp. To check the last recorded timestamp, query the LAST_ARCHIVE_TS column of the DBA_AUDIT_MGMT_LAST_ARCH_TS data dictionary view, described in Section 13.2. The default value is TRUE.
  - FALSE: Deletes all audit records without considering the last archive timestamp.

4.8.4 Step 3: Optionally, Set a Record Batch Size for the Purge Operations

When Oracle Database purges records from the database audit trail, it deletes them in batched groups during the cleanup process. Before the purge occurs, you can set the number of records that best suits your environment. If the database audit trail is very large (and audit trails can grow quite large), deleting the records in groups facilitates the purge operation. To find the current batch size setting, query the PARAMETER_NAME and PARAMETER_VALUE columns of the DBA_AUDIT_MGMT_CONFIG_PARAMS data dictionary view, which is described in Section 13.1.

To set a record batch size, use the DBMS_AUDIT_MGMT.SET_AUDIT_TRAIL_PROPERTY procedure.

For example:

```
BEGIN
 DBMS_AUDIT_MGMT.SET_AUDIT_TRAIL_PROPERTY(
  AUDIT_TRAIL_TYPE            => DBMS_AUDIT_MGMT.AUDIT_TRAIL_AUD_STD,
  AUDIT_TRAIL_PROPERTY        => DBMS_AUDIT_MGMT.DB_DELETE_BATCH_SIZE,
  AUDIT_TRAIL_PROPERTY_VALUE  => 100000);
END
```

In this example:

- AUDIT_TRAIL_TYPE: Specifies the audit trail type, which in this case is the database system audit trail. Enter one of the following values:
  - DBMS_AUDIT_MGMT.AUDIT_TRAIL_DB_AUD: Standard audit trail table, AUD$.
  - DBMS_AUDIT_MGMT.AUDIT_TRAIL_FGA_STD: Fine-grained audit trail table, FGA_LOG$.
- AUDIT_TRAIL_PROPERTY: Uses the DBMS_AUDIT_MGMT.DB_DELETE_BATCH_SIZE property to indicate the batch size setting. To find the status of the current property settings, query the PARAMETER_NAME and PARAMETER_VALUE columns of the DBA_AUDIT_MGMT_CONFIG_PARAMS data dictionary view.
- AUDIT_TRAIL_PROPERTY_VALUE: Sets the number of audit records to be 100,000 for each batch. Enter a value between 100 and 1000000. To determine this number, consider the total number of records being purged, and the time interval in which the purge operation is performed. The default is 10000.

4.8.5 Step 4: Perform Maintenance Tasks as Needed

This section contains:

- Verifying That the Audit Trail Is Initialized for Cleanup
- Enabling or Disabling an Audit Trail Purge Job
- Setting the Default Audit Trail Purge Interval for Any Audit Trail Type
- Setting the Default Audit Trail Purge Job Interval for a Specified Purge Job
- Clearing the Database Audit Trail Records Batch Size
- Canceling the Initialization Cleanup Settings
- Deleting an Audit Trail Purge Job
- Configuring Tracing Debug Levels for Purge Operations
- Setting the Size of the Operating System Audit Trail
- Setting the Age of the Operating System Audit Trail

4.8.5.1 Verifying That the Audit Trail Is Initialized for Cleanup

You can check if the audit trail has been initialized for cleanup by running the `DBMS_AUDIT_MGMT.IS_CLEANUP_INITIALIZED` function. If the audit trail has been initialized, then this function returns `TRUE`. Otherwise, it returns `FALSE`.

For example:

```
SET SERVEROUTPUT ON
BEGIN
 IF
   DBMS_AUDIT_MGMT.IS_CLEANUP_INITIALIZED(DBMS_AUDIT_MGMT.AUDIT_TRAIL_AUD_STD)
 THEN
   DBMS_OUTPUT.PUT_LINE('AUD$ is initialized for cleanup');
 ELSE
   DBMS_OUTPUT.PUT_LINE('AUD$ is not initialized for cleanup.');
 END IF;
END;
```

In this example, the audit trail type is the database standard audit trail. To select a setting for a different audit trail, choose from the `AUDIT_TRAIL_TYPE` settings described in Step 2B: Initialize the Audit Trail Cleanup Operation.

4.8.5.2 Enabling or Disabling an Audit Trail Purge Job

To enable or disable an audit trail purge job, use the `DBMS_AUDIT_MGMT.SET_PURGE_JOB_STATUS` PL/SQL procedure.

For example:

```
BEGIN
 DBMS_AUDIT_MGMT.SET_PURGE_JOB_STATUS(
  AUDIT_TRAIL_PURGE_NAME      => 'OS_Audit_Trail_PJ',
  AUDIT_TRAIL_STATUS_VALUE    => DBMS_AUDIT_MGMT.PURGE_JOB_ENABLE);
END
```

In this example:

- `AUDIT_TRAIL_PURGE_NAME`: Specifies a purge job called `OS_Audit_Trail_PJ`. To find existing purge jobs, query the `JOB_NAME` and `JOB_STATUS` columns of the `DBA_AUDIT_MGMT_CLEANUP_JOBS` data dictionary view.

- AUDIT_TRAIL_STATUS_VALUE: Enter one of the following properties:
  - DBMS_AUDIT_MGMT.PURGE_JOB_ENABLE: Enables the specified purge job.
  - DBMS_AUDIT_MGMT.PURGE_JOB_DISABLE: Disables the specified purge job.

4.8.5.3 Setting the Default Audit Trail Purge Interval for Any Audit Trail Type

You can set a default purge operation interval, in hours, that must pass before the next purge operation occurs for a specified audit trail type.

For example:

```
BEGIN
 DBMS_AUDIT_MGMT.SET_AUDIT_TRAIL_PROPERTY(
  AUDIT_TRAIL_TYPE            => DBMS_AUDIT_MGMT.AUDIT_TRAIL_AUD_STD,
  AUDIT_TRAIL_PROPERTY        => DBMS_AUDIT_MGMT.CLEAN_UP_INTERVAL,
  AUDIT_TRAIL_PROPERTY_VALUE  => 24);
END
```

In this example:

- AUDIT_TRAIL_TYPE: Specifies the audit trail type, which in this case is the database system audit trail. Choose from the following settings:
  - DBMS_AUDIT_MGMT.AUDIT_TRAIL_DB_AUD: Standard audit trail table, AUD$
  - DBMS_AUDIT_MGMT.AUDIT_TRAIL_FGA_STD: Fine-grained audit trail table, FGA_LOG$
  - DBMS_AUDIT_MGMT.AUDIT_TRAIL_DB_STD: Both standard and fine-grained audit trail tables
  - DBMS_AUDIT_MGMT.AUDIT_TRAIL_OS: Operating system audit trail files with the .aud extension. (This setting does not apply to Windows event log entries.)
  - DBMS_AUDIT_MGMT.AUDIT_TRAIL_XML: XML operating system audit trail files with the .xml extension
  - DBMS_AUDIT_MGMT.AUDIT_TRAIL_FILES: Both operating system and XML audit trail files
  - DBMS_AUDIT_MGMT.AUDIT_TRAIL_ALL: All audit trail records, that is, both database audit trail and operating system audit trail types

  You can set a default interval for multiple audit trail types, so long as they do not conflict. For example, you can set individual intervals for the DBMS_AUDIT_MGMT.AUDIT_TRAIL_DB_AUD and DBMS_AUDIT_MGMT.AUDIT_TRAIL_FGA_STD properties, but not for the DBMS_AUDIT_MGMT.AUDIT_TRAIL_DB_STD property.
- AUDIT_TRAIL_PROPERTY: Sets the DBMS_AUDIT_MGMT.CLEAN_UP_INTERVAL property to indicate the purge operation interval setting. To find the current property settings, query the PARAMETER_NAME and PARAMETER_VALUE columns of the DBA_AUDIT_MGMT_CONFIG_PARAMS data dictionary view. The timing begins when you set the DBMS_AUDIT_MGMT.CLEAN_UP_INTERVAL property.
- AUDIT_TRAIL_PROPERTY_VALUE: Updates the default hourly interval set by the DBMS_AUDIT_MGMT.INIT_CLEANUP procedure. Enter a value between 1 and 999.

4.8.5.4 Setting the Default Audit Trail Purge Job Interval for a Specified Purge Job

You can set a default purge operation interval, in hours, that must pass before the next purge job operation occurs. The interval setting that is used in the DBMS_AUDIT_MGMT.CREATE_PURGE_JOB procedure takes precedence over this setting.

For example:

```
BEGIN
 DBMS_AUDIT_MGMT.SET_PURGE_JOB_INTERVAL(
  AUDIT_TRAIL_PURGE_NAME       => 'OS_Audit_Trail_PJ',
  AUDIT_TRAIL_INTERVAL_VALUE   => 24 );
END
```

In this example:

- AUDIT_TRAIL_PURGE_NAME: Specifies the name of the audit trail purge job. To find a list of existing purge jobs, query the JOB_NAME and JOB_STATUS columns of the DBA_AUDIT_MGMT_CLEANUP_JOBS data dictionary view.
- AUDIT_TRAIL_INTERVAL_VALUE: Updates the default hourly interval set by the DBMS_AUDIT_MGMT.CREATE_PURGE_JOB procedure. Enter a value between 1 and 999. The timing begins when you run the purge job.

4.8.5.5 Clearing the Database Audit Trail Records Batch Size

To clear this setting, use the DBMS_AUDIT_MGMT.CLEAR_AUDIT_TRAIL_PROPERTY procedure.

For example:

```
BEGIN
  DBMS_AUDIT_MGMT.CLEAR_AUDIT_TRAIL_PROPERTY(
   AUDIT_TRAIL_TYPE        =>  DBMS_AUDIT_MGMT.AUDIT_TRAIL_AUD_STD,
   AUDIT_TRAIL_PROPERTY    =>  DBMS_AUDIT_MGMT.DB_DELETE_BATCH_SIZE,
   USE_DEFAULT_VALUES      =>  TRUE );
END;
```

In this example:

- AUDIT_TRAIL_TYPE: Specifies the audit trail type, which in this case is the database system audit trail. Enter one of the AUDIT_TRAIL_TYPE values listed in Section 4.8.4.
- AUDIT_TRAIL_PROPERTY: Specifies the DB_DELETE_BATCH_SIZE property. Query the PARAMETER_NAME and PARAMETER_VALUE columns of the DBA_AUDIT_MGMT_CONFIG_PARAMS data dictionary view to find the current value of this property.
- USE_DEFAULT_VALUES: Enter one of the following values:
  - TRUE: Clears the current audit record batch size and uses the default value, 10000, instead.
  - FALSE: Oracle Database does not set any batch size for audit records. The default setting is FALSE.

4.8.5.6 Canceling the Initialization Cleanup Settings

You can cancel the DBMS_AUDIT_MGMT.INIT_CLEANUP settings, that is, the default cleanup interval, by invoking the DBMS_AUDIT_MGMT.DEINIT_CLEANUP procedure.

For example, to cancel all purge settings for the standard audit trail:

```
BEGIN
 DBMS_AUDIT_MGMT.DEINIT_CLEANUP(
  AUDIT_TRAIL_TYPE  => DBMS_AUDIT_MGMT.AUDIT_TRAIL_AUD_STD);
END;
```

In this example:

- AUDIT_TRAIL_TYPE: Enter one of the AUDIT_TRAIL_TYPE settings listed in Step 2B: Initialize the Audit Trail Cleanup Operation.

4.8.5.7 Deleting an Audit Trail Purge Job

To delete an audit trail purge job, use the DBMS_AUDIT_MGMT.DROP_PURGE_JOB PL/SQL procedure. To find existing purge jobs, query the JOB_NAME and JOB_STATUS columns of the DBA_AUDIT_MGMT_CLEANUP_JOBS data dictionary view.

For example:

```
BEGIN
 DBMS_AUDIT_MGMT.DROP_PURGE_JOB(
  AUDIT_TRAIL_PURGE_NAME  => 'FGA_Audit_Trail_PJ');
END
```

In this example:

- AUDIT_TRAIL_PURGE_NAME: Specifies a purge job called FGA_Audit_Trail_PJ.

4.8.5.8 Configuring Tracing Debug Levels for Purge Operations

To diagnose errors, you can set the trace level for purge operations. Oracle Database creates trace files in the location set by the USER_DUMP_DEST initialization parameter. To find this location, log in to SQL*Plus and enter SHOW PARAMETER USER_DUMP_DEST.

As an example of the type of error the trace debug levels can catch, suppose you try to move the database audit trail table from SYSTEM to a different tablespace. Before moving the tables to the new tablespace, Oracle Database checks the space of the destination tablespace to ensure that it can hold the database audit trail tables. The debug log level can reveal if there is not enough space.

Use the DBMS_AUDIT_MGMT.SET_DEBUG_LEVEL PL/SQL procedure to set the trace level.

For example:

```
BEGIN
DBMS_AUDIT_MGMT.SET_DEBUG_LEVEL(
 DEBUG_LEVEL   => TRACE_LEVEL_DEBUG);
END
```

In this example:

- DEBUG_LEVEL: Specify one of the following values:
  - TRACE_LEVEL_ERROR records errors. This is the default setting.
  - TRACE_LEVEL_DEBUG records detailed information that you may want to capture for debugging purposes. Use this setting to diagnose a problem that is occurring.

4.8.5.9 Setting the Size of the Operating System Audit Trail

To control the size of the operating system audit trail, set the DBMS_AUDIT_MGMT.OS_FILE_MAX_SIZE property by using the DBMS_AUDIT_MGMT.SET_AUDIT_TRAIL_PROPERTY PL/SQL procedure. Remember that you must have the EXECUTE privilege for the DBMS_AUDIT_MGMT PL/SQL package before you can use it. When the operating system file meets the size limitation you set, Oracle Database stops adding records to the current file and then creates a new operating system file for the subsequent records.

If you set both the DBMS_AUDIT_MGMT.OS_FILE_MAX_SIZE and the DBMS_AUDIT_MGMT.OS_FILE_MAX_AGE (described in Section 4.8.5.9) properties, then Oracle Database performs the action based on the property value limit that is met first.

For example:

```
BEGIN
  DBMS_AUDIT_MGMT.SET_AUDIT_TRAIL_PROPERTY(
   AUDIT_TRAIL_TYPE            =>  DBMS_AUDIT_MGMT.AUDIT_TRAIL_OS,
   AUDIT_TRAIL_PROPERTY        =>  DBMS_AUDIT_MGMT.OS_FILE_MAX_SIZE,
   AUDIT_TRAIL_PROPERTY_VALUE  =>  102400);
END;
```

In this example:

- AUDIT_TRAIL_TYPE: Specifies the operating system audit trail. Enter one of the following values:
  - DBMS_AUDIT_MGMT.AUDIT_TRAIL_OS: Operating system audit trail files with the .aud extension. (This setting does not apply to Windows event log entries.)
  - DBMS_AUDIT_MGMT.AUDIT_TRAIL_XML: XML audit trail files.
  - DBMS_AUDIT_MGMT.AUDIT_TRAIL_FILES: Both operating system and XML audit trail files.
- AUDIT_TRAIL_PROPERTY: Specifies the DBMS_AUDIT_MGMT.OS_FILE_MAX_SIZE property to set the maximum size. To find the status of the current property settings, query the PARAMETER_NAME and PARAMETER_VALUE columns of the DBA_AUDIT_MGMT_CONFIG_PARAMS data dictionary view.
- AUDIT_TRAIL_PROPERTY_VALUE: Sets the maximum size to 102400 kilobytes. The default setting is 10,000 kilobytes (approximately 10 MB). Do not exceed 2 gigabytes.

Clearing the DBMS_AUDIT_MGMT.OS_FILE_MAX_SIZE Setting

To clear the maximum file size setting, use the DBMS_AUDIT_MGMT.CLEAR_AUDIT_TRAIL_PROPERTY procedure.

For example:

```
BEGIN
  DBMS_AUDIT_MGMT.CLEAR_AUDIT_TRAIL_PROPERTY(
   AUDIT_TRAIL_TYPE         =>  DBMS_AUDIT_MGMT.AUDIT_TRAIL_OS,
   AUDIT_TRAIL_PROPERTY     =>  DBMS_AUDIT_MGMT.OS_FILE_MAX_SIZE,
   USE_DEFAULT_VALUES       =>  TRUE );
END;
```

In this example:

- AUDIT_TRAIL_TYPE: Specifies the operating system audit trail. Enter one of the AUDIT_TRAIL_TYPE values described in Section 4.8.5.3.

- AUDIT_TRAIL_PROPERTY: Specifies the DBMS_AUDIT_MGMT.OS_FILE_MAX_SIZE property. Query the PARAMETER_NAME and PARAMETER_VALUE columns of the DBA_AUDIT_MGMT_CONFIG_PARAMS data dictionary view to find the current status of this property.
- USE_DEFAULT_VALUES: Enter one of the following values:
  - TRUE: Clears the current value and uses the default value, 10,000 kilobytes, instead.
  - FALSE: Oracle Database does not use a default maximum size for the operating system or XML file growth. The files will continue to grow without limitation unless you configure the DBMS_AUDIT_MGMT.OS_FILE_MAX_AGE property. The default setting is FALSE.

4.8.5.10 Setting the Age of the Operating System Audit Trail

To control the age of the operating system audit trail, use the DBMS_AUDIT_MGMT.SET_AUDIT_TRAIL_PROPERTY PL/SQL procedure. Remember that you must have the EXECUTE privilege for the DBMS_AUDIT_MGMT PL/SQL package before you can use it. When the operating system file meets the age limitation you set, Oracle Database stops adding records to the current file and then creates a new operating system file for the subsequent records. For more information about the DBMS_AUDIT_MGMT PL/SQL package, see *Oracle Database PL/SQL Packages and Types Reference*.

If you set both the DBMS_AUDIT_MGMT.OS_FILE_MAX_AGE and the DBMS_AUDIT_MGMT.OS_FILE_MAX_SIZE (described in Section 4.8.5.9) properties, then Oracle Database performs the action based on the property value limit that is met first.

For example:

```
BEGIN
  DBMS_AUDIT_MGMT.SET_AUDIT_TRAIL_PROPERTY(
   AUDIT_TRAIL_TYPE            =>  DBMS_AUDIT_MGMT.AUDIT_TRAIL_OS,
   AUDIT_TRAIL_PROPERTY        =>  DBMS_AUDIT_MGMT.OS_FILE_MAX_AGE,
   AUDIT_TRAIL_PROPERTY_VALUE  =>  10 );
END;
```

In this example:

- AUDIT_TRAIL_TYPE: Specifies the operating system audit trail. Enter one of the following values:
  - DBMS_AUDIT_MGMT.AUDIT_TRAIL_OS: Operating system audit trail files with the .aud extension. (This setting does not apply to Windows Event Log entries.)
  - DBMS_AUDIT_MGMT.AUDIT_TRAIL_XML: XML audit trail files.
  - DBMS_AUDIT_MGMT.AUDIT_TRAIL_FILES: Both operating system and XML audit trail files.
- AUDIT_TRAIL_PROPERTY: Specifies the DBMS_AUDIT_MGMT.OS_FILE_MAX_AGE property to set the maximum age. To find the status of the current property setting, query the PARAMETER_NAME and PARAMETER_VALUE columns of the DBA_AUDIT_MGMT_CONFIG_PARAMS data dictionary view.
- AUDIT_TRAIL_PROPERTY_VALUE: Sets the maximum age to 10 days. Enter a value between 1 and 495. The default age is 5 days.

Clearing the DBMS_AUDIT_MGMT.OS_FILE_MAX_AGE Setting

To clear the maximum file age setting, use the DBMS_AUDIT_MGMT.CLEAR_AUDIT_TRAIL_PROPERTY procedure.

For example:

```
BEGIN
  DBMS_AUDIT_MGMT.CLEAR_AUDIT_TRAIL_PROPERTY(
   AUDIT_TRAIL_TYPE        =>  DBMS_AUDIT_MGMT.AUDIT_TRAIL_OS,
   AUDIT_TRAIL_PROPERTY    =>  DBMS_AUDIT_MGMT.OS_FILE_MAX_AGE,
   USE_DEFAULT_VALUES      =>  TRUE );
END;
```

In this example:

- AUDIT_TRAIL_TYPE: Specifies the operating system audit trail. Enter one of the AUDIT_TRAIL_TYPE values listed in Section 4.8.5.9.
- AUDIT_TRAIL_PROPERTY: Specifies the DBMS_AUDIT_MGMT.OS_FILE_MAX_AGE property. Query the PARAMETER_NAME and PARAMETER_VALUE columns of the DBA_AUDIT_MGMT_CONFIG_PARAMS data dictionary view, described in Section 13.1, to find the current status of this property.
- USE_DEFAULT_VALUES: Specify one of the following values:
  - TRUE: Clears the current value and uses the default value, 5 days, instead.
  - FALSE: Oracle Database does not use a default maximum age for the operating system or XML file growth. The files will continue to age without limitation unless you configure the DBMS_AUDIT_MGMT.OS_FILE_MAX_SIZE property. The default setting is FALSE.

5

Managing Oracle Audit Vault Security

This chapter contains:

- About Managing Oracle Audit Vault Security
- Managing Authentication Metadata Using Oracle Advanced Security
- Using Oracle Database Vault with Oracle Audit Vault
- Changing Oracle Audit Vault User Passwords on a Regular Basis
- Configuring HTTPS Communication for Oracle Audit Vault

5.1 About Managing Oracle Audit Vault Security

Oracle Audit Vault includes Oracle Advanced Security and Oracle Database Vault features to protect audit data that it collects and stores.

This chapter explains how to manage Oracle Audit Vault security. You should perform Oracle Audit Vault security tasks in this order of importance:

1. Secure management communication between the Oracle Audit Vault Server and collection agent, described in Section 5.5.
2. Manage user authentication metadata, described in Section 5.2.

Section 5.3 explains how Oracle Database Vault protects audit data and provides strong access control.

5.2 Managing Authentication Metadata Using Oracle Advanced Security

As part of the Audit Vault Server and the Oracle Audit Vault collection agent installation, two wallets are created. One wallet resides on the Audit Vault Server and this one contains the credentials of the `AV_ADMIN` user. The Audit Vault Console uses this wallet to communicate with the Oracle Audit Vault database. The Audit Vault Console provides the management service that initiates the communication with collection agents using HTTP. Audit Vault Configuration Assistant (`AVCA`) modifies the Database Control console `server.xml` file and other related files to enable Oracle Audit Vault management through the Oracle Enterprise Manager Database Control console. The wallet is located in the `$ORACLE_HOME/network/admin/avwallet` directory.

The other wallet resides on the Audit Vault collection agent and contains the `AV_AGENT` credentials. The collection agent uses this wallet to get configuration data from Oracle Audit Vault. This wallet is located in the `$ORACLE_HOME/network/admin/avwallet` directory. This wallet also contains the credentials used by the collectors to communicate with the source database (Oracle Database,

Microsoft SQL Server database, Sybase ASE, or IBM DB2 database). The three ORCLDB collectors, the MSSQLDB collector, the SYBDB collector, and the DB2DB collector all use these credentials to connect to the source database and to:

- Open a connection to the source database to read, extract, and send audit records to the Audit Vault repository
- Obtain metadata and metrics for all the collectors
- Start and stop the collectors
- Obtain audit settings as part of Audit Settings management for ORCLDB collectors

The Oracle wallet is a password-protected container that stores credentials, such as certificates, authentication credentials, and private keys, all of which are used by SSL for strong authentication. You can manage Oracle wallets by using Oracle Wallet Manager. Oracle Wallet Manager can perform tasks such as wallet creation, certificate request generation, and importing certificates into the wallet.

Oracle Audit Vault uses third-party network authentication services (PKI-based authentication) to authenticate its user clients. Authentication systems based on **public key infrastructure** (**PKI**) issue digital certificates to user clients, which use them to authenticate directly to servers in the enterprise without involving an authentication server. These user certificates, along with the private key of the user and the set of trust points of a user (trusted certificate authorities), are stored in Oracle wallets.

5.3 Using Oracle Database Vault with Oracle Audit Vault

By default, Oracle Database Vault is enabled in the Audit Vault Server. Oracle Database Vault restricts access to the data in the Audit Vault Server from any user, including users who have administrative access. For Oracle Audit Vault, Oracle Database Vault protects the Audit Vault Server by using a realm. To ensure that the data in the Audit Vault Server is protected, do not disable Oracle Database Vault.

The inclusion of Oracle Database Vault provides the `DV_OWNER` and `DV_ACCTMGR` roles. The `DV_OWNER` role manages the database roles and configuration, and the `DV_ACCTMGR` role manages user accounts. As with all Oracle Database roles, grant these roles only to those users who are responsible for the tasks associated with the role.

Be aware that Oracle Database Vault revokes some privileges from several roles supplied by the Oracle database roles, including `SYS` and `SYSTEM`. *Oracle Database Vault Administrator's Guide* describes roles and privileges that Oracle Database Vault affects. Remember that only the user who has been granted the `DV_ACCTMGR` role can create, alter, and drop users. However, the `DV_ACCTMGR` user cannot grant these roles to these users. Only the user who has been granted the `AV_ADMIN` role can grant the `AV_ADMIN` and `AV_AUDITOR` roles to another user.

Table 5–1 shows the roles and privileges an administrative user is granted when that user is granted and Oracle Audit Vault or Oracle Database Vault roles. For detailed information about the Oracle Audit Vault or Oracle Database Vault roles, see Section 1.5.

Table 5–1 Roles and Privileges Granted to Audit Vault or Database Vault Administrators

Role Granted to User	Roles Granted to This Role	Privileges Granted
AV_ADMIN	SELECT_CATALOG_ROLE	CREATE SESSION
	AQ_ADMINISTRATOR_ROLE	GRANT ANY ROLE
	AV_AUDITOR[1]	
	AV_AGENT	
	XDBADMIN	
AV_AUDITOR	SELECT_CATALOG_ROLE	CREATE SESSION
AV_AGENT	No additional roles granted	CREATE SESSION
		CREATE ANY VIEW
DV_ACCTMGR	DV_PUBLIC	CREATE SESSION
	CONNECT	CREATE USER
		ALTER USER
		DROP USER
		CREATE PROFILE
		ALTER PROFILE
		DROP PROFILE
DV_OWNER	DV_PUBLIC	CREATE SESSION
	CONNECT	GRANT ANY ROLE
	DV_ADMIN	ALTER ANY TRIGGER
	DV_SECANALYST	ADMINISTER DATABASE TRIGGER

[1] The AV_ADMIN role is granted the AV_AUDITOR role only if you did not create the AV_AUDITOR user during installation.

Table 5–2 shows other database core accounts that are created in the default Oracle Audit Vault installation. Oracle Audit Vault permits operating system authentication to the database. It disables remote authentication to the database if you try to use the SYSDBA privilege, but if it is needed, you can enable it by using a password file. See the sections that discuss postinstallation tasks in the *Oracle Audit Vault Installation Guide* for more information about unlocking and resetting user passwords and enabling or disabling connections with the SYSDBA privilege.

Table 5–2 Database Core Accounts Created and Privileges Use

Account	Privileges	Privilege In Use	Password to Use
SYS SYSTEM SYSMAN DBSNMP	Many[1]	Yes	Use same password as user granted AV_ADMIN role for basic installation or password may be set separately in advanced installation
SYS AS or / AS	SYSDBA	Yes, allowed	Operating system authentication to the database is enabled by default.
SYS AS	SYSDBA	No, not allowed for remote connection	To use for remote connection, user must create a password file to enable its use. Password is set when password file is created.

Table 5–2 (Cont.) Database Core Accounts Created and Privileges Use

Account	Privileges	Privilege In Use	Password to Use
SYS AS	SYSOPER	Yes, allowed	Use same password as user granted AV_ADMIN role

[1] To find the privileges associated with the user account, log in to SQL*Plus as the user and then run the following query: `SELECT * FROM SESSION_ROLES;`

5.4 Changing Oracle Audit Vault User Passwords on a Regular Basis

This section contains:

- About Oracle Audit Vault User Passwords
- Changing the AV_ADMIN User Password
- Changing the AV_AGENT Password
- Changing the Source User Password
- Changing the AV_AUDITOR Password
- Ensuring That All Changed User Name Passwords Work Correctly

5.4.1 About Oracle Audit Vault User Passwords

You should have a policy in place for changing passwords for the Oracle Audit Vault user accounts. For example, you may require that users change their passwords on a regular basis, such as every 120 days, and that they create passwords that are not easily guessed.

Table 5–3 summarizes guidelines that you must follow when you change passwords for the Oracle Audit Vault user accounts.

Table 5–3 Storage Location of Audit Vault and Source User Name Passwords

Audit Vault Role	Is Password Stored in Wallet?	How Do I Change the Password?
AV_ADMIN	Yes	Use the `avca create_credential` command to change the password in the wallet in the Audit Vault Server home. You must also change the password of this user in the database. To do so, use the `ALTER USER` SQL statement. See Section 5.4.2.
AV_AGENT	Yes	Use the `avca create_credential` command to change the password in the wallet in the Audit Vault collection agent home. You must also change the password of this user in the database. To do so, use the `ALTER USER` SQL statement. See Section 5.4.3.
Source user on source database	Yes	For an Oracle Database source user account, use the `ALTER USER` SQL statement in the source database Audit Vault Server home. Use the `setup` command of the `AVORCLDB`, `AVMSSQLDB`, `AVSYBDB`, or `AVDB2DB` utility to change the password in the wallet in the Audit Vault collection agent home See Section 5.4.3

Table 5–3 (Cont.) Storage Location of Audit Vault and Source User Name Passwords

Audit Vault Role	Is Password Stored in Wallet?	How Do I Change the Password?
AV_AUDITOR	No	Use the ALTER USER SQL statement in the Audit Vault Server home. See Section 5.4.5.

5.4.2 Changing the AV_ADMIN User Password

After you have updated the AV_ADMIN user account using the ALTER USER SQL statement, you must update the password credentials of this user.

To change the password of a user who has been granted the AV_ADMIN role:

1. In the server where you installed the Oracle Audit Vault Server, open a shell.
2. Log in to SQL*Plus as the user whose password you must change, another user who has been granted the ALTER_USER privilege, or a user with the DV_ACCTMGR role, and then change the password.

   For example:

   ```
   sqlplus avadmindva
   Enter password: password
   Connected.

   SQL> ALTER USER avadminusr IDENTIFIED BY password;
   ```

3. Exit SQL*Plus.
4. Set the environment variables for the Audit Vault Server home, as described in Section 2.2.2.
5. From the shell, run the avca create_credential command to change the password credentials of the AV_ADMIN user.

   For example:

   ```
   avca create_credential -wrl $ORACLE_HOME/network/admin/avwallet -dbalias orcl
   AVCA started
   Storing user credentials in wallet...
   Enter source user username: avadminuser
   Enter source user password: password
   Re-enter source user password: password
   Create credential Modify credential
   Modify 2
   done.
   ```

   In this example, the dbalias parameter specifies the Audit Vault Server SID in the Audit Vault Server home. You can find this information by running the lsnrctl status command on the computer where you installed the source database. For detailed information about using the avca create_credential command, see Section 6.2.

5.4.3 Changing the AV_AGENT Password

After you have updated the AV_AGENT stored password credentials, you must update the password credentials of this account.

To change the password credentials for the AV_AGENT user account:

1. In the server where you installed the Oracle Audit Vault collection agent, open a shell.
2. Set the environment variables for the Audit Vault collection agent home, as described in Section 2.2.3.

   If you installed the collection agent on Microsoft Windows, go to the *ORACLE_HOME**agent_dir*\bin directory. You do not need to set any environment variables.
3. Log in to SQL*Plus and use the ALTER USER SQL statement to change the password of the AV_AGENT user.

   For example:

```
sqlplus avadmindva
Enter password: password
Connected.
SQL> ALTER USER avagent_usr IDENTIFIED BY password;
```

4. Change the password credential of the AV_AGENT user account.

   For example:

```
avca create_credential -wrl $ORACLE_HOME/network/admin/avwallet -dbalias av
AVCA started
Storing user credentials in wallet...
Enter source user username: avagentuser
Enter source user password: password
Re-enter source user password: password
Create credential Modify credential
Modify 2
done.
```

   For detailed information about using the avca create_credential command, see Section 6.2.

5.4.4 Changing the Source User Password

After you have updated the source database stored password credential, you must update the password credentials of this account.

To change the password credentials for the source user account:

1. In the server where you installed the Audit Vault Server, open a shell and then set the environment variables for the Audit Vault Server home, as described in Section 2.2.2.
2. In the Audit Vault Server home, use the ALTER USER SQL statement to change the password for the source user account if it is an Oracle Database source user account.

   For example:

```
sqlplus avadmindva
Enter password: password
Connected.
SQL> ALTER USER srcuser_ora IDENTIFIED BY password;
```

   For source user accounts created for Microsoft Windows, Sybase ASE, and IBM DB2, log in to the appropriate source database and then change the password there.

3. Open a shell for the Audit Vault collection agent, and then set its environment variables as described in Section 2.2.3.

   If you installed the collection agent on Microsoft Windows, do not set any environment variables. Instead, go to the *ORACLE_HOME**agent_dir*\bin directory.

4. Run the `avorcldb setup` command.

   For example:

```
avorcldb setup -srcname hrdb.example.com
Enter Source user name: srcuser_ora
Enter Source password: password
adding credentials for user srcuser_ora for connection [SRCDB1]
Storing user credentials in wallet...
Create credential oracle.security.client.connect_string3
done.
updated tnsnames.ora with alias [SRCDB1] to source database
verifying SRCDB1 connection using wallet
```

   For detailed information about using the `avorcldb setup` command, see Section 8.9. Depending on where you created the source user account, see the following sections:

   - **Microsoft SQL Server:** Section 9.9
   - **Sybase ASE:** Section 10.9
   - **IBM DB2:** Section 11.9

5.4.5 Changing the AV_AUDITOR Password

To change the password of a user who has been granted the `AV_AUDITOR` role, you must change the passwords in both the Audit Vault Server home in the Audit Vault database by using the SQL `ALTER_USER` command. Log in as the user with the role of Database Vault Account Manager.

For example:

1. In the server where you installed the Audit Vault Server, open a shell and then set the environment variables for the Audit Vault Server home, as described in Section 2.2.2.

2. Log in to SQL*Plus as the Database Vault Account Manager (that is, a user who has been granted the `DV_ACCTMGR` role).

   For example:

```
sqlplus avadmindva
Enter password: password
Connected.
SQL>
```

3. Use the `ALTER USER` SQL statement to change the `AV_AUDITOR` user account.

   For example:

```
SQL> ALTER USER avauditorusr-name IDENTIFIED BY password;
```

5.4.6 Ensuring That All Changed User Name Passwords Work Correctly

To test the changed passwords for users who have been granted the `AV_ADMIN` and `AV_AUDITOR` roles, log in to the Audit Vault Console as the Audit Vault administrator

and then as the Audit Vault auditor. See Section 3.2.3 for instructions on logging in to the Audit Vault Console. If the login is not successful, repeat the procedures described in this section to re-create the passwords, and then retest them.

For the AV_ADMIN role, you must also test that the credentials were stored correctly in the wallet.

Follow these steps:

1. In the server where you installed the Audit Vault Server, open a shell and then set the environment variables for the Audit Vault Server home, as described in Section 2.2.2.
2. In SQL*Plus, log in to the Audit Vault Server.

   For example, assuming the SID of the Audit Vault Server is av:

   ```
   sqlplus /@av
   ```

To test the AV_AGENT and source database user account passwords, stop the collection agents, and then restart the collection agent and each collector. See Chapter 7 for information about the commands you use to perform this test. If you are able to collect new audit records, then the AV_AGENT and source database user account passwords are working. If you cannot collect audit records, then check the log files (see Appendix A for more information) to determine which user name password might be the cause of the problem. If necessary, re-create the passwords and then retest them.

5.5 Configuring HTTPS Communication for Oracle Audit Vault

This section contains:

- About Configuring HTTPS Communication for Oracle Audit Vault
- Step 1: Generate the Certificate Request
- Step 2: Configure the Audit Vault Server and Agent HTTPS Communication

5.5.1 About Configuring HTTPS Communication for Oracle Audit Vault

You can secure management communication between the Oracle Audit Vault Server and collection agent by using the **HTTPS** protocol to encrypt data. In this case, you provide **X.509 certificate**s for authentication. This section explains how to configure Secure Sockets Layer (SSL) for the mutual authentication between Oracle Audit Vault on the server side and each collection agent over HTTPS. A certificate authority (CA) must provide these certificates to you, the Oracle Audit Vault administrator.

To accomplish this, you secure the following services on the server side:

- Oracle Audit Vault Web application, which you secure by using the avca secure_av command.
- XDB services, which you secure by using the avca generate_crs and avca import commands. These commands enable you to generate a

For the agent side, you secure OC4J by using avca secure_agent command.

After you secure the Audit Vault Server and Audit Vault collection agent communication to use HTTPS, you must enable the browser to use HTTPS to access the Audit Vault Console. At this stage, HTTP will no longer be available for the browser user because the browser to the Audit Vault Console communication is also made secure.

Before you follow the procedures described in this section, you must understand how to use keystores, which are in JKS (Java Keystore) format from Sun Microsystems. You can create and manage keystores by using the keystore application from Sun Microsystems. See the following URLs for more information:

http://java.sun.com/j2se/1.3/docs/tooldocs/win32/keytool.html

http://www.sslshopper.com/article-most-common-java-keytool-keystore-commands.html

> **See Also:** *Oracle Database Advanced Security Administrator's Guide* for more information about PKI-based authentication, digital certificates, secure external password stores, and Oracle wallets.

5.5.2 Step 1: Generate the Certificate Request

To generate the certificate request:

1. Open a shell for the Audit Vault Server.
2. Follow the instructions in Section 2.2.2 to set the environment variables for the Audit Vault Server.
3. Generate a certificate request for Oracle XML Database using the `avca generate_csr` command.

   (The Oracle Audit Vault reporting interface uses Oracle XML Database.)

   For example:

   ```
   $ avca generate_csr -certdn CN=sales_
   srv.us.example.com,OU=SalesReps,O=RisingDoughCo,ST=CA,C=US -out ca_
   certificate.cer
   ```

   In this example, the certificate request file is called `ca_certificate.cer`.

   See Section 6.6 for detailed information about the `avca generate_csr` command.
4. Send this certificate request file to a CA to be signed and returned to you.
5. Import this signed certificate into the wallet using the `avca import_cert` command.Ensure that you import the trusted CA as well, if the CA is a self-signed one.

   For example:

   ```
   $ avca import_cert -cert user_certificate.cer
   ```

   See Section 6.8 for detailed information about the `avca import_cert` command.
6. Leave the Audit Vault Server shell open.

Next, you can configure both the Audit Vault Server and Oracle XML Database communication using the `avca secure_av` command, as described in the next section.

5.5.3 Step 2: Configure the Audit Vault Server and Agent HTTPS Communication

To configure the Audit Vault Server and collection agent HTTPS communication:

1. Access the shell for the Audit Vault Server.

   If you have closed this shell, reset its environment variables, as described in Section 2.2.2.

If you prefer open a shell for the Audit Vault collection agent, then set its environment variables, as described in Section 2.2.3. If you installed the collection agent on Microsoft Windows, go to the *ORACLE_HOME**agent_dir*\bin directory. You do not need to set any environment variables.

2. Run the **keytool** utility, located in the `$ORACLE_HOME/jdk/bin` directory, to generate a keystore.

   For an example of using the `keytool` utility, see the section that explains how to enable SSL with *i*SQL*Plus in *SQL*Plus User's Guide and Reference*. This utility creates a storage file named `keystore` in the current directory.

   For detailed information about the `keytool` utility, visit the following Web sites:

   `http://java.sun.com/j2se/1.4.2/docs/tooldocs/windows/keytool.html`

   `http://www.sslshopper.com/article-most-common-java-keytool-keystore-commands.html`

   Next, you are ready to configure the mutual authentication between the Audit Vault Server and its collection agents.

3. Access the shell used for the Audit Vault Server.

4. Configure the Audit Vault Server communication with the collection agent.

   For example:

   ```
   $ avca secure_av -avkeystore /tmp/avkeystore -avtruststore /tmp/avkeystore
   Enter keystore password: password
   ```

   See Section 6.12 for detailed information about the `avca secure_av` command.

5. Open a shell for the Audit Vault collection agent, and then follow the instructions in Section 2.2.3 to set its environment variables.

   If you installed the collection agent on Microsoft Windows, go to the *ORACLE_HOME**agent_dir*\bin directory. You do not need to set any environment variables.

6. Secure OC4J and configure the collection agent communication with the Audit Vault Server.

   For example:

   ```
   $ avca secure_agent -agentkeystore /tmp/agentkeystore
   -agentdn "CN=agent1, OU=SalesReps, O=RisingDoughCo, L=Bredville, ST=ca, C=us"
   -avdn "CN=av1, OU=SalesReps, O=RisingDoughCo, L=Bredville, ST=ca, C=us"
   Enter keystore password: password
   ```

   See Section 6.11 for detailed information about the `avca secure_agent` command.

6

Audit Vault Configuration Assistant (AVCA) Reference

Audit Vault Configuration Assistant (AVCA) is a command-line utility you use to manage various Audit Vault components (for example, adding or dropping collection agents). When you run these commands, remember the following:

- **Enter the command in lowercase letters.** The commands are case-sensitive.
- **When you open a new shell to run the command, first set the appropriate environment variables.** See Section 2.2 for more information.
- **Oracle Audit Vault creates a log file of AVCA command activity.** See Section A.1 and Section A.2 for more information.

Table 6–1 describes the Audit Vault Configuration Assistant commands and where each is used, whether on the Audit Vault Server, on the Audit Vault collection agent, or in both places.

Table 6–1 Audit Vault Configuration Assistant Commands

Command	Used Where?	Description
add_agent	Server	Adds a collection agent to Oracle Audit Vault
create_credential	Both	Creates or updates a credential to be stored in the wallet
create_wallet	Collection agent	Creates a wallet to hold credentials
deploy_av	Server	Deploys the `av.ear` file to another node in an Oracle RAC environment
drop_agent	Server	Drops a collection agent from Oracle Audit Vault
generate_csr	Server	Generates a certificate request
-help	Both	Displays help information for the `AVCA` commands
import_cert	Server	Imports the specified certificate into the wallet
redeploy	Both	Redeploys the `av.ear` file on the Audit Vault Server system or the `AVAgent.ear` file on the Audit Vault collection agent system
remove_cert	Server	Removes the specified certificate from the wallet
secure_agent	Collection agent	Secures the Audit Vault collection agent by enabling mutual authentication with Oracle Audit Vault
secure_av	Server	Secures Audit Vault Server by enabling mutual authentication with the Audit Vault collection agent

Table 6–1 (Cont.) Audit Vault Configuration Assistant Commands

Command	Used Where?	Description
set_warehouse_retention	Server	Controls the amount of data kept online in the data warehouse fact table
set_warehouse_schedule	Server	Sets the schedule for refreshing data from the raw audit data store to the audit data warehouse

Note: In an Oracle RAC environment, you must run `AVCA` commands from the node on which Oracle Enterprise Manager resides. This is the same node on which the `av.ear` file is deployed.

If the node on which the `av.ear` file is deployed is down, deploy the `av.ear` file to another node using the `AVCA deploy_av` command.

6.1 add_agent

Adds or registers a collection agent to Oracle Audit Vault. Run this command on the Audit Vault Server.

Syntax

```
avca add_agent -agentname agent_name [-agentdesc desc] -agenthost host
```

Arguments

Argument	Description
`-agentname` *`agent_name`*	Enter the name of the collection agent (by collection agent name) to be added.
`-agentdesc` *`desc`*	Enter a description of the collection agent. Optional.
`-agenthost` *`host`*	Enter the name of an agent host name where this collection agent is to be installed.

Usage Notes

You will be prompted for the agent user name and agent user name password. See the example.

Example

```
$ avca add_agent -agentname TTAgent2 -agenthost stapj40

AVCA started
Adding agent...
Enter agent user name: agent_user_name
Enter agent user password: agent_user_pwd
Re-enter agent user password: agent_user_pwd
Agent added successfully.
```

6.2 create_credential

Creates or updates a credential to be stored in an Oracle wallet. Run this command on both the Audit Vault Server and Audit Vault collection agent during collector development.

Syntax

```
avca create_credential -wrl wallet_location -dbalias db_alias
```

Arguments

Argument	Description
`-wrl` *`wallet_location`*	Enter the location of the Oracle Audit Vault wallet. Locations are as follows: ■ **UNIX and Linux-based systems:** `$ORACLE_HOME/network/admin/avwallet` ■ **Microsoft Windows systems:** *`ORACLE_HOME`*`\network\ADMIN\avwallet`
`-dbalias` *`db_alias`*	Enter the database alias. In the Audit Vault Server home, the database alias is the SID or Oracle instance identifier. You can find this SID by running the `lsnrctl status` command on the computer where you installed the source database.

Usage Notes

- Use this command to create a new certificate if another user changes the source user password on the source database, thus eventually breaking the connection between the collector and the source.
- If you installed the collection agent on a Microsoft Windows computer and want to run the `avca create_credential` command from there, run it from the *`ORACLE_HOME`*`\`*`agent_directory`*`\bin` directory. For UNIX or Linux installations, set the appropriate environment variables before running this command. See Section 2.2 for more information.

Example

```
$ avca create_credential -wrl $ORACLE_HOME/network/admin/avwallet -dbalias av

AVCA started
Storing user credentials in wallet...
Enter source user username: srcuser1
Enter source user password: password
Re-enter source user password: password
Create credential oracle.security.client.connect_string4
done.
```

6.3 create_wallet

Creates a wallet to hold credentials. Run this command on the Audit Vault collection agent.

Syntax

```
avca create_wallet -wrl wallet_location
```

Arguments

Argument	Description
`-wrl` *wallet_location*	Enter the directory location for the wallet. Ensure that this directory already exists. Locations are as follows: ■ **Linux and UNIX-based systems:** `$ORACLE_HOME/network/admin/avwallet` ■ **Microsoft Windows systems:** *ORACLE_HOME*`\network\ADMIN\avwallet`

Usage Notes

- If you installed the collection agent on a Microsoft Windows computer, run the `avca create_wallet` command from the *ORACLE_HOME*`\`*agent_directory*`\bin` directory. For UNIX or Linux installations, set the appropriate environment variables before running this command. See Section 2.2 for more information.
- After you execute this command, `.sso` and `.p12` files are generated in the wallet location.

Example

The following example shows how to create a wallet in the location specified as `$T_WORK/tt_1`:

```
$ avca create_wallet -wrl $T_WORK/tt_1
Enter wallet password: password
```

6.4 deploy_av

Deploys the `av.ear` file to another node in an Oracle Real Application Clusters (Oracle RAC) environment. This command also modifies the `server.xml` file and other related files to enable Oracle Audit Vault management through the Oracle Enterprise Manager Database Control console. Run this command on the Audit Vault Server.

Syntax

```
deploy_av -sid sid -dbalias db_alias -avconsoleport av_console_port
```

Arguments

Argument	Description
`-sid` *sid*	Enter the Oracle Database system identifier (SID) for the instance. You can verify the SID by running the `lsnrctl status` command on the computer where you installed the source database.
`-dbalias` *db_alias*	Enter the database alias
`-avconsoleport` *av_console_port*	Enter the port number for the Audit Vault Console. You can find this number by entering the following command in the Audit Vault Server shell: `avctl show_av_status`

Usage Notes

In an Oracle RAC environment, you must run the AVCA commands from the node on which Oracle Enterprise Manager resides. This is the same node on which the av.ear file is deployed.

If the node on which the av.ear file is deployed is down, deploy the av.ear file to another node using the avca deploy_av command.

When you run the avca deploy_av command on Oracle RAC database, a wallet containing the default avadmin entries is created on the other node. However, other entries, such as the source user credentials must be added to the wallet using the avca create_credential command) being used that matches the collectors that are in use.

To use the Audit Vault Console from this other node, enter its host name or IP address (*host*) and port number (*port*) as you did previously in the Address field of the browser window (http://*host*:*port*/av), but replace the original host name or IP address with that for the other node.

Example

```
$ avca deploy_av -sid av -dbalias av -avconsoleport 5700
```

6.5 drop_agent

Disables (but does not remove) a collection agent from Oracle Audit Vault. Run this command on the Audit Vault Server.

Syntax

```
avca drop_agent -agentname agent_name
```

Arguments

Argument	Description
-agentname *agent_name*	Enter the name of the collection agent to be dropped from Oracle Audit Vault.

Usage Notes

- The drop_agent command does not delete the collection agent from Oracle Audit Vault. It only disables the collection agent. The collection agent metadata is still in the database after you run the drop_agent command. If you want to re-create the collection agent, create it with a different name.
- Oracle Audit Vault displays an error if active collectors are still running in the collection agent.

Example

The following example shows how to drop a collection agent named sales_agt from Oracle Audit Vault:

```
$ avca drop_agent -agentname sales_agt

AVCA started
Dropping agent...
Agent dropped successfully.
```

6.6 generate_csr

Generates a certificate request in the format of a text file. Run this command on the Audit Vault Server.

Syntax

```
generate_csr -certdn Audit_Vault_Server_host_DN [-keysize size]
             -out certificate_request_output_file
```

Arguments

Argument	Description
`-certdn` *Audit_Vault_Server_host_DN*	Enter the distinguished name (DN) of the Audit Vault Server host
`keysize` *size*	Enter the certificate key size (in bits). Optional. Possible values are: ■ `512` ■ `1024` (default) ■ `2048`
`-out` *certificate_request_output_file*	Enter the path and name of the certificate request output file. Ensure that you have write permissions for this directory.

Usage Notes

- You must use this command to generate a certificate request. After generating the certificate request, send it to your certificate authority (CA) and get it signed and then returned as a signed certificate.

  The DN of the Audit Vault Server is typically of the following form:

  `CN=`*fully_qualified_hostname*`,OU=`*Org_Unit*`,O=`*Organization*`,ST=`*State*`,C=`*Country*

- For detailed information about generating certificate requests when setting up the HTTPS protocol for Oracle Audit Vault, see Section 5.5.

Example

The following example shows how to generate a certificate request.

```
$ avca generate_csr -certdn CN=sales_srv.us.example.com,OU=SalesReps,O=RisingDoughCo,ST=CA,C=US
-out user_certificate.cer
```

6.7 -help

Displays help information for the `AVCA` commands. Run this command on both the Audit Vault Server and Audit Vault collection agent.

Syntax

```
avca -help

avca command -help
```

Arguments

Argument	Description
command	Enter the name of an AVCA command for which you want help messages to appear

Usage Notes

If you installed the collection agent on a Microsoft Windows computer and want to run the `avca help` command from there, run it from the *ORACLE_HOME*`\`*agent_directory*`\bin` directory. For UNIX or Linux installations, ensure that you have set the appropriate environment variables before running this command. See Section 2.2 for more information.

Example

The following example shows how to display general AVCA utility Help in the Audit Vault Server home.

```
$ avca -help

  -----------------------------------------
  AVCA Usage
  -----------------------------------------
  Oracle Audit Vault Server Installation commands
      avca deploy_av -sid <sid> -dbalias <db alias> -avconsoleport <av console port>
      avca generate_csr -certdn <Audit Vault Server host DN> [-keysize 512|1024|2048]
                        -out <certificate request output file>
      avca import_cert -cert <User/Trusted certificate> [-trusted]
      avca remove_cert -certdn <Audit Vault Server host DN>
      avca secure_av -avkeystore <keystore location> -avtruststore <truststore location>
      avca secure_av -remove

  Oracle Audit Vault Configuration commands - Agent:
      avca add_agent -agentname <agent name> [-agentdesc <desc>] -agenthost <host>
      avca drop_agent -agentname <agent name>

  Oracle Audit Vault Configuration commands - Warehouse:
      avca set_warehouse_schedule -schedulename <schedule name>
      avca set_warehouse_schedule -startdate <start date> -rptintrv <repeat interval>
                                  [-dateformat <date format>]
      avca set_warehouse_retention -intrv <year-month interval>

  Oracle Audit Vault Agent Installation commands
      avca secure_agent -agentkeystore <keystore location> -avdn <DN of Audit Vault>
                        -agentdn <DN of agent>
      avca secure_agent -remove

  Oracle Audit Vault Configuration commands - Authentication:
      avca create_wallet -wrl <wallet_location>
      avca create_credential -wrl <wallet_location> -wpwd <wallet_pwd> -dbalias <db alias>
                             -usr <usr>/<pwd>

  avca -help
```

The following example shows how to display specific AVCA help for the `add_agent` command in Audit Vault.

```
$ avca add_agent -help
```

```
avca add_agent -agentname <agent name> [-agentdesc <desc>] -agenthost <host>
-----------------------------------------------
-agentname <agent name>
[-agentdesc <agent description>]
-agenthost <agent host>
-----------------------------------------------
```

This example shows how to display general AVCA utility help in the Audit Vault collection agent home.

```
$ avca -help
  -------------------------------------------
  AVCA Usage
  -------------------------------------------
  Oracle Audit Vault Agent Installation commands
      avca secure_agent -agentkeystore <keystore location>
                        -avdn <DN of Audit Vault> -agentdn <DN of agent>
      avca secure_agent -remove

  Oracle Audit Vault Configuration commands - Authentication:
      avca create_wallet -wrl <wallet_location>
      avca create_credential -wrl <wallet_location> -wpwd <wallet_pwd>
                             -dbalias <db alias> -usr <usr>/<pwd>

  avca -help
```

6.8 import_cert

Imports the specified user or trusted certificate into the wallet. Run this command on the Audit Vault Server.

Syntax

```
import_cert -cert User/Trusted_certificate [-trusted]
```

Arguments

Argument	Description
`-cert` *User/Trusted_certificate*	Enter the path and file name of the certificate to be imported into the wallet. See the usage notes.
`-trusted`	Include this argument if you want to indicate that the certificate is trusted. If it is a user certificate, then omit the `trusted` argument. Optional.

Usage Notes

- To obtain the certificate, contact the certificate authority. Place the certificate in a directory that you can easily access, for the `-cert` argument. Ensure that the certificate matches a pending certificate request in the wallet. You must import the trusted certificate for this certificate first.
- For detailed information about configuring wallets when setting up the HTTPS protocol for Oracle Audit Vault, see Section 5.5.

Example

The following example shows how to import a certificate into the wallet.

```
$ avca import_cert -cert user_certificate.cer
```

This example shows how to import a trusted certificate into the wallet.

```
$ avca import_cert -cert ca_certificate.cer -trusted
```

6.9 redeploy

Redeploys the `av.ear` file on the Audit Vault Server system or the `AVAgent.ear` file on the Audit Vault collection agent system.

Syntax

```
avca redeploy
```

Arguments

None

Usage Notes

If you installed the collection agent on a Microsoft Windows computer and want to run the `avca redeploy` command from there, run it from the *ORACLE_HOME*`\`*agent_directory*`\bin` directory. For UNIX or Linux installations, ensure that you have set the appropriate environment variables before running this command. See Section 2.2 for more information.

Example

The following example shows how to redeploy either the `av.ear` file on the Audit Vault Server system or the `AVAgent.ear` file on the Audit Vault collection agent system.

```
$ avca redeploy
```

6.10 remove_cert

Removes the specified certificate from the wallet. Run this command on the Audit Vault Server.

Syntax

```
remove_cert -cert Audit_Vault_Server_host_DN
```

Arguments

Argument	Description
`-cert` *Audit_Vault_Server_host_DN*	Enter the distinguished name (DN) of the Audit Vault Server host that was used for the `avca generate_csr` command.

Usage Notes

Oracle Audit Vault removes the certificate or key pair for the DN matching the given DN from the wallet. For example, you can use this command to remove a certificate that expires or is revoked by the CA, and replace it with a renewed certificate.

You, the Oracle Audit Vault administrator, provide the DN of the Audit Vault Server is typically of the form:

```
CN=hostname_fully_qualified,OU=Org_Unit,O=Organization,ST=State,C=Country
```

Example

The following example shows how to remove a certificate from the wallet.

```
$ avca remove_cert -hrdb.example.com CN=AV_Server_host_
DN,OU=DBSEC,O=Oracle,ST=CA,C=US
```

6.11 secure_agent

Secures the Audit Vault collection agent by enabling mutual authentication with the Audit Vault Server. Run this command on the Audit Vault collection agent. If you specify the `remove` argument, this command removes mutual authentication with the Audit Vault Server.

Syntax

```
avca secure_agent -agentkeystore keystore_location
 -avdn Audit_Vault_Server_host_DN
 -agentdn agent_DN [-agentkeystore_pwd keystore_pwd]

avca secure_agent -remove
```

Arguments

Argument	Description
`-agentkeystore` *keystore_location*	Enter the keystore file location for this collection agent. See Section 5.5.3 for more information about the keystore file.
`-avdn` *Audit_Vault_Server_host_DN*	Enter the distinguished name (DN) of the Audit Vault Server.
`-agentdn` *agent_DN*	Enter the DN of this Audit Vault collection agent.
`-remove`	Include this keyword to remove mutual authentication with the Audit Vault Server.

Usage Notes

- If you installed the collection agent on a Microsoft Windows computer, run the `avca secure_agent` command from the *ORACLE_HOME*`\`*agent_directory*`\bin` directory. For UNIX or Linux installations, set the appropriate environment variables before running this command. See Section 2.2 for more information.
- The `avca secure_agent` command prompts for the agent key password. You can bypass this prompt if the corresponding environment variable, `AVCA_AGENTKEYSTOREPWD` is set. If you enter the password, then it overrides the environment variable. This argument is provided for backward compatibility
- The keystore and certificate must be in place at the collection agent site before you execute this command.
- Use the following command to generate a keystore:

  ```
  $ORACLE_HOME/jdk/bin/keytool
  ```

- When you issue the `secure_agent` command for the specified collection agent with both the collection agent and its collectors in a running state, the collection agent and all its collectors will shut down when the agent OC4J shuts down and then restarts. You must manually restart the collection agent and its collectors.
- For detailed information about configuring mutual authentication when setting up the HTTPS protocol for Oracle Audit Vault, see Section 5.5.

Example

The following example shows how to secure the Audit Vault collection agent by enabling mutual authentication with the Audit Vault Server.

```
$ avca secure_agent -agentkeystore /tmp/agentkeystore
  -agentdn "CN=agent1, OU=development, O=oracle, L=redwoodshores, ST=ca, C=us"
  -avdn "CN=av1, OU=development, O=oracle, L=redwoodshores, ST=ca, C=us"
Enter keystore password: *******
```

The following example shows how to unsecure the Oracle Audit Vault collection agent by disabling mutual authentication with the Audit Vault Server.

```
$ avca secure_agent -remove

AVCA started
Restarting OC4J...
OC4J restarted successfully.
```

6.12 secure_av

Secures the Audit Vault Server by enabling mutual authentication with the Audit Vault collection agent. Run this command on the Audit Vault Server. If you specify the `remove` argument, this command removes mutual authentication with Audit Vault collection agent.

Syntax

```
avca secure_av -avkeystore keystore_location -avtruststore truststore_location
               [-avkeystorepwd keystore_pwd>]

avca secure_av -remove
```

Arguments

Argument	Description
`-avkeystore` *keystore_location*	Enter the keystore file location for the Audit Vault Server. By default, this file is located in the Audit Vault Server home directory. It has the file extension of `.keystore`. See Section 5.5.3 for more information about the keystore file.
`-avtruststore` *truststore_location*	Enter the **trust store** location for the Audit Vault Server. This file can be the same file as the `avkesytore` file. Ensure that this file has the CA certificates imported into it.
`-remove`	Include this keyword to remove mutual authentication with the Audit Vault collection agent

Usage Notes

- The keystore and certificate files must be in place at the Audit Vault Server before you run this command.
- Use the following command to generate a keystore:

  ```
  $ORACLE_HOME/jdk/bin/keytool
  ```

- When you issue the `avca secure_av` command, the Audit Vault Console agent OC4J restarts, which requires you to log in to Audit Vault Console again.
- The `avca secure_av` command prompts for the keystore password for the Audit Vault Server. If the corresponding environment variable, `AVCA_AVKEYSTOREPWD`, is set, then you can bypass this prompt. If you enter the password anyway, it overrides the environment variable. This argument is provided for backward compatibility.
- For detailed information about configuring mutual authentication when setting up the HTTPS protocol for Oracle Audit Vault, see Section 5.5.

Example

The following example shows how to secure the Audit Vault Server by enabling mutual authentication with the Oracle Audit Vault collection agent.

```
$ avca secure_av -avkeystore /tmp/avkeystore -avtruststore /tmp/avkeystore
Enter keystore password: password
```

The following example shows how to unsecure Audit Vault Server by disabling mutual authentication with the Audit Vault collection agent.

```
$ avca secure_av -remove

AVCA started
Stopping OC4J...
OC4J stopped successfully.
Starting OC4J...
OC4J started successfully.
Oracle Audit Vault 10g Database Control Release 10.2.3.1.0  Copyright (c)
1996,2008 Oracle Corporation.  All rights reserved.
http://av_srv.us.example.com:5700/av
Oracle Audit Vault 10g is running.
-------------------------------------

Logs are generated in directory $ORACLE_HOME/10.2.3/av_1/av/log
```

6.13 set_warehouse_retention

Controls the amount of data kept online in the data warehouse fact table. Run this command on the Audit Vault Server.

Syntax

```
avca set_warehouse_retention -intrv year_month_interval
```

Arguments

Argument	Description
`-intrv` *year_month_interval*	Enter the year-month interval in the following format: `+YY-MM`

Usage Notes

- The interval setting must be a positive value.
- Oracle Audit Vault removes the data loaded using the `avctl refresh_warehouse` command based on the warehouse retention that was using the `AVCA set_warehouse_retention` command.
- See Section 3.4 for detailed information about creating a retention period.

Example

The following example shows how to control the amount of data kept online in the data warehouse table. In this case, a time interval of 1 year is specified.

```
$ avca set_warehouse_retention -intrv +01-00

AVCA started
Setting warehouse retention period...
done.
```

6.14 set_warehouse_schedule

Sets the schedule for refreshing data from the raw audit data store to the audit data warehouse tables. Run this command on the Audit Vault Server.

Syntax

```
avca set_warehouse_schedule -schedulename schedule_name

avca set_warehouse_schedule -startdate start_date
     -rptintrv repeat_interval [-dateformat date_format]
```

Arguments

Argument	Description
`-schedulename` *schedule_name*	Enter the schedule name created using the `DBMS_SCHEDULER.create_schedule` procedure. To find the names of existing schedules created with the `DBMS_SCHEDULE` package, query the `ALL_SCHEDULER_JOBS` data dictionary view. See *Oracle Database Reference* for more information.
`-startdate` *start_date*	Enter the start date for a warehouse refresh job using the default format DD-MON-YY. To use a different format, specify the `-dateformat` argument.
`-rptintrv` *repeat_interval*	Enter the repeat interval for the schedule using the syntax used in the `DBMS_SCHEDULER.create_schedule` procedure.
`-dateformat` *date_format*	Enter the date format for the `-startdate` argument. Optional.

Usage Notes

- You can select an existing schdule that was created with the `DBMS_SCHEDULER.CREATE_SCHEDULE` PL/SQL procedure, or you can set the schedule by providing the start date and repeat interval.
- The following are error conditions:
  - The schedule name argument must be a valid schedule created using the `DBMS_SCHEDULER.CREATE_SCHEDULE` procedure.
  - The repeat interval argument must be a valid interval specification consistent with the `DBMS_SCHEDULER` package.
- See Section 3.4 for detailed information about creating a refresh schedule.

Example

The following examples show how to set the schedule for refreshing data from the raw audit data store to the audit data warehouse tables by schedule name and by start date using the `avca set_warehouse_schedule` command.

The first example uses a schedule name argument based on a valid schedule created using the `DBMS_SCHEDULER.create_schedule` procedure.

```
avca set_warehouse_schedule -schedulename daily_refresh

$ AVCA started
Set warehouse schedule...
done.
```

This example uses a start date and repeat interval argument.

```
$ avca set_warehouse_schedule -startdate 01-JUL-06 -rptintrv 'FREQ=DAILY;BYHOUR=0'

AVCA started
Set warehouse schedule...
done.
```

The following example uses a start date with a specified date format and a repeat interval argument.

```
$ avca set_warehouse_schedule -startdate 01-07-2006 -dateformat 'DD-MM-YYYY'

-rptintrv 'FREQ=DAILY;BYHOUR=0'
AVCA started
Set warehouse schedule...
done.
```

7

Audit Vault Control (AVCTL) Reference

Use the Audit Vault Control (AVCTL) command-line utility to manage various Oracle Audit Vault components (for example, checking the status of collector agents or managing the Audit Vault Data Warehouse). When you run these commands, remember the following:

- **Enter the command in lowercase letters.** The commands are case-sensitive.
- **When you open a new shell to run the command, first set the appropriate environment variables.** See Section 2.2 for more information.
- **Oracle Audit Vault creates a log file of AVCTL command activity.** See Section A.1 and Section A.2 for more information.

Table 7–1 describes the Audit Vault Control commands and where each is used, whether on the Audit Vault Server, on the Audit Vault collection agent, or in both places.

Table 7–1 Audit Vault Control Commands

Command	Where Used	Description
-help	Both	Displays help information for the AVCTL commands
load_warehouse	Server	Loads older data from the raw audit data store into the data warehouse tables for analysis
purge_warehouse	Server	Purges audit data that was reloaded into the warehouse
refresh_warehouse	Server	Refreshes the data warehouse with the data in the raw audit data store since the last refresh operation
show_agent_status	Server	Shows the status (metric) of a collection agent
show_av_status	Server	Shows the status (metric) of the Audit Vault Console
show_collector_status	Server	Shows the status (metric) of a collector
show_oc4j_status	Collection agent	Shows the status (metric) of OC4J
start_agent	Server	Starts the collection agent
start_av	Server	Starts the Audit Vault Console
start_collector	Server	Starts the collector
start_oc4j	Collection agent	Starts the agent OC4J
stop_agent	Server	Stops the collection agent
stop_av	Server	Stops the Audit Vault Console

Table 7–1 (Cont.) Audit Vault Control Commands

Command	Where Used	Description
stop_collector	Server	Stops the collector
stop_oc4j	Collection Agent	Stops the agent OC4J

Note: In an Oracle RAC environment, you must issue the AVCTL commands from the node on which Oracle Enterprise Manager resides. This is the same node on which the `av.ear` file is deployed.

If the node on which the `av.ear` file is deployed is down, deploy the `av.ear` file to another node using the `AVCA deploy_av` command.

7.1 -help

Displays help information for the `AVCTL` commands. You can run this command on both the Audit Vault Server and the Audit Vault collection agent.

Syntax

```
avctl -help

avctl command -help
```

Arguments

Argument	Description
command	Enter the name of an `AVCTL` command for which you want help to appear

Usage Notes

If you installed the collection agent on a Microsoft Windows computer and want to run the `avctl help` command from there, run it from the *ORACLE_HOME*`\`*agent_directory*`\bin` directory. For UNIX or Linux installations, set the appropriate environment variables before running this command. See Section 2.2 for more information.

Example

The following example shows how to display general `AVCTL` utility help in the Audit Vault Server home.

```
$ avctl -help

  -----------------------------------------
  AVCTL Usage
  -----------------------------------------
  Oracle Audit Vault Control commands - AV Server:
      avctl start_av [-loglevel error|warning|info|debug]
      avctl stop_av
      avctl show_av_status

  Oracle Audit Vault Control commands - Agent:
      avctl start_agent -agentname <agent name>
```

```
      avctl stop_agent -agentname <agent name>
      avctl show_agent_status -agentname <agent name>

  Oracle Audit Vault Control commands - Collector:
      avctl start_collector -collname <collector name> -srcname <source name>
      avctl stop_collector -collname <collector name> -srcname <source name>
      avctl show_collector_status -collname <collector name> -srcname <source
name>

  Oracle Audit Vault Control commands - Warehouse:
      avctl refresh_warehouse [-wait]
      avctl load_warehouse -startdate <start date> -numofdays <num of days>
[-dateformat <date format>] [-wait]
      avctl purge_warehouse -startdate <start date> -numofdays <num of days>
[-dateformat <date format>] [-wait]

  avctl -help
```

The following example shows how to display specific `AVCTL` Help for the `start_agent` command in Oracle Audit Vault.

```
$ avctl start_agent -help
  avctl start_agent -agentname <agent name>
  -------------------------------------------------
  -agentname <agent name>
  -------------------------------------------------
```

7.2 load_warehouse

Loads audit trail data from the raw audit data store after it has been removed from the warehouse repository due to the retention period that was set. Run this command on the Audit Vault Server.

Syntax

```
avctl load_warehouse -startdate start_date-numofdays num_of_days
                    [-dateformat date_format] [-wait]
```

Arguments

Argument	Description
`-startdate` *`start_date`*	Enter the start date for the audit trail data to be loaded into the data warehouse repository using the default format DD-MON-YY. To use a different format, specify the `-dateformat` argument. Use any supported Oracle Database date format. See *Oracle Database Globalization Support Guide* for more information about date formats.
`-numofdays` *`num_of_days`*	Enter the number of days' worth of audit trail data to be loaded.
`-dateformat` *`date_format`*	Enter the date format for the `-startdate` argument. Optional. Ensure that the date argument used for `startdate` matches the date format you choose. For Oracle Database supported date formats, see *Oracle Database Globalization Support Guide*.

Argument	Description
-wait	Enter the command wait for the load job to complete. If you do not specify this argument, a DBMS job is started, and the command returns immediately. Optional.

Usage Notes

- The audit records received from the value of the -startdate argument for the given number of days specified by the -numofdays argument will be loaded into the data warehouse.
- See Section 3.4 for more information about managing the Oracle Audit Vault data warehouse.

Example

The following example shows how to load the data warehouse with 10 days' worth of audit data beginning with January 1, 2004:

```
$ avctl load_warehouse -startdate 01-JAN-04 -numofdays 10

AVCTL started
Loading older audit records into warehouse...
done.
```

The following example shows how to load the data warehouse with 10 days' worth of audit data beginning with January 1, 2004 using the DD/MM/YYYY date format, and to specify that the operation wait until the previous load job completes.

```
$ avctl load_warehouse -startdate 01/01/2004 -numofdays 10 -dateformat DD/MM/YYYY -wait

AVCTL started
Loading older audit records into warehouse...
done.
```

7.3 purge_warehouse

Purges audit trail data from the warehouse repository that was previously loaded into the warehouse using the avctl load_warehouse command. Run this command on the Audit Vault Server.

Syntax

```
avctl purge_warehouse -startdate start_date -numofdays num_of_days
                    [-dateformat date_format] [-wait]
```

Arguments

Argument	Description
-startdate *start_date*	Enter the start date for the events to be removed from the data warehouse tables using the default format DD-MON-YY. To use a different format, specify the -dateformat argument. Use any supported Oracle Database date format. See *Oracle Database Globalization Support Guide* for more information about date formats.
-numofdays *num_of_days*	Enter the number of days' worth of data to be removed.

Argument	Description
`-dateformat` *`date_format`*	Specify the date format for the `-startdate` argument. Optional.
`-wait`	Optionally, enter this keyword to have the command wait for the purge job to complete. If you omit this argument, then Oracle Audit Vault starts the job and then returns to the command prompt immediately. Optional.

Usage Notes

- The audit records received from the `-startdate` argument for the given number of days specified by the `-numofdays` argument will be removed from the data warehouse tables.
- Only data loaded using the `avctl load_warehouse` command can be purged using the `avctl purge_warehouse` command. The data loaded using the `avctl refresh_warehouse` command is removed automatically based on the warehouse duration specified using the `avca set_warehouse_retention` command.
- See Section 3.4 for more information about managing the Oracle Audit Vault data warehouse.

Example

The following example shows how to purge 10 days' worth of data from the data warehouse beginning with January 1, 2004:

```
$ avctl purge_warehouse -startdate 01-JAN-04 -numofdays 10

AVCTL started
Purging older audit records from warehouse...
done.
```

The following example shows how to purge 10 days' worth of data from the data warehouse beginning with January 1, 2004 and to specify that the operation wait until the previous purge job completes:

```
$ avctl purge_warehouse -startdate 01-JAN-04 -numofdays 10 -wait

AVCTL started
Purging older audit records from warehouse...
Waiting for purge to complete...
done.
```

The following example shows how to purge 10 days' worth of data from the data warehouse beginning with January 1, 2004 using the date format of DD/MM/YYYY.

```
$ avctl purge_warehouse -startdate 01/01/2004 -numofdays 10 -dateformat DD/MM/YYYY

AVCTL started
Purging older audit records from warehouse...
done.
```

7.4 refresh_warehouse

Refreshes the data warehouse repository with the data from the raw audit data store since the last refresh operation. Run this command on the Audit Vault Server.

Syntax

```
avctl refresh_warehouse [-wait]
```

Arguments

Argument	Description
`-wait`	Enter this keyword to specify that the command wait for the refresh job to complete. If you omit this argument, Oracle Audit Vault starts the job and then returns to the command prompt immediately. Optional.

Usage Notes

- The last refresh operation could have been an explicit refresh using this command or a scheduled refresh based on the schedule set using the `avca set_warehouse_schedule` command.
- See Section 3.4 for more information about managing the Oracle Audit Vault data warehouse.

Example

The following example shows how to refresh the data warehouse:

```
$ avctl refresh_warehouse

AVCTL started
Refreshing warehouse...
done.
```

This example shows how to specify that the refresh operation wait until the previous refresh job completes before refreshing the data warehouse:

```
$ avctl refresh_warehouse -wait

AVCTL started
Refreshing warehouse...
Waiting for refresh to complete...
done.
```

7.5 show_agent_status

Shows the status (metric) of a collection agent. Run this command on the Audit Vault Server.

Syntax

```
avctl show_agent_status -agentname agent_name
```

Arguments

Argument	Description
`-agentname` *`agent_name`*	Enter the collection agent (by collection agent name).

Usage Notes

If you installed the collection agent on a Microsoft Windows computer, run the `avctl show_agent_status` command from the *`ORACLE_HOME`*`\`*`agent_directory`*`\bin`

directory. For UNIX or Linux installations, ensure that you have set the appropriate environment variables before running this command. See Section 2.2 for more information.

Example

The following example shows the collection agent status for the `sales_agt` agent:

```
$ avctl show_agent_status -agentname SALES_AGT

AVCTL started
Getting agent metrics...
--------------------------------
Agent is running
--------------------------------
Metrics retrieved successfully.
```

7.6 show_av_status

Shows the Audit Vault Console status or the metric of the Audit Vault Server. Run this command on the Audit Vault Server.

Syntax

```
avctl show_av_status
```

Arguments

None

Usage Notes

When the Audit Vault Console becomes inaccessible, issue this command to determine its status.

Example

The following example shows the Audit Vault Console status:

```
$ avctl show_av_status

AVCTL started
Oracle Audit Vault 10g Database Control Release 10.2.3.1.0  Copyright (c) 1996,
 2008 Oracle Corporation.  All rights reserved.
http://hrdb.us.example.com:5570/av
Oracle Audit Vault 10g is running.
-----------------------------------
Logs are generated in directory /oracle/product/10.2.3/av_1/av/log
```

7.7 show_collector_status

Shows the status (metric) of a collector. Run this command on the Audit Vault Server.

Syntax

```
avctl show_collector_status -collname collector_name -srcname source_name
```

Arguments

Argument	Description
`-collname` *`collector_name`*	Enter the target collector (by collector name).
`-srcname` *`source_name`*	Enter the name of the source database to which this collector belongs.

Usage Notes

None

Example

The following example shows the collector status for the DBAUD_Collector collector:

```
$ avctl show_collector_status -collname DBAUD_Collector
                              -srcname RODSRC1.US.EXAMPLE.COM

AVCTL started
Getting collector metrics...
--------------------------------
Collector is running
Records per second  =  0.00
Bytes per second  =  0.00
--------------------------------
```

7.8 show_oc4j_status

Shows the OC4J status (metric). Run this command on the Audit Vault collection agent.

Syntax

```
avctl show_oc4j_status
```

Arguments

None

Usage Notes

If you installed the collection agent on a Microsoft Windows computer, run the `avctl show_oc4j_status` command from the *`ORACLE_HOME`*`\`*`agent_directory`*`\bin` directory. For UNIX or Linux installations, set the appropriate environment variables before running this command. See Section 2.2 for more information.

Example

The following example shows the OC4J status for when it is running and when it is not running:

```
$ avctl show_oc4j_status

AVCTL started
-----------------------------------
OC4J is running
-----------------------------------
```

This example shows the OC4J status for when it is not running:

```
$ avctl stop_oc4j
```

```
AVCTL started
Stopping OC4J...
OC4J stopped successfully.

$ avctl show_oc4j_status
AVCTL started
-----------------------------------
OC4J is not running
-----------------------------------
```

7.9 start_agent

Starts the specified collection agent. Run this command on the Audit Vault Server.

Syntax

```
avctl start_agent -agentname agent_name
```

Arguments

Argument	Description
`-agentname` *agent_name*	Enter the collection agent (by collection agent name) to be started.

Usage Notes

- On successful completion of this command, the collection agent is moved to a `RUNNING` state. If an error is encountered, the collection agent is moved to an `ERROR` state.
- Oracle Audit Vault accepts audit records only from collection agents in the `RUNNING` state.
- If you set the `NLS_LANG` environment value before running the `avctl start_oc4j` command in the Audit Vault Agent shell and running the `avctl start_agent` command or `avctl start_collector` command in the Audit Vault Server shell, the `avctl start_collector` command can accept a multibyte source name or collector name.

Example

The following example shows how to start the collection agent in Oracle Audit Vault:

```
$ avctl start_agent -agentname sales_agt

AVCTL started
Starting Agent...
Agent started successfully.
```

7.10 start_av

Starts the Audit Vault Console. Run this command on the Audit Vault Server.

Syntax

```
avctl start_av [-loglevel level]
```

Arguments

Argument	Description
`-loglevel` *level*	Optionally, enter the desired level of logging from the following options. ■ `error`: Logs only error messages ■ `warning`: Logs both warning and error messages ■ `info`: Logs informational and error messages (default) ■ `debug`: Logs debug, error, warning, and informational messages

Usage Notes

This command executes the `emctl start dbconsole` command.

Example

The following example shows how to start the Audit Vault Console:

```
$ avctl start_av

AVCTL started
Starting OC4J...
OC4J started successfully.
Oracle Audit Vault 10g Database Control Release 10.2.3.1.0  Copyright (c)
1996,2008 Oracle Corporation.  All rights reserved.
http://shobeen.us.example.com:5700/av
Oracle Audit Vault 10g is running.
------------------------------------
Logs are generated in directory /oracle/product/10.2.3/av_1/av/log
```

7.11 start_collector

Starts the collector. Run this command on the Audit Vault Server.

Syntax

```
avctl start_collector -collname collector_name -srcname source_name
```

Arguments

Argument	Description
`-collname` *collector_name*	Enter the name of the collector to be started.
`-srcname` *source_name*	Enter the name of the source database to which the collector (specified in the `-collname` argument) belongs.

Usage Notes

- On successful completion of this command, Oracle Audit Vault sets the collector to a `RUNNING` state. If an error is encountered, the collector is set to an `ERROR` state. If you receive a message saying that the collector is not in a `RUNNING` state, ensure that the agent has been started. Use the `avctl start_agent` command to start the agent, as described in Section 7.9.
- Oracle Audit Vault accepts audit records only from collectors in the `RUNNING` state.

- If you set the NLS_LANG environment value before running the avctl start_oc4j command in the Audit Vault Agent shell and running the avctl start_agent command or avctl start_collector command in the Audit Vault Server shell, the avctl start_collector command can accept a multibyte source name or collector name.

Example

The following example shows how to start the collector in Oracle Audit Vault:

```
$ avctl start_collector -collname REDO_Collector -srcname ORCLSRC1.EXAMPLE.COM

AVCTL started
Starting Collector...
Collector started successfully.
```

7.12 start_oc4j

Starts the agent OC4J. Run this command on the Audit Vault collection agent.

Syntax

```
avctl start_oc4j [-loglevel level] [-maxheapsize maximum_heap_memory]
```

Arguments

Argument	Description
-loglevel *level*	Optionally, enter the desired level of logging from the following options: ■ error: Logs only error messages ■ warning: Logs both warning and error messages ■ info: Logs informational and error messages (default) ■ debug: Logs debug, error, warning, and informational messages
-maxheapsize *maximum_heap_memory*	Enter the maximum amount of heap memory allocated for the Java OC4J process. The default value is 1000 MB. Optional. This setting enables you to fine-tune the OC4J performance based on the size of your Oracle Audit Vault installation. Check the size of the physical memory of the computer on which the Audit Vault collection agents are installed before setting this value.

Usage Notes

- If you installed the collection agent on a Microsoft Windows computer, run the avctl start_oc4j command from the *ORACLE_HOME**agent_directory*\bin directory. For UNIX or Linux installations, set the appropriate environment variables before running this command. See Section 2.2 for more information.
- If you set the NLS_LANG environment value before running the avctl start_oc4j command in the Audit Vault Agent shell and running avctl start_agent command or avctl start_collector command in the Audit Vault

Server shell, it will ensure that the `avctl start_collector` command can accept with a multibyte source name or collector name.

Example

The following example shows how to start OC4J. For the `-maxheapsize` setting, include `M` (for megabytes) as shown below. You can set it for other sizes, such as `G` for gigabyte, but in most cases, you should set it in megabytes.

```
$ avctl start_oc4j -maxheapsize 500M

AVCTL started
Starting OC4J...
OC4J started successfully.
```

7.13 stop_agent

Stops the collection agent. Run this command on the Audit Vault Server.

Syntax

```
avctl stop_agent -agentname agent_name
```

Arguments

Argument	Description
`-agentname` *`agent_name`*	Enter the collection agent (by collection agent name) to be stopped.

Usage Notes

- This command will first stop all collectors running at this collection agent, and then stop the collection agent itself.
- On successful completion of this command, the collection agent and its collectors are moved to a `STOPPED` state.
- f an error is encountered, Oracle Audit Vault sets the collection agent to an `ERROR` state. Oracle Audit Vault accepts audit records only from collection agents in the `RUNNING` state.

Example

The following example shows how to stop the collection agent in Oracle Audit Vault:

```
$ avctl stop_agent -agentname sales_agt

AVCTL started
Stopping Agent...
Agent stopped successfully.
```

7.14 stop_av

Stops the Audit Vault Console. Run this command on the Audit Vault Server.

Syntax

```
avctl stop_av
```

Arguments

None

Usage Notes

Oracle Audit Vault includes Enterprise Management Database Control as part of the user interfaces. When you issue the `stop_av` commend, it not only shuts down the Audit Vault Console, but it also stops Enterprise Management Database Control by executing the `emctl stop dbconsole` command. You do not need to issue the `emctl` command separately.

Example

The following example shows how to stop the Audit Vault Console:

```
$ avctl stop_av

AVCTL started
Stopping OC4J...
OC4J stopped successfully.
```

7.15 stop_collector

Stops the collector. Run this command on the Audit Vault Server.

Syntax

```
avctl stop_collector -collname collector_name -srcname source_name
```

Arguments

Argument	Description
`-collname` *`collector_name`*	Enter the name of the collector to be stopped.
`-srcname` *`source_name`*	Enter the name of the source database to which the collector (specified in the `-collname` argument) belongs.

Usage Notes

- On successful completion of this command, Oracle Audit Vault moves the collector a `STOPPED` state.
- If an error is encountered, Oracle Audit Vault sets collector to an `ERROR` state.
- Oracle Audit Vault accepts audit records only from collectors in the `RUNNING` state.

Example

The following example shows how to stop the collector in Oracle Audit Vault:

```
$ avctl stop_collector -collname STREAMSCOLLECTOR

-srcname ORCL.REGRESS.RDBMS.DEV.US.ORACLE.COM
AVCTL started
Stopping Collector...
Collector stopped successfully.
```

7.16 stop_oc4j

Stops the agent OC4J. Run this command on the Audit Vault collection agent.

Syntax

```
avctl stop_oc4j
```

Arguments

None

Usage Notes

If you installed the collection agent on a Microsoft Windows computer, run the `avctl stop_oc4j` command from the *ORACLE_HOME*`\`*agent_directory*`\bin` directory. For UNIX or Linux installations, set the appropriate environment variables before running this command. See Section 2.2 for more information.

Example

The following example shows how to stop OC4J:

```
$ avctl stop_oc4j

AVCTL started
Stopping agent OC4J...
OC4J stopped successfully.
```

8

Audit Vault Oracle Database (AVORCLDB) Utility Commands

Use the Audit Vault Oracle Database (AVORCLDB) command-line utility to manage the relationship between Oracle Audit Vault and an Oracle source database and collector. When you run these commands, remember the following:

- **Enter the command in lowercase letters.** The commands are case-sensitive.
- **When you open a new shell to run the command, first set the appropriate environment variables.** See Section 2.2 for more information.
- **Oracle Audit Vault creates a log file of AVORCLDB command activity.** See Section A.1 and Section A.2 for more information.

Table 8–1 describes the AVORCLDB commands and where each is used, whether on the Audit Vault Server, on the Audit Vault collection agent, or in both places.

Table 8–1 AVORCLDB Commands

Command	Where Used?	Description
add_collector	Server	Adds a collector to Oracle Audit Vault
add_source	Server	Registers an audit source with Oracle Audit Vault
alter_collector	Server	Alters the attributes of a collector
alter_source	Server	Alters the attributes of a source
drop_collector	Server	Drops a collector from Oracle Audit Vault
drop_source	Server	Drops a source database from Oracle Audit Vault
-help	Both	Displays help information for the AVORCLDB commands
setup	Collection agent	Adds the source user credentials to the wallet, creates a database alias in the wallet for the source user, verifies the connection to the source using the wallet, and updates the `tnsnames.ora` file
verify	Both	Verifies that the source is compatible with the collectors that are specified for setup

8.1 avorcldb

The AVORCLDB command-line utility, which you use to configure an Oracle database with Oracle Audit Vault.

Syntax

```
avorcldb command -help
```

```
avorcldb command [options] arguments
```

Arguments

Argument	Description
command	Enter one of the commands listed in Table 8–1 on page 8-1.
arguments	Enter one or more of the AVORCLDB command arguments.
-help	Displays help information for the AVORCLDB commands.

Usage Notes

Issuing an AVORCLDB command generates the following log file: $ORACLE_HOME/av/log/avorcldb.log.

8.2 add_collector

Adds a collector for the given Oracle source database to Audit Vault. Oracle Audit Vault verifies the source database for the collector requirements. Run this command on the Audit Vault Server.

Syntax

```
avorcldb add_collector -srcname srcname
-agentname agentname -colltype [OSAUD,DBAUD,REDO]
[-collname collname] [-desc desc]
[-av host:port:service] [-instname instname] [-orclhome orclhome]
```

Arguments

Argument	Description
-srcname *srcname*	Enter the source database name for which the collector is to be added. Remember that the source database name is case-sensitive.
-agentname *agentname*	Enter the name of the collection agent that was created when you ran the avca add_agent command.
-colltype *colltype*	Enter the collector type to be added. ■ DBAUD ■ OSAUD ■ REDO See Table 1–4 on page 1-6 for more information about the collector types.
-collname *collname*	Create a name for the collector. Optional. If you do not create a name, Oracle Audit Vault names the collector *colltype*_Collector (for example, OSAUD_Collector for the OSAUD collector type).
-desc *desc*	Enter a brief description of the collector. Optional.

Argument	Description
`-av` *`host:port:service`*	Enter the connection information for Oracle Audit Vault used for the database link from the source database to Oracle Audit Vault. You must include this argument if the `-colltype` argument is REDO; otherwise, this argument is optional.
`-instname` *`instname`*	Enter the instance name of Audit Vault Oracle RAC installation. You must include this argument if you are adding multiple OSAUD collectors, that is, one collector for each database instance.
`-orclhome` *`orclhome`*	Enter the Oracle home of the source database.You must include this argument if the `-colltype` argument is OSAUD; otherwise, this argument is optional. See the usage notes.

Usage Notes

- Run any collector-specific preparation scripts before you execute the `avca add_collector` command.
- On Microsoft Windows systems, specifying the OSAUD collector type automatically includes the event log and XML audit trails.
- When specifying the value for the `-orclhome` argument, enter the value as either a quoted string using a backslash. For example:

  ```
  -orclhome "c:\app\oracle\product\10.2.3\av_1"
  ```

  Alternatively, enter it as a nonquoted string using a slash. For example:

  ```
  -orclhome c:/app/oracle/product/10.2.3/av_1
  ```

- There is a 2 GB audit file size limit for the OSAUD collector to be able to collect audit records from audit trails stored in files, which includes the `SYSLOG`, `.AUD`, and `.XML` files. If the file size is greater than 2 GB, then the OSAUD collector ignores all audit records beyond 2 GB. To control the size of the operating system audit trail and select the audit trail type to set, set the `DBMS_AUDIT_MGMT.OS_FILE_MAX_SIZE` property and the `DBMS_AUDIT_MGMT.AUDIT_TRAIL_TYPE` type by using the `DBMS_AUDIT_MGMT.SET_AUDIT_TRAIL_PROPERTY` PL/SQL procedure. See Section 14.4.11 for more information.

Example

The following example shows how to add an OSAUD collector to Oracle Audit Vault on Linux and UNIX platforms in an Oracle Real Application Clusters (Oracle RAC) installation using the `-instname` argument.

```
$ avorcldb add_collector -srcname source1db.example.com
-agentname Agent1 -colltype OSAUD -instname av01
-orclhome /u01/app/oracle/product/10.2.0/db_1

source SOURCE1DB.EXAMPLE.COM verified for OS File Audit Collector collector
Adding collector...
Collector added successfully.
collector successfully added to Audit Vault

remember the following information for use in avctl
Collector name (collname): OSAUD_Collector
```

This example shows how to add a DBAUD collector to Oracle Audit Vault:

```
$ avorcldb add_collector -srcname source1db.example.com

-agentname Agent1 -colltype DBAUD
source SOURCE1DB.DOMAIN.COM verified for Aud$/FGA_LOG$ Audit Collector collector

Adding collector...
Collector added successfully.
collector successfully added to Audit Vault

remember the following information for use in avctl
Collector name (collname): DBAUD_Collector
```

The next example shows how to add a REDO collector to Oracle Audit Vault.

```
$ avorcldb add_collector -srcname source1db.example.com
-agentname Agent1 -colltype REDO
-av system1.example.com:1521:av

source SOURCE1DB.EXAMPLE.COM verified for REDO Log Audit Collector collector
Adding collector...
Collector added successfully.
collector successfully added to Audit Vault

remember the following information for use in avctl
Collector name (collname): REDO_Collector
initializing REDO Collector
setting up APPLY process on Audit Vault server
setting up CAPTURE process on source database
```

8.3 add_source

Registers an Oracle source database with Oracle Audit Vault for audit data consolidation. Run this command on the Audit Vault Server.

Syntax

```
avorcldb add_source -src host:port:service
      [-srcname srcname] [-desc desc] [-agentname agentname]
```

Arguments

Argument	Description
`-src` *host*`:`*port*`:`*service*	Enter the source database connection information: host name, port number, and service ID (SID), separated by a colon. If you are unsure of this connection information, run the `lsnrctl status` command on the computer where you installed the source database.
`-srcname` *srcname*	Enter the name of the source database. Remember that the source database name is case-sensitive. Optional. If you do not specify this argument, Oracle Audit Vault uses the global database name.You can check this name by selecting from the `GLOBAL_NAME` data dictionary view in SQL*Plus. For example: `SQL> SELECT * FROM GLOBAL_NAME;`
`-desc` *desc*	Enter a brief description of the source database. Optional.

Argument	Description
-agentname *agentname*	Create a name for a collection agent. Optional. However, you must specify an agent name if auditors plan to configure policy management using the Audit Vault Console.

Usage Notes

- The global database name of the source database is used as the source name in Oracle Audit Vault.
- The avorcldb add_source command prompts for the source user name and password. This user account must exist on the source database.

  To find this user, query the SESSION_PRIVS and SESSION_ROLES data dictionary views. The source user should have the privileges and roles that are listed in the zarsspriv.sql file, such as the CREATE DATABASE LINK privilege and DBA role.

  If the AVORCLDB_SRCUSR environment variable is set to this user account and password, then you can bypass the Enter Source user name and Enter Source password prompts. If you do specify these values, they override the environment variable.
- You must specify the -agentname *agentname* parameter so that auditors can configure policy management using the Audit Vault Console.

Example

The following example shows how to register a source with Oracle Audit Vault.

```
$ avorcldb add_source -src hrdb.example.com:1521:orcl -agentname agent1
Enter Source user name: username
Enter Source password: password

Adding source...
Source added successfully.
source successfully added to Audit Vault

remember the following information for use in avctl
Source name (srcname): RDBMSRC1.US.EXAMPLE.COM
Storing user credentials in wallet...
Create credential oracle.security.client.connect_string3
done.
Mapping Source to Agent...
```

8.4 alter_collector

Modifies the attributes of an Oracle Database collector. Run this command on the Audit Vault Server.

Syntax

```
avorcldb alter_collector -srcname srcname -collname collname
      [attrname=attrvalue...attrname=attrvalue]
```

Arguments

Argument	Description
`-srcname` *`srcname`*	Enter the name of the source database to which this collector belongs. Remember that the source database name is case-sensitive.
`-collname` *`collname`*	Enter the name of the collector to be modified.
`attrname=attrvalue`	Enter the attribute pair (attribute name, new attribute value) for mutable collector attributes for this collector type. This argument is optional. Separate multiple pairs by a space on the command line.

Usage Notes

You can modify one or more collector attributes at a time. The following tables list the collector attributes (parameters) by collector type, whether the parameter is mutable, and its default value. See Section 3.3 for a description of these attributes.

Table 8–2 describes the DBAUD collector attributes.

Table 8–2 DBAUD Collector Attributes

Parameter	Description	Mutable	Default Value
AUDAUDIT_ACTIVE_SLEEP_TIME	The amount of active sleep time (in milliseconds) for the DBAUD process when the last retrieval actually did retrieve records.	Yes	1000 milliseconds
AUDAUDIT_AUDIT_VAULT_ALIAS	The alias name for the Audit Vault Server.	No	NULL
AUDAUDIT_DELAY_TIME	The amount of delay time (in seconds) for the DBAUD process.	Yes	20 seconds
AUDAUDIT_MAX_PROCESS_RECORDS	The maximum number of records after which the collector commits records to the raw audit data store and generates minor recovery context. In fine-grained auditing (FGA_LOG$) and 9.*x* sources, the collector might need to delay this until the record with the higher timestamp is retrieved. A valid value is an integer value from 10 to 10000.	Yes	1000 records
AUDAUDIT_SLEEP_TIME	The amount of sleep time (in milliseconds) for the DBAUD process. For example, if it is now 10:00:00 AM, the collector will retrieve the records with the timestamps that are less than 9:59:40. However, the next time the collector will only retrieve records with the timestamps of 9:59:40 or higher. The assumption is that within 20 seconds after the timestamp is assigned to the record, the record would be visible (retrievable). This attribute is used only for time-based retrieval in fine-grained auditing (FGA_LOG$) on 9.*x* source databases. In Oracle Audit Vault, time-based retrieval is used for all retrievals.	Yes	5000 milliseconds
AUDAUDIT_SORT_POLICY	The audit data sort policy. This attribute is not implemented. It was deprecated for Oracle Audit Vault Release 10.2.3.	Yes	NULL
AUDAUDIT_SOURCE_ALIAS	The alias name for the audit data source	No	NULL

Table 8–3 describes the OSAUD collector attributes.

Table 8–3 OSAUD Collector Attributes

Parameter	Description	Mutable	Default Value
OSAUDIT_AUDIT_VALUE_ALIAS	The alias name for the Audit Vault Server	No	NULL
OSAUDIT_CHANNEL_TYPE	The channel type being used by the collector This attribute is not implemented. It was deprecated in Oracle Audit Vault Release 10.2.3.	No	NULL
OSAUDIT_DEFAULT_FILE_DEST[1]	The default directory for Oracle Database operating system audit files. This directory contains mandatory audit record files.	Yes	$ORACLE_HOME/rdbms/audit
OSAUDIT_FILE_DEST	The directory for the Oracle Database operating system audit files. This directory contains SYS and regular audit record files.	Yes	$ORACLE_HOME/admin/*DB_UNIQUE_NAME*/adump
OSAUDIT_MAX_PROCESS_RECORDS	The maximum number of records to be processed during each call to process the collector. A valid value is an integer value from 10 to 10000.	Yes	10000
OSAUDIT_MAX_PROCESS_TIME	The maximum processing time for each call to process the collector (in centiseconds). A valid value is an integer value from 10 to 10000.	Yes	600 centiseconds
OSAUDIT_NLS_CHARSET	The NLS character set of the data source	Yes	WE8ISO8859P1
OSAUDIT_NLS_LANGUAGE	The NLS language of the data source	Yes	AMERICAN
OSAUDIT_NLS_TERRITORY	The NLS territory of the data source	Yes	AMERICA
OSAUDIT_NT_ORACLE_SID	The Oracle SID name on Microsoft Windows systems	Yes	NULL
OSAUDIT_RAC_INSTANCE_ID	The instance ID in an Oracle RAC environment	Yes	1.0
OSAUDIT_SOURCE_ALIAS	The alias or connection string to the source database	Yes	NULL
OSAUDIT_SYSLOG_FILE	The syslog file name and location, if other than the default as indicated in the etc/syslog.conf file. Setting this parameter to a valid syslog file name overrides the default setting.	Yes	NULL

[1] To avoid collecting duplicate operating system audit trail records, do not set the attribute value for the OSAUDIT_DEFAULT_FILE_DEST attribute and the OSAUDIT_FILE_DEST attribute such that the values, although different, resolves to the same directory.

Table 8–4 describes the REDO collector attributes.

Table 8–4 REDO Collector Attributes

Parameter	Description	Mutable	Default Value
AV.DATABASE.NAME	The Oracle Audit Vault database name	No	NULL
STRCOLL_DBPORT	The port number of the audit data Oracle source database	Yes	NULL
STRCOLL_DBSERVICE	The service name of the audit data Oracle source database	No	NULL
STRCOLL_HEARTBEAT_TIME	The time, in seconds, between events for monitoring the status of the Audit Vault REDO collection system	Yes	60 seconds
STRCOLL_SRCADM_ALIAS	The alias name for the audit data source	No	NULL
STRCOLL_SRCADM_NAME	The name of the audit data source database	No	NULL

On Microsoft Windows systems, if the path value for the OSAUDIT_DEFAULT_FILE_DEST attribute is set incorrectly using backslashes, use the Audit Vault Console to log in as the Audit Vault administrator and connect as AV_ADMIN, click **Configuration**, click **Collector**, select the **OSAUD_Collector** name, then click **Edit** and edit the value for this attribute using slashes instead of backslashes. When finished, click **OK** to save your changes.

Example

The following example shows how to alter the AUDAUDIT_DELAY_TIME attribute for the DBAUD_Collector collector in Oracle Audit Vault:

```
$ avorcldb alter_collector -srcname hrdb.example.com -collname DBAUD_Collector
AUDAUDIT_DELAY_TIME=60

Altering collector...
Collector altered successfully.
```

8.5 alter_source

Modifies the attributes of an Oracle source database. Run this command on the Audit Vault Server.

Syntax

```
avorcldb alter_source -srcname srcname
      [attrname=attrvalue...attrname=attrvalue]
```

Arguments

Argument	Description
-srcname *srcname*	Enter the name of the source database to be modified. Remember that the source database name is case-sensitive.
attrname=attrvalue	Enter the pair (attribute name, new attribute value) for the mutable source attributes of this source to be modified. Optional. Separate multiple pairs by a space on the command line.

Usage Notes

Table 8–5 lists source attributes that you can specify for the *attrname=attrvalue* argument.

Table 8–5 Source Attributes

Parameter	Description	Mutable	Default Value
HOSTIP	The Internet protocol address of the host system on which the source database resides	Yes	NULL
VERSION	The source database version	Yes	NULL
DESCRIPTION	The description for this source database	Yes	NULL
DB_SERVICE	A new audit data source database service name	Yes	NULL

Table 8–5 (Cont.) Source Attributes

Parameter	Description	Mutable	Default Value
PORT	A new port number for this system where the source database audit data resides	Yes	NULL
GLOBAL_DATABASE_NAME	The new global database name	Yes	NULL

Example

The following example shows how to alter the PORT attribute for the source database named hrdb.example.com in Oracle Audit Vault:

```
$ avorcldb alter_source -srcname hrdb.example.com PORT=1522
Altering source...
Source altered successfully.
```

8.6 drop_collector

Disables (but does not remove) a collector from Oracle Audit Vault. Run this command from the Audit Vault Server.

Syntax

avorcldb drop_collector -srcname *srcname* -collname *collname*

Arguments

Argument	Description
-srcname *srcname*	Enter the name of the source database to which the collector (specified in the -collname argument) belongs. Remember that the source database name is case-sensitive.
-collname *collname*	Enter the name of the collector to be dropped from Oracle Audit Vault.

Usage Notes

The drop_collector command does not delete the collector from Oracle Audit Vault. It only disables the collector. The collector metadata is still in the database after you run the drop_collector command. If you want to recreate the collector, create it with a different name.

Example

```
$ avorcldb drop_collector -srcname hrdb.example.com -collname DBAud_Collector

Dropping collector...
Collector dropped successfully.
```

8.7 drop_source

Disables (but does not remove) a source database from Oracle Audit Vault. Run this command on the Audit Vault Server.

Syntax

avorcldb drop_source -srcname *srcname*

Arguments

Argument	Description
`-srcname` *srcname*	Enter the name of the source database to be dropped from Oracle Audit Vault. Remember that the source database name is case-sensitive.

Usage Notes

- The `drop_source` command does not delete the source database from Oracle Audit Vault. It only disables the source database definition in Oracle Audit Vault. The source database metadata is still in the database after you run the `drop_source` command. If you want to re-create the source database definition, create it with a different name.
- You cannot drop a source database if there are any active collectors for this source. You must drop all collectors associated with the source database before you can run the `drop_source` command on it.

Example

The following example shows how to drop the source named `hrdb.example.com` from Oracle Audit Vault:

```
$ avorcldb drop_source -srcname hrdb.example.com

Dropping source...
Source dropped successfully.
```

8.8 -help

Displays help information for the `AVORCLDB` commands. Run this command on either the Audit Vault Server and the Audit Vault collection agent.

Syntax

```
avorcldb -help

avorcldb command -help
```

Arguments

Argument	Description
command	Enter the name of an `AVORCLDB` command for which you want help to appear

Usage Notes

None

Example

The following example shows how to display general `AVORCLDB` utility help in Oracle Audit Vault:

```
$ avorcldb -help
```

The following example shows how to display specific `AVORCLDB` help for the `add_source` command in the Audit Vault Server home shell.

```
$ avorcldb add_source -help

  avorcldb add_source command

    add_source
          -src <host:port:service> [-srcusr <usr>/<pwd>]
         [-srcname <srcname>] [-desc <desc>] [-agentname <agentname>]

  Purpose: The source is added to Audit Vault. The global DB Name
       of the source database is used as the Source Name in Audit Vault.

  Arguments:
       -src        : Source DB connection information
       -srcusr     : Optional source user name and password. Will be prompted.
       -srcname    : Optional name of source, default : <global_dbname>
       -desc       : Optional description of the source
       -agentname  : Optional agent name to configure policy management

  Examples:
     avorcldb add_source -src lnxserver:4523:hrdb.domain.com
       -desc 'HR Database'
```

8.9 setup

Adds the source user credentials to the wallet, creates a database alias in the wallet for the source user, verifies the connection to the source using the wallet, and updates the `tnsnames.ora` file. You also can use this command to change the source user credentials in the wallet after these credentials have been changed in the source database. Run this command on the Audit Vault collection agent.

Syntax

```
avorcldb setup -srcname srcname
```

Arguments

Argument	Description
`-srcname` *srcname*	Enter the name of the source database. Remember that the source database name is case-sensitive.

Usage Notes

- If you installed the collection agent on a Microsoft Windows computer, run the `avorcldb setup` command from the *ORACLE_HOME*`\`*agent_directory*`\bin` directory. For UNIX or Linux installations, set the appropriate environment variables before running this command. See Section 2.2 for more information.
- The `avorcldb setup` command prompts for the source user name and password. This user account must exist on the source database.

  To find the privileges and roles granted to this user, query the `SESSION_PRIVS` and `SESSION_ROLES` data dictionary views. The source user should have the privileges and roles that are listed in the `zarsspriv.sql` file, such as the `CREATE DATABASE LINK` privilege and `DBA` role.

  If the `AVORCLDB_SRCUSR` environment variable is set to this user account and password, then you can bypass the `Enter Source user name` and `Enter`

`Source password` prompts. If you do specify these values, they override the environment variable.

Example

The following example configures the REDO and OSAUD collectors.

```
$ avorcldb setup -srcname hrdb.example.com
Enter Source user name: username
Enter Source password: password

adding credentials for user srcuser_ora for connection [SRCDB1]
Storing user credentials in wallet...
Create credential oracle.security.client.connect_string3
done.
updated tnsnames.ora with alias [SRCDB1] to source database
verifying SRCDB1 connection using wallet
```

To change the source user name password in the wallet in the Audit Vault collection agent home, use the following setup command, where the source name is `orcl1` and the source user name is `srcuser_ora`.

```
$ avorcldb setup -srcname orcl1
Enter Source user name: srcuser_ora
Enter Source password: password

adding credentials for user srcuser_ora for connection [SRCDB1]
Storing user credentials in wallet...
Create credential oracle.security.client.connect_string3
done.
updated tnsnames.ora with alias [SRCDB1] to source database
verifying SRCDB1 connection using wallet
```

8.10 verify

Verifies that the source is compatible for setting up the specified collectors. Run this command on either the Audit Vault Server or the Audit Vault collection agent.

Syntax

```
avorcldb verify -src host:port:service
                -colltype [OSAUD,DBAUD,REDO,ALL]
```

Arguments

Argument	Description
`-src` *host*`:`*port*`:`*service*	Enter the source database connection information: host name, port number, and service name, separated by a colon. Typically, the host is the fully qualified domain name or IP address of the server on which the source database is running, and the port number is 1521. If you are unsure of the host and port number, run the `lsnrctl status` command on the computer where you installed the source database.

Argument	Description
`-colltype` *colltype*	Enter one of the following collector types: ■ ALL ■ DBAUD ■ OSAUD ■ REDO See Table 1–4 on page 1-6 for more information about the collector types.

Usage Notes

- If you installed the collection agent on a Microsoft Windows computer and want to run the `avorcldb verify` command from there, run it from the *ORACLE_HOME*`\`*agent_directory*`\bin` directory. For UNIX or Linux installations, set the appropriate environment variables before running this command. See Section 2.2 for more information.
- The `avorcldb verify` command prompts for the source user name and password. This user account must exist on the source database. To find this user, query the `SESSION_PRIVS` and `SESSION_ROLES` data dictionary views. The source user should have the privileges and roles that are listed in the `zarsspriv.sql` file, such as the `CREATE DATABASE LINK` privilege and `DBA` role.
- If the `AVORCLDB_SRCUSR` environment variable is set to this user account, then you can bypass the `Enter Source user name` and `Enter Source password` prompts. If you do specify these values, they override the environment variable.

Example

The following example verifies that the source is compatible with the OSAUD, DBAUD, and REDO collectors on a Linux or UNIX system.

```
$ avorcldb verify -src hrdb.example.com:1521:orcl -colltype ALL
Enter Source user name: username
Enter Source password: password

source HRDB.EXAMPLE.COM verified for OS File Audit Collector collector
source HRDB.EXAMPLE.COM verified for Aud$/FGA_LOG$ Audit Collector collector
source HRDB.EXAMPLE.COM verified for REDO Log Audit Collector collector
```

9

Audit Vault Microsoft SQL Server (AVMSSQLDB) Utility Commands

Use the Audit Vault SQL Server Database (AVMSSQLDB) command-line utility to manage the relationship between Oracle Audit Vault and a Microsoft SQL Server source database and collector. When you run these commands, remember the following:

- **Enter the command in lowercase letters.** The commands are case-sensitive.
- **When you open a new shell to run the command, first set the appropriate environment variables.** See Section 2.2 for instructions.
- **Oracle Audit Vault creates a log file of AVMSSQLDB command activity.** See Section A.1 and Section A.2 for more information.

Table 9–1 describes the AVMSSQLDB commands and where each is used, whether on the Audit Vault Server, on the Audit Vault collection agent, or in both places.

Table 9–1 AVMSSQLDB Commands

Command	Where Used?	Description
add_collector	Server	Adds a collector to Oracle Audit Vault
add_source	Server	Registers an audit source with Oracle Audit Vault
alter_collector	Server	Alters the attributes of a collector
alter_source	Server	Alters the attributes of a source
drop_collector	Server	Drops a collector from Oracle Audit Vault
drop_source	Server	Drops a source from Oracle Audit Vault
-help	Both	Displays help information for the AVMSSQLDB commands
setup	Collection agent	Adds the source user credentials to the wallet, creates a database alias in the wallet for the source user, and verifies the connection to the source using the wallet
verify	Both	Verifies that the source is compatible with the collectors

9.1 avmssqldb

The AVMSSQLDB command-line utility, which you use to configure a Microsoft SQL Server database with Oracle Audit Vault.

Syntax

```
avmssqldb command -help
```

```
avmssqldb command [options] arguments
```

Arguments

Argument	Description
command	Enter one of the commands listed in Table 9–1 on page 9-1.
arguments	Enter one or more of the AVMSSQLDB command arguments.
-help	Displays help information for the AVMSSQLDB commands.

Usage Notes

Issuing an AVMSSQLDB command generates the following log file: $ORACLE_HOME/av/log/mssqldb-%g.log. The *%g* is a generation number that starts from 0 (zero) and increases once the file size reaches the 100 MB limit.

9.2 add_collector

Adds a collector for the given SQL Server source database to Oracle Audit Vault. Oracle Audit Vault verifies the source database for the collector requirements. Run this command on the Audit Vault Server.

Syntax

```
avmssqldb add_collector -srcname srcname -agentname agentname
         [-collname collname] [-desc desc]
```

Arguments

Argument	Description
-srcname srcname	Enter the name of the source database for which the collector is to be added. Remember that the source database name is case-sensitive.
-agentname *agentname*	Create a name for the agent that will use the MSSQLDB collector.
-collname *collname*	Create a name for the MSSQLDB collector. Optional. If you do not create a name, Oracle Audit Vault names the collector MSSQLCollector.
-desc *desc*	Enter a brief description of the collector. Optional.

Usage Notes

- Run any collector-specific preparation scripts before you execute the avmssqldb add_collector command.
- The avmssqldb add_collector command prompts for the source user name and password. This user account must exist on the source database.

Example

The following example shows how to add the MSSQLDB collector to Oracle Audit Vault.

```
$ avmssqldb add_collector -srcname mssqldb4 -agentname agent1
Enter a username :source_user_name
```

```
Enter a password : password

***** Collector Added Successfully*****
```

9.3 add_source

Registers a SQL Server source database with Oracle Audit Vault for audit data consolidation. Run this command on the Audit Vault Server.

Syntax

```
avmssqldb add_source -src host:port -srcname srcname
[-desc desc]
```

Arguments

Argument	Description
`-src host:port`	Enter the source database connection information: host name and port number, separated by a colon. Typically, the host is the fully qualified domain name or IP address of the server on which the SQL Server source database is running, and the port number is 1433.
`-srcname srcname`	Create a name for the source database connection. Remember that the source database name is case-sensitive. Oracle Audit Vault uses this name to connect to the Microsoft SQL Server source database.
`-desc desc`	Enter a brief description for the source database. Optional.

Usage Notes

The `avmssqldb add_source` command prompts for the source user name and password. This user account must exist on the source database. See the example.

Example

The following example shows how to register a source with Oracle Audit Vault.

```
$ avmssqldb add_source -src mssqlserver:1433 -srcname mssqldb4 -desc 'HR Database'
Enter a username :source_user_name
Enter a password : password

***** Source Verified *****
***** Source Added Successfully *****
```

9.4 alter_collector

Modifies the attributes of an MSSQLDB collector. Run this command on the Audit Vault Server.

Syntax

```
avmssqldb alter_collector -srcname srcname -collname collname
      [attrname=attrvalue...attrname=attrvalue]
```

Arguments

Argument	Description
`-srcname` *srcname*	Enter the name of the source database to which this collector belongs. Remember that the source database name is case-sensitive.
`-collname` *collname*	Enter the name of the collector to be modified.
attrname=attrvalue	Enter the attribute pair (attribute name, new attribute value) for mutable collector property and attributes for this collector type. This argument is optional. Separate multiple pairs by a space on the command line.

Usage Notes

- You can modify the collector `DESCRIPTION` property and one or more attributes at a time. Table 9–2 lists the collector attributes (parameters), whether the parameter is mutable, the default value, and a brief description of the attribute.

Table 9–2 MSSQLDB Collector Attributes

Parameter	Mutable	Default Value	Description
`DESCRIPTION`	Yes	`NULL`	The description for this collector
`dbconnection`	No	`1`	Number of connections to the database
`AUDIT_C2_FLAG`	Yes	`1`	Whether C2 logs can be collected by the MSSQLDB collector. Values can be 0 or 1.
`AUDIT_SERVERSIDE_TRACES_FLAG`	Yes	`1`	Whether server side trace logs can be collected by the MSSQLDB collector. Values can be 0 or 1. See the usage notes.
`AUDIT_EVENT_LOG_FLAG`	Yes	`1`	Whether events logs can be collected by the MSSQLDB collector. Values can be 0 or 1.
`C2_TRACE_FILEPATH`	Yes	`NULL`	The C2 trace file path. See the usage notes.
`SERVERSIDE_TRACE_FILEPATH`	Yes	`NULL`	The value for server-side trace file path. See the usage notes.
`DELAY_TIME`	Yes	`20000`	The delay time (in milliseconds) of the collector
`NO_OF_RECORDS`	Yes	`1000`	The maximum number of records to be fetched by the collector. This attribute is mutable.

- For SQL Server 2000 source databases only, the trace file (`.trc`) audit trail is not released to the collector until either the file reaches its maximum file size and another trace file is created, or the source database is shut down and restarted.
- If the server side `TRACEPATH` parameter or the `C2_TRACE_FILEPATH` parameter is set to null, and the `AUDIT_SERVERSIDE` traces flag is set to `true`, then the

collector queries the SQL Server database for active trace files and collects audit data from them.

- For the C2_TRACE_FILEPATH and the SERVERSIDE_TRACE_FILEPATH parameters, the value for the path can be of the form *Drive*:*Directory*....*File Prefix**.

Example

The following example shows how to alter the NO_OF_RECORDS attribute and the collector description for the MSSQLCollector collector in ORacle Audit Vault:

```
$ avmssqldb alter_collector -srcname mssqldb4 -collname MSSQLCollector NO_OF_
RECORDS=1500 DESCRIPTION="MSSQLDB collector 45" SERVERSIDE_TRACE_
FILEPATH="c:\SQLAuditFile*

***** Collector Altered Successfully *****
```

9.5 alter_source

Modifies the attributes of a SQL Server source database. Run this command on the Audit Vault Server.

Syntax

```
avmssqldb alter_source -srcname sourcename
          [attrname=attrvalue...attrname=attrvalue]
```

Arguments

Argument	Description
-srcname *sourcename*	Enter the name of the source database to be modified. Remember that the source database name is case-sensitive.
attrname=attrvalue	Enter the attribute pair (attribute name, new attribute value) for mutable source properties and attributes for this source type. This argument is optional. Separate multiple pairs by a space on the command line.

Usage Notes

Table 9–3 lists the source attributes, a brief description of the attribute, whether the attribute is mutable, and the default value. You can modify one or more source attributes at a time.

Table 9–3 Source Attributes

Attribute	Description	Mutable	Default Value
SOURCETYPE	The source type name for this source database. The default name is MSSQLDB.	No	NULL
NAME	The name for this source database	No	NULL
HOST	The source database host name	No	NULL
HOSTIP	The source database host IP address	No	NULL
VERSION	The source database version	Yes	NULL
DESCRIPTION	The description for this source database	Yes	NULL

Table 9–3 (Cont.) Source Attributes

Attribute	Description	Mutable	Default Value
PORT	A new port number for this system where the source database audit data resides	Yes	None

Example

The following example shows how to alter the DESCRIPTION attribute for the source database named mssqldb4 in Oracle Audit Vault:

```
$ avmssqldb alter_source -srcname mssqldb4 DESCRIPTION="HR Database"

***** Source Altered Successfully *****
```

9.6 drop_collector

Disables (but does not remove) an MSSQLDB collector from Oracle Audit Vault. Run this command from the Audit Vault Server.

Syntax

```
avmssqldb drop_collector -srcname srcname -collname collname
```

Arguments

Argument	Description
-srcname *srcname*	Enter the name of the source database to which the collector (specified in the -collname argument) belongs. Remember that the source database name is case-sensitive.
-collname *collname*	Enter the name of the collector to be dropped from Oracle Audit Vault.

Usage Notes

The drop_collector command does not delete the collector from Oracle Audit Vault. It only disables the collector. The collector metadata is still in the database after you run the drop_collector command. If you want to recreate the collector, create it with a different name.

Example

The following example shows how to drop a collector named MSSQLCollector from Oracle Audit Vault:

```
$ avmssqldb drop_collector -srcname mssqldb4 -collname MSSQLCollector

***** Collector Dropped Successfully *****
```

9.7 drop_source

Disables (but does not remove) a SQL Server source database from Oracle Audit Vault. Run this command on the Audit Vault Server.

Syntax

```
avmssqldb drop_source -srcname srcname
```

Arguments

Argument	Description
`-srcname` *srcname*	Enter the source (by source name) to be dropped from Oracle Audit Vault. Remember that the source database name is case-sensitive.

Usage Notes

- The `drop_source` command does not delete the source database from Oracle Audit Vault. It only disables the source database definition in Oracle Audit Vault. The source database metadata is still in the database after you run the `drop_source` command. If you want to re-create the source database definition, create it with a different name.
- You cannot drop a source database if it has any active collectors for this source database. You must drop all collectors associated with the source database before you can run the `drop_source` command on it.

Example

The following example shows how to drop the source named `mssqldb4` from Oracle Audit Vault:

```
$ avmssqldb drop_source -srcname mssqldb4

***** Drop Source Successfully *****
```

9.8 -help

Displays help information for the `AVMSSQLDB` commands. Run this command on either the Audit Vault Server and the Audit Vault collection agent.

Syntax

```
avmssqldb -help

avmssqldb command -help
```

Arguments

Argument	Description
command	Enter the name of an `AVMSSQLDB` command for which you want help to appear.

Usage Notes

None

Example

The following example shows how to display general `AVMSSQLDB` utility help in Oracle Audit Vault:

```
avmssqldb -help
```

The following example shows how to display specific `AVMSSQLDB` help for the `add_source` command in the Audit Vault Server home shell.

```
$ avmssqldb add_source -help
  avmssqldb add_source command

    add_source
          -src <host:port>
          -srcname <srcname> [-desc <desc>]

  Purpose: The source is added to Audit Vault.

  Arguments:
       -src        : Source DB connection information to coolect audit data.
       -srcname    : Name of a source
       -desc       : Optional description of the source

  Examples:
     avmssqldb add_source -src 10.105.118.91:1433
        -desc 'source for admin databases' -srcname mssource
```

9.9 setup

Adds the SQL Server source user credentials to the wallet, creates a database alias in the wallet for the source user, and verifies the connection to the source using the wallet. You also can use this command to change the source user credentials in the wallet after these credentials have been changed in the source database. Run this command on the Audit Vault collection agent.

Syntax

`avmssqldb setup -srcname` *`srcname`*

Arguments

Argument	Description
`-srcname` *`srcname`*	Enter the name of the source database. Remember that the source database name is case-sensitive.

Usage Notes

- If you installed the collection agent on a Microsoft Windows computer, run the `avmssqldb setup` command from the *`ORACLE_HOME`*`\`*`agent_directory`*`\bin` directory. For UNIX or Linux installations, set the appropriate environment variables before running this command. See Section 2.2 for more information.
- The `avmssqldb setup` command prompts for the source user name and password. This user account must exist on the source database.

Example

```
$ avmssqldb setup -srcname mssqldb4
Enter a username : source_user_name
Enter a password : password

***** Credentials Successfully added *****
```

9.10 verify

Verifies that a SQL Server source database is compatible for setting up the specified collector. Run this command on either the Audit Vault Server or the Audit Vault collection agent.

Syntax

```
avmssqldb verify -src host:port
```

Arguments

Argument	Description
`-src` *host*`:`*port*	Enter the source database connection information: host name and port number, separated by a colon. Typically, the host is the fully qualified domain name or IP address of the server on which the SQL Server source database is running, and the port number is 1433.

Usage Notes

- The `avmssqldb verify` command checks the following:
  - Whether the version of the SQL Server database is supported: SQL Server 2000 or SQL Server 2005
  - Whether the source user has the required privileges in the source database that is to be registered with Oracle Audit Vault
  - Whether auditing (C2 auditing and server-side trace auditing) is enabled in the source database
- If you installed the collection agent on a Microsoft Windows computer and want to run the `avmssqldb verify` command from there, run it from the *ORACLE_HOME*`\`*agent_directory*`\bin` directory. For UNIX or Linux installations, set the appropriate environment variables before running this command. See Section 2.2 for more information.
- The `avmssqldb verify` command prompts for the source user name and password. This user account must exist on the source database.

Example

The following example verifies that the source is compatible with the MSSQLDB collector on Windows.

```
$ avmssqldb verify -src 192.0.2.1:4523
Enter a username : source_user_name
Enter a password : password

***** Source Verified *****
```

10

Audit Vault Sybase ASE (AVSYBDB) Utility Commands

Use the Audit Vault Sybase Database (AVSYBDB) command-line utility to manage the relationship between Oracle Audit Vault and a Sybase ASE source database and collector. When you run these commands, remember the following:

- **Enter the command in lowercase letters.** The commands are case-sensitive.
- **When you open a new shell to run the command, first set the appropriate environment variables.** See Section 2.2 for instructions.
- **Oracle Audit Vault creates a log file of AVSYBDB command activity.** See Section A.1 and Section A.2 for more information.

Table 10–1 describes the AVSYBDB commands and where each is used, whether on the Audit Vault Server, on the Audit Vault collection agent, or in both places.

Table 10–1 AVSYBDB Commands

Command	Where Used?	Description
add_collector	Server	Adds a collector to Oracle Audit Vault
add_source	Server	Registers an audit source with Oracle Audit Vault
alter_collector	Server	Alters the attributes of a collector
alter_source	Server	Alters the attributes of a source
drop_collector	Server	Drops a collector from Oracle Audit Vault
drop_source	Server	Drops a source from Oracle Audit Vault
-help	Both	Displays help information for the AVSYBDB commands
setup	Collection agent	Adds the source user credentials to the wallet, creates a database alias in the wallet for the source user, and verifies the connection to the source using the wallet
verify	Both	Verifies that the source is compatible with the collectors

10.1 avsybdb

The AVSYBDB command-line utility, which you use to configure a Sybase ASE database with Oracle Audit Vault.

Syntax

```
avsybdb command -help
```

```
avsybdb command [options] arguments
```

Arguments

Argument	Description
command	Enter one of the commands listed in Table 10–1 on page 10-1.
arguments	Enter one or more of the AVSYBDB command arguments.
-help	Displays help information for the AVSYBDB commands.

Usage Notes

Issuing an AVSYBDB command generates the following log file: $ORACLE_HOME/av/log/sybdb-%g.log. The *%g* is a generation number that starts from 0 (zero) and increases once the file size reaches the 100 MB limit.

10.2 add_collector

Adds a SYBDB collector for a Sybase ASE source database to Oracle Audit Vault. Oracle Audit Vault verifies the source database for the collector requirements. Run this command on the Audit Vault Server.

Syntax

```
avsybdb add_collector -srcname srcname -agentname agentname
        [-collname collname] [-desc desc]
```

Arguments

Argument	Description
-srcname *srcname*	Enter the name of the source database for which the collector is to be added. Remember that the source database name is case-sensitive. Typically, the host is the fully qualified domain name or IP address of the server on which the Sybase ASE source database is running, and the port number is 5000.
-agentname *agentname*	Create a name for the agent that will use the SYBDB collector.
-collname *collname*	Create a name for the SYBDB collector. Optional. If you do not create a name, Oracle Audit Vault names the collector SybaseCollector.
-desc *desc*	Enter a brief description of the collector. Optional.

Usage Notes

- Run any collector-specific preparation scripts before you execute the avsybdb add_collector command.
- The avsybdb add_collector command prompts for the source user name and password. This user account must exist on the source database.

Example

The following example shows how to add a SYBDB collector to Oracle Audit Vault on Linux and UNIX platforms.

```
$ avsybdb add_collector -srcname sybdb4 -agentname agent1
Enter a username : source_user_name
Enter a password : password

***** Collector Added Successfully*****
```

10.3 add_source

Registers a Sybase ASE source database with Oracle Audit Vault for audit data consolidation. Run this command on the Audit Vault Server.

Syntax

```
avsybdb add_source -src host:port -srcname srcname [-desc desc]
```

Arguments

Argument	Description
`-src` *host*`:`*port*	Enter the source database connection information: host name and port number, separated by a colon. Typically, the host is the fully qualified domain name or IP address of the server on which the Sybase ASE source database is running, and the port number is 5000.
`-srcname` *srcname*	Create a name to associate with this source database. Remember that the source database name is case-sensitive. Oracle Audit Vault uses this name to connect to the Sybase ASE source database.
`-desc` *desc*	Enter a brief description of the source database. Optional.

Usage Notes

The `avsybdb add_source` command prompts for the source user name and password. This user account must exist on the source database.

Example

The following example shows how to register a source with Oracle Audit Vault.

```
$ avsybdb add_source -src lnxserver:5000 -srcname sybdb4 -desc 'HR Database'
Enter a username : source_user_name
Enter a password : password

***** Source Verified *****
***** Source Added Successfully *****
```

10.4 alter_collector

Modifies the attributes of a SYBDB collector. Run this command on the Audit Vault Server.

Syntax

```
avsybdb alter_collector -srcname srcname -collname collname
       [attrname=attrvalue...attrname=attrvalue]
```

Arguments

Argument	Description
`-srcname` *srcname*	Enter the name of the source database to which this collector belongs. Remember that the source database name is case-sensitive.
`-collname` *collname*	Enter the name of the collector to be modified.
attrname=attrvalue	Enter the attribute pair (attribute name, new attribute value) for mutable collector property and attributes for this collector type. This argument is optional. Separate multiple pairs by a space on the command line.

Usage Notes

You can modify one or more collector attributes at a time. Table 10–2 lists the collector attributes (parameters), whether the parameter is mutable, its default value, and a brief description.

Table 10–2 SYBDB Collector Attributes

Parameter	Mutable	Default Value	Description
`DESCRIPTION`	Yes	`NULL`	The description for this collector
`dbconnection`	No	`1`	Number of connections to the database
`DELAY_TIME`	Yes	`20000`	The delay time (in milliseconds) of the collector
`NO_OF_RECORDS`	Yes	`1000`	The maximum number of records to be fetched by the collector

Example

The following example shows how to alter the `NO_OF_RECORDS` attribute and the collector description for the `SybaseCollector` collector in Oracle Audit Vault:

```
$ avsybdb alter_collector -srcname sybdb4 -collname SybaseCollector
NO_OF_RECORDS=1500 DESCRIPTION="Sybase collector 45"

***** Collector Altered Successfully *****
```

10.5 alter_source

Modifies the attributes of the Sybase ASE source database. Run this command on the Audit Vault Server.

Syntax

```
avsybdb alter_source -srcname srcname
      [attrname=attrvalue...attrname=attrvalue]
```

Arguments

Argument	Description
`-srcname` *srcname*	Enter the name of the source database to be modified. Remember that the source database name is case-sensitive.

Argument	Description
attrname=attrvalue	Enter the attribute pair (attribute name, new attribute value) for mutable source properties and attributes for this source type. This argument is optional. Separate multiple pairs by a space on the command line. See Table 10–3 for more information.

Usage Notes

Table 10–3 lists the source database attributes, a brief description of the attribute, whether the attribute is mutable, and the default value. You can modify one or more source attributes at a time.

Table 10–3 Source Attributes

Attribute	Description	Mutable	Default Value
SOURCETYPE	The source type name for this source database. The default name is SYBDB.	No	NULL
NAME	The name for this source database	No	NULL
HOST	The source database host name	No	NULL
HOSTIP	The source database host IP address	No	NULL
VERSION	The source database version	Yes	NULL
DESCRIPTION	A new description for this source database	Yes	NULL
PORT	A new port number for this system where the source database audit data reside	Yes	None

Example

The following example shows how to alter the DESCRIPTION attribute for the source database named sybdb4 in Oracle Audit Vault:

```
$ avsybdb alter_source -srcname sybdb4 DESCRIPTION="HR Database"

***** Source Altered Successfully *****
```

10.6 drop_collector

Disables (but does not remove) a SYBDB collector from Oracle Audit Vault. Run this command from the Audit Vault Server. The drop_collector command does not delete the collector from Oracle Audit Vault; instead, it disables the collector. Therefore, you can neither add a collector by the same name as the one that was dropped nor enable a collector that has been dropped.

Syntax

```
avsybdb drop_collector -srcname srcname -collname collname
```

Arguments

Argument	Description
-srcname *srcname*	Enter the name of the source database to which the collector (specified in the -collname argument) belongs. Remember that the source database name is case-sensitive.

Argument	Description
`-collname` *`collname`*	Enter the name of the collector to be dropped from Oracle Audit Vault.

Usage Notes

The `drop_collector` command does not delete the collector from Oracle Audit Vault. It only disables the collector. The collector metadata is still in the database after you run the `drop_collector` command. If you want to recreate the collector, create it with a different name.

Example

The following example shows how to drop the collector named `SybaseCollector` from Oracle Audit Vault:

```
$ avsybdb drop_collector -srcname sybdb4 -collname SybaseCollector

***** Collector Dropped Successfully *****
```

10.7 drop_source

Disables (but does not remove) a Sybase ASE source database from Oracle Audit Vault. Run this command on the Audit Vault Server.

Syntax

```
avsybdb drop_source -srcname srcname
```

Arguments

Argument	Description
`-srcname` *`srcname`*	Enter the name of the source database to be dropped from Oracle Audit Vault. Remember that the source database name is case-sensitive.

Usage Notes

- The `drop_source` command does not delete the source database from Oracle Audit Vault. It only disables the source database definition in Oracle Audit Vault. The source database metadata is still in the database after you run the `drop_source` command. If you want to re-create the source database definition, create it with a different name.
- You cannot drop a source database if there are any active collectors for this source. You must drop all collectors associated with the source database before you can run the `drop_source` command on it.

Example

The following example shows how to drop the source named `sybdb4` from Oracle Audit Vault:

```
$ avsybdb drop_source -srcname sybdb4

***** Drop Source Successfully *****
```

10.8 -help

Displays help information for the AVSYBDB commands. Run this command on either the Audit Vault Server or the Audit Vault collection agent.

Syntax

```
avsybdb -help

avsybdb command -help
```

Arguments

Argument	Description
command	Enter the name of an AVSYBDB command for which you want help to appear.

Usage Notes

None

Example

The following example shows how to display general AVSYBDB utility help in Oracle Audit Vault:

```
avsybdb -help
```

The following example shows how to display specific AVSYBDB Help for the add_source command in the Audit Vault Server home shell.

```
$ avsybdb add_source -help
  avsybdb add_source command

    add_source
          -src <host:port> -srcname <srcname>
          [-desc <desc>]

  Purpose: The source is added to Audit Vault.

  Arguments:
       -src        : Source DB connection information
       -srcname    : Name of a source
       -desc       : Optional description of the source

  Examples:
     avsybdb add_source -src lnxserver:5000
        -desc 'HR Database'
```

10.9 setup

Adds the Sybase ASE source user credentials to the wallet, creates a database alias in the wallet for the source user, and verifies the connection to the source using the wallet. You also can use this command to change the source user credentials in the wallet after these credentials have been changed in the source database. Run this command on the Audit Vault collection agent.

Syntax

```
avsybdb setup -srcname srcname
```

Arguments

Argument	Description
`-srcname` *srcname*	Enter the name of the source database. Remember that the source database name is case-sensitive.

Usage Notes

- If you installed the collection agent on a Microsoft Windows computer, run the `avsybdb setup` command from the *ORACLE_HOME*`\`*agent_directory*`\bin` directory. For UNIX or Linux installations, set the appropriate environment variables before running this command. See Section 2.2 for more information.
- The `avsybdb setup` command prompts for the source user name and password. This user account must exist on the source database.

Example

```
$ avsybdb setup -srcname sybdb4
Enter a username : source_user_name
Enter a password : password

***** Credentials Successfully added *****
```

10.10 verify

Verifies that the Sybase ASE source database is compatible for setting up the specified collectors. Run this command on either the Audit Vault Server or the Audit Vault collection agent.

Syntax

```
avsybdb verify -src host:port
```

Arguments

Argument	Description
`-src` *host*`:`*port*	Enter the source database connection information: host name and port number, separated by a colon. Typically, the host is the fully qualified domain name or IP address of the server on which the Sybase ASE source database is running, and the port number is 5000.

Usage Notes

- The `avsybdb verify` command checks the following:
  - Whether the version of the database is supported: Sybase ASE 15.0.2 or Sybase ASE 12.5.4
  - Whether the source user has the required privileges in the source database that is to be registered with Oracle Audit Vault
  - Whether auditing is enabled in the source database
  - Whether the operating system on which the source database is running is supported

- If you installed the collection agent on a Microsoft Windows computer and want to run the `avsybdb verify` command from there, run it from the *ORACLE_HOME*`\`*agent_directory*`\bin` directory. For UNIX or Linux installations, set the appropriate environment variables before running this command. See Section 2.2 for more information.
- The `avsybdb verify` command prompts for the source user name and password. This user account must exist on the source database.

Example

The following example verifies that the source is compatible with the SYBDB collector on a Linux or UNIX system.

```
$ avsybdb verify -src 192.0.2.7:5000
Enter a username : source_user_name
Enter a password : password

***** Source Verified *****
```

11

Audit Vault IBM DB2 (AVDB2DB) Utility Commands

Use the Audit Vault IBM DB2 Database (AVDB2DB) command-line utility to manage the relationship between Oracle Audit Vault an IBM DB2 source database and B2DB collector. When you run these commands, remember the following:

- **Enter the command in lowercase letters.** The commands are case-sensitive.
- **When you open a new shell to run the command, first set the appropriate environment variables.** See Section 2.2 for instructions.
- **Oracle Audit Vault creates a log file of AVDB2DB command activity.** See Section A.1 and Section A.2 for more information.

Table 11–1 describes the AVDB2DB commands and where each is used, whether on the Audit Vault Server, on the Audit Vault collection agent, or in both places.

Table 11–1 AVDB2DB Commands

Command	Where Used?	Description
add_collector	Server	Adds a collector to Oracle Audit Vault
add_source	Server	Registers an audit source with Oracle Audit Vault
alter_collector	Server	Alters the attributes of a collector
alter_source	Server	Alters the attributes of a source
drop_collector	Server	Drops a collector from Oracle Audit Vault
drop_source	Server	Drops a source from Oracle Audit Vault
-help	Both	Displays help information for the AVDB2DB commands
setup	Collection agent	Adds the source user credentials to the wallet, creates a database alias in the wallet for the source user, and verifies the connection to the source using the wallet
verify	Both	Verifies that the source is compatible with the collectors

11.1 avdb2db

The AVDB2DB command-line utility, which you use to configure an IBM DB database with Oracle Audit Vault.

Syntax

```
avdb2db command -help
```

```
avdb2db command [options] arguments
```

Arguments

Argument	Description
command	Enter one of the commands listed in Table 11–1 on page 11-1.
arguments	Enter one or more of the AVDB2DB command arguments.
-help	Displays help information for the AVDB2DB commands

Usage Notes

Issuing an AVDB2DB command generates the following log file: $ORACLE_HOME/av/log/db2db-%g.log. The *%g* is a generation number that starts from 0 (zero) and increases once the file size reaches the 100 MB limit.

11.2 add_collector

Adds a collector for the given IBM DB2 source database to Oracle Audit Vault. Oracle Audit Vault verifies the source database for the collector requirements. Run this command on the Audit Vault Server.

Syntax

```
avdb2db add_collector -srcname srcname -agentname agentname
        [-collname collname] [-desc desc]
```

Arguments

Argument	Description
-srcname *srcname*	Enter the source database name for which the collector is to be added. Remember that the source database name is case-sensitive. Typically, the host is the fully qualified domain name or IP address of the server on which the IBM DB2 source database is running, and the port number is 50000.
-agentname *agentname*	Create a name for the agent that will use the DB2DB collector.
-collname *collname*	Create a name for the DB2DB collector. Optional. If you do not create a name, Oracle Audit Vault names the collector DB2_Coll.
-desc *desc*	Enter a brief description of the collector. Optional.

Usage Notes

- Run any collector-specific preparation scripts before you execute the avdb2db add_collector command.
- The avdb2db add_collector command prompts for a user name and password. This user account must have privileges to run the IBM DB2 db2audit command (for example, a user who has the sysadmin privilege).

Example

The following example shows how to add an DB2DB collector to Oracle Audit Vault on Linux and UNIX platforms.

```
$ avdb2db add_collector -srcname db2db4 -agentname agent1
Enter a username : source_user_name
Enter a password : password

***** Collector Added Successfully*****
```

11.3 add_source

Registers an IBM DB2 source database with Oracle Audit Vault for audit data consolidation. Run this command on the Audit Vault Server.

Syntax

```
avdb2db add_source -src host:port -srcname srcname [-desc desc]
```

Arguments

Argument	Description
`-src` *host*`:`*port*	Enter the source database connection information: host name and port number, separated by a colon. Typically, the host is the fully qualified domain name or IP address of the server on which the IBM DB2 source database is running, and the port number is 50000.
`-srcname` *srcname*	Create a name to associate with this source database. Remember that the source database name is case-sensitive. Oracle Audit Vault uses this name to connect to the IBM DB2 source database.
`-desc` *desc*	Enter a brief description of the source database. Optional.

Usage Notes

The `avdb2db add_source` command prompts for a user name and password. This user account must have privileges to run the IBM DB2 `db2audit` command (for example, a user who has the `sysadmin` privilege).

Example

The following example shows how to register a source with Oracle Audit Vault.

```
$ avdb2db add_source -src lnxserver:50000 -srcname db2db4 -desc 'HR Database'
Enter a username : source_user_name
Enter a password : password

***** Source Verified *****
***** Source Added Successfully *****
```

11.4 alter_collector

Modifies the attributes of a DB2DB collector. Run this command on the Audit Vault Server.

Syntax

```
avdb2db alter_collector -srcname srcname -collname collname
```

```
      [attrname=attrvalue...attrname=attrvalue]
```

Arguments

Argument	Description
`-srcname srcname`	Enter the name of the source database to which this collector belongs. Remember that the source database name is case-sensitive.
`-collname collname`	Enter the name of the collector to be modified.
`attrname=attrvalue`	Enter the attribute pair (attribute name, new attribute value) for mutable collector property and attributes for this collector type. This argument is optional. Separate multiple pairs by a space on the command line.

Usage Notes

You can modify one or more collector attributes at a time. Table 11–2 lists the collector attributes (parameters), whether the parameter is mutable, its default value, and a brief description.

Table 11–2 DB2DB Collector Attributes

Parameter	Mutable	Default Value	Description
DESCRIPTION	Yes	NULL	The description for this collector
dbconnection	No	1	Number of connections to the database
DELAY_TIME	Yes	20000	The delay time (in milliseconds) of the collector
NO_OF_RECORDS	Yes	1000	The maximum number of records to be fetched by the collector
SINGLE_FILEPATH	Yes	NULL	The location of the directory where the DB2 collector will look for files to collect audit records from, or the location to which the DB2 extraction utility writes the text files

Example

The following example shows how to alter the NO_OF_RECORDS attribute and the collector description for the DB2Collector collector in Oracle Audit Vault:

```
$ avdb2db alter_collector -srcname db2db4 -collname DB2Collector
NO_OF_RECORDS=1500 DESCRIPTION="IBM DB2 collector 9"

***** Collector Altered Successfully *****
```

11.5 alter_source

Modifies the attributes of an IBM DB2 source database. Run this command on the Audit Vault Server.

Syntax

```
avdb2db alter_source -srcname srcname
      [attrname=attrvalue...attrname=attrvalue]
```

Arguments

Argument	Description
`-srcname srcname`	Enter the name of the source database to be modified. Remember that the source database name is case-sensitive.
`attrname=attrvalue`	Enter the attribute pair (attribute name, new attribute value) for mutable source properties and attributes for this source type. This argument is optional. Separate multiple pairs by a space on the command line. See Table 11–3 for more information.

Usage Notes

Table 11–3 lists the source database attributes, a brief description of the attribute, whether the attribute is mutable, and the default value. You can modify one or more source attributes at a time.

Table 11–3 Source Attributes

Attribute	Description	Mutable	Default Value
SOURCETYPE	The source type name for this source database. The default name is DB2DB.	No	NULL
NAME	The name for this source database.	No	NULL
HOST	The source database host name.	No	NULL
HOSTIP	The source database host IP address.	No	NULL
VERSION	The source database version.	Yes	NULL
DESCRIPTION	A new description for this source database.	Yes	NULL
PORT	A new port number for this system where the source database audit data resides	Yes	None

Example

The following example shows how to alter the DESCRIPTION attribute for the source database named db2db4 in Oracle Audit Vault:

```
$ avdb2db alter_source -srcname db2db4 DESCRIPTION="HR Database"

***** Source Altered Successfully *****
```

11.6 drop_collector

Disables (but does not remove) a DB2DB collector from Oracle Audit Vault. Run this command from the Audit Vault Server.

Syntax

```
avdb2db drop_collector -srcname srcname -collname collname
```

Arguments

Argument	Description
`-srcname` *`srcname`*	Enter the name of the source database to which the collector (specified in the `-collname` argument) belongs. Remember that the source database name is case-sensitive.
`-collname` *`collname`*	Enter the name of the collector to be dropped from Oracle Audit Vault.

Usage Notes

The `drop_collector` command does not delete the collector from Oracle Audit Vault. It only disables the collector. The collector metadata is still in the database after you run the `drop_agent` command. If you want to recreate the collector, create it with a different name.

Example

The following example shows how to drop a collector named `DB2Collector` from Oracle Audit Vault:

```
$ avdb2db drop_collector -srcname db2db4 -collname DB2Collector

***** Collector Dropped Successfully *****
```

11.7 drop_source

Disables (but does not remove) an IBM DB2 source database from Oracle Audit Vault. Run this command on the Audit Vault Server.

Syntax

`avdb2db drop_source -srcname` *`srcname`*

Arguments

Argument	Description
`-srcname` *`srcname`*	Enter the name of the source database to be dropped from Oracle Audit Vault. Remember that the source database name is case-sensitive.

Usage Notes

- The `drop_source` command does not delete the source database from Oracle Audit Vault. It only disables the source database definition in Oracle Audit Vault. The source database metadata is still in the database after you run the `drop_source` command. If you want to re-create the source database definition, create it with a different name.
- You cannot drop a source database if there are any active collectors for this source. You must drop all collectors associated with the source database before you can run the `drop_source` command on it.

Example

The following example shows how to drop the source named `db2db4` from Oracle Audit Vault:

```
$ avdb2db drop_source -srcname db2db4

***** Drop Source Successfully *****
```

11.8 -help

Displays help information for the AVDB2DB commands. Run this command on either the Audit Vault Server or the Audit Vault collection agent.

Syntax

```
avdb2db -help

avdb2db command -help
```

Arguments

Argument	Description
command	Enter the name of an AVDB2DB command for which you want help to appear.

Usage Notes

None

Example

The following example shows how to display general AVDB2DB utility help in Oracle Audit Vault:

```
avdb2db -help
```

The following example shows how to display specific AVDB2DB help for the add_source command in the Audit Vault Server home shell.

```
$ avdb2db add_source -help
  avdb2db add_source command

    add_source
          -src <host:port> -srcname <srcname>
          [-desc <desc>]

  Purpose: The source is added to Audit Vault.

  Arguments:
       -src        : Source DB connection information
       -srcname    : Name of a source
       -desc       : Optional description of the source

  Examples:
     avdb2db add_source -src lnxserver:50000
        -desc 'HR Database'
```

11.9 setup

Adds the IBM DB2 source user credentials to the wallet, creates a database alias in the wallet for the source user, and verifies the connection to the source using the wallet. You also can use this command to change the source user credentials in the wallet after

these credentials have been changed in the source database. Run this command on the Audit Vault collection agent.

Syntax

```
avdb2db setup -srcname srcname
```

Arguments

Argument	Description
`-srcname srcname`	Enter the name of the IBM DB2 source database. Remember that the source database name is case-sensitive.

Usage Notes

- If you installed the collection agent on a Microsoft Windows computer, run the `avdb2db setup` command from the *ORACLE_HOME*`\`*agent_directory*`\bin` directory. For UNIX or Linux installations, set the appropriate environment variables before running this command. See Section 2.2 for more information.
- The `avdb2db setup` command prompts for a user name and password. This user account must have privileges to run the IBM DB2 `db2audit` command (for example, a user who has the `sysadmin` privilege).

Example

```
$ avdb2db setup -srcname db2db4
Enter a username : source_user_name
Enter a password : password

***** Credentials Successfully added *****
```

11.10 verify

Verifies that the IBM DB2 source database is compatible for setting up the specified collectors. Run this command on either the Audit Vault Server or the Audit Vault collection agent.

Syntax

```
avdb2db verify -src host:port:/database_name
```

Arguments

Argument	Description
`-src host:port:/database_name`	Enter the source database connection information: host name and port number, separated by a colon. Typically, the host is the fully qualified domain name or IP address of the server on which the IBM DB2 source database is running, and the port number is 50000. The *database_name* setting refers to the name of the DB2 source database.

Usage Notes

- The `avdb2db verify` command checks the following:
  - Whether the version of the database is supported: Version 8.2 or 9.5

  - Whether the source user has the required privileges in the source database that is to be registered with Oracle Audit Vault
  - Whether auditing is enabled in the source database
  - Whether the operating system on which the source database is running is supported

- If you installed the collection agent on a Microsoft Windows computer and want to run the `avdb2db verify` command from there, run it from the *`ORACLE_HOME`*`\`*`agent_directory`*`\bin` directory. For UNIX or Linux installations, set the appropriate environment variables before running this command. See Section 2.2 for more information.
- The `avdb2db verify` command prompts for a user name and password. This user account must have privileges to run the IBM DB2 `db2audit` command (for example, a user who has the `sysadmin` privilege).

Example

The following example verifies that the source database is compatible with the DB2DB collector on a Linux or UNIX system.

```
$ avdb2db verify -src 192.0.2.7:50000:sales_db
Enter a username : source_user_name
Enter a password : password

***** Source Verified *****
```

12

REDO Collector Database Reference

This chapter contains:

- Oracle9i Database Release 2 (9.2) Audit Source Parameter Recommendations
- Oracle Database 10g Release 1 (10.1) Audit Source Parameter Recommendations
- Oracle Database 10g Release 2 (10.2) Audit Source Parameter Recommendations
- Oracle Database 11g Release 1 (11.1) Audit Source Parameter Recommendations

12.1 About the Recommended Settings for the REDO Collector

This chapter describes recommendations for setting initialization parameters if you plan to use the REDO collector to collect audit data. After you change the initialization parameters described in these sections, you must restart the source database before configuring the REDO collect to collect audit data.

12.2 Oracle9*i* Database Release 2 (9.2) Audit Source Parameter Recommendations

At each participating source site, configure the initialization parameters for each database to include the following hidden parameters (see Table 12–1).

Table 12–1 Hidden Initialization Parameters to Be Configured for the Database Source

Parameter Name and Recommendation	Mandatory or Recommended Parameter	Default Value	Description
`_first_spare_parameter=200M/(current_shared_pool_size+200M)`	Mandatory	10	The threshold (percent) of `SHARED_POOL_SIZE` memory at which spillover to disk is triggered for captured messages
`_kghdsidx_count=1`	Recommended	Range: 10 to 80	This parameter prevents the `SHARED_POOL` from being divided among CPUs.
`_job_queue_interval=1`	Recommended	5	Scan rate interval (seconds) of job queue
`_spin_count=5000`	Recommended	2000	Controls the amount of time spent waiting (that is, "spinning") for a serialization latch to be released. Its default value is 2000. Set this parameter if Memory Queue and Memory Queue Subscriber latch sleeps are high.

At each participating source site, confirm that the following required initialization parameters are set appropriately for each database (see Table 12–2). The `SHARED_POOL_SIZE` parameter is of particular importance for REDO collectors.

Table 12–2 Initialization Parameters to Be Configured for the Database Source

Parameter Name and Recommendation	Mandatory or Recommended Parameter	Default Value	Description
AQ_TM_PROCESSES=4	Mandatory	Default: 0 Range: 0 to 10	Establishes queue monitor processes. Setting the parameter to 1 or higher starts the specified number of queue monitor processes. These queue monitor processes manage time-based operations of messages such as delay and expiration, clean up retained messages after the specified retention time, and clean up consumed messages if the retention time is zero. This parameter is required for both Streams captured messages and user-enqueued messages.
COMPATIBLE=9.2.0	Mandatory	Default: 8.1.0 Range: 8.1.0 to Current Release Number	This parameter specifies the release with which the Oracle database must maintain compatibility. Oracle databases with different compatibility levels can interoperate. To use Streams, then set this parameter to 9.2.0 or higher.
GLOBAL_NAMES=true	Recommended	Default: false Range: true or false	Specifies whether a database link is required to have the same name as the database to which it connects. If you want to use Streams to share information between databases, then set this parameter to true for each database that in your Streams environment.
JOB_QUEUE_ PROCESSES=4	Mandatory	Default: 0 Range: 0 to 1000	Specifies the number of job queue processes for each instance (J000 ... J999). Job queue processes handle requests created by DBMS_JOB. You can change the setting for JOB_QUEUE_PROCESSES dynamically by using the ALTER SYSTEM SQL statement. Set this parameter to at least 2 for each database that propagates events in your Streams environment, and set it to the same value as the maximum number of jobs that can run simultaneously, plus 2.
LOG_PALALLELISM=1 This parameter has to be set to 1. Note that the default value is 1.	Mandatory	Default: 1 Range: 1 to 255	Specifies the level of concurrency for redo allocation within Oracle. If you plan to run one or more capture processes on a database, then set this parameter to 1. Setting this parameter to 1 does not affect the parallelism of capture. You can set parallelism for a capture process running the SET_PARAMETER procedure in the DBMS_CAPTURE_ADM package.
LOGMNR_MAX_ PERSISTENT_ SESSIONS=3 This parameter must be set to at least 1 which is also the default value.	Mandatory	Default: 1 Range: 1 to LICENSE_MAX_ SESSIONS	Specifies the maximum number of persistent LogMiner mining sessions that are concurrently active when all sessions are mining redo logs generated by instances. If you plan to run multiple Streams capture processes on a single database, then set this parameter equal to or higher than the number of planned capture processes.
OPEN_LINKS=4	Recommended	Default: 4 Range: 0 to 255	Specifies the maximum number of concurrent open connections to remote databases in one session. These connections include database links, external procedures, and cartridges, each of which uses a separate process. In a Streams environment, set this parameter to the default value of 4 or higher.

Table 12–2 (Cont.) Initialization Parameters to Be Configured for the Database Source

Parameter Name and Recommendation	Mandatory or Recommended Parameter	Default Value	Description
PARALLEL_MAX_SERVERS=20	Mandatory	Default: Derived from the values of the following parameters: CPU_COUNT PARALLEL_ADAPTIVE_MULTI_USER PARALLEL_AUTOMATIC_TUNING Range: 0 to 3599	Specifies the maximum number of parallel execution processes and parallel recovery processes for an instance. As demand increases, Oracle Database increases the number of processes from the number created at instance startup up to this value. In a Streams environment, each capture process and apply process can use multiple parallel execution servers. Set this initialization parameter to an appropriate value to ensure that there are enough parallel execution servers.
PROCESSES	Recommended	Default: Derived from PARALLEL_MAX_SERVERS Range: 6 to operating system dependent limit	Specifies the maximum number of operating system user processes that can simultaneously connect to Oracle Database. Ensure that the value of this parameter allows for all background processes, such as locks, job queue processes, and parallel execution processes. In Streams, capture processes and apply processes use background processes and parallel execution processes, and propagation jobs use job queue processes.
SESSIONS	Recommended	Default: Derived from: (1.1 * PROCESSES) + 5 Range: 1 to 231	Specifies the maximum number of sessions that can be created in the system. If you plan to run one or more capture processes or apply processes in a database, then you may need to increase the size of this parameter. Each background process in a database requires a session.
SGA_MAX_SIZE Increase by at least 200M	Mandatory	Default: Initial size of SGA at startup Range: 0 to operating system dependent limit	Specifies the maximum size of SGA for the lifetime of a database instance. If you plan to run multiple capture processes on a single database, then you may need to increase the size of this parameter.

Table 12–2 (Cont.) Initialization Parameters to Be Configured for the Database Source

Parameter Name and Recommendation	Mandatory or Recommended Parameter	Default Value	Description
SHARED_POOL_SIZE= (Increase by at least 200M)	Mandatory	Default: 32-bit platforms: 8 MB, rounded up to the nearest granule size 64-bit platforms: 64 MB, rounded up to the nearest granule size Range: Minimum: the granule size Maximum: operating system-dependent	Specifies (in bytes) the size of the shared pool. The shared pool contains shared cursors, stored procedures, control structures, and other structures. You should increase the size of the shared pool by 10 MB for each capture process on a database. Additional memory is required from the shared pool to store logical change records (LCRs) in the buffer queue. Size this parameter so that LCRs remain in memory as long as possible. Use the following formula to calculate the point at which LCRs will spill to disk. SHARED_POOL_SIZE * _first_spare_parameter / 100
TIMED_STATISTICS	Recommended	Default: If STATISTICS_LEVEL is set to TYPICAL or ALL, then true If STATISTICS_LEVEL is set to BASIC, then false The default for STATISTICS_LEVEL is TYPICAL. Range: true or false	Specifies whether statistics related to time are collected. If you want to collect elapsed time statistics in the data dictionary views related to Streams, then set this parameter to true. The following views include elapsed time statistics: V$STREAMS_CAPTURE V$STREAMS_APPLY_COORDINATOR V$STREAMS_APPLY_READER V$STREAMS_APPLY_SERVER
TRANSACTION_AUDITING=TRUE	Mandatory	Default: TRUE Range: true or false	If TRANSACTION_AUDITING is set to true, Oracle Database generates a special redo record that contains the user logon name, username, the session ID, some operating system information, and client information. For each successive transaction, Oracle Database generates a record that contains only the session ID. These subsequent records link back to the first record, which also contains the session ID. These records can be useful if you are using a redo log analysis tool. You can access the records by dumping the redo log. If TRANSACTION_AUDITING is false, no redo record will be generated. Set TRANSACTION_AUDITING to TRUE for databases that have a Streams capture process configured

An additional initialization parameter must be configured at each instance involved in the Oracle Real Application Clusters (Oracle RAC) configuration. In addition to the parameters referenced previously, the parameter Table 12–3 should be included.

Table 12–3 ARCHIVE_LAG_TARGET Recommended Setting

Parameter Name and Recommendation	Mandatory or Recommended Parameter	Default Value	Description
ARCHIVE_LAG_TARGET=1800	Recommended	Default: 0 Range: 0 or any integer in [60, 7200]	Limits the amount of data that can be lost and effectively increases the availability of the standby database by forcing a log switch after a user-specified time period elapses. If you are using Streams in an Oracle Real Application Clusters environment, then set this parameter to a value greater than zero to switch the log files automatically. **See Also:** The section titled "Streams Capture Processes and Oracle Real Application Clusters" in *Oracle9i Streams* release 2 (9.2)

12.3 Oracle Database 10*g* Release 1 (10.1) Audit Source Parameter Recommendations

At each participating source site, configure the initialization parameters for each database to include the following hidden parameters (see Table 12–4).

Table 12–4 Hidden Initialization Parameters to Be Configured for the Database Source

Parameter Name and Recommendation	Mandatory or Recommended Parameter	Default Value	Description
`_job_queue_interval=1`	Recommended	5	Scan rate interval (seconds) of job queue
`_spin_count=5000`	Recommended	2000	Controls the amount of time spent waiting (that is, "spinning") for a serialization latch to be released. Its default value is 2000. Set this parameter if Memory Queue and Memory Queue Subscriber latch sleeps are high.

At each participating source site, confirm that the following required initialization parameters are set appropriately for each database (see Table 12–5).

Table 12–5 Initialization Parameters to Be Configured for the Database Source

Parameter Name and Recommendation	Mandatory or Recommended Parameter	Default Value	Description
`COMPATIBLE= 10.1.0`	Mandatory	Default: `9.2.0` Range: `9.2.0` to Current Release Number Modifiable?: No	This parameter specifies the release with which the Oracle database must maintain compatibility. Oracle databases with different compatibility levels can interoperate. To use the new Streams features introduced in Oracle Database 10*g*, set this parameter to `10.1.0` or higher. To use downstream capture, set the parameter to `10.1.0` or higher for both the source database and the downstream database.
`Cursor_space_for_time= FALSE` This parameter has to be set to `FALSE`. Note that `FALSE` is the default value for this parameter.	Mandatory	Default: `FALSE` Range: `FALSE` or `TRUE`	Do not change this parameter when using Streams or Logical Standby.
`GLOBAL_NAMES=true`	Recommended	Default: `false` Range: `true` or `false` Modifiable?: Yes	Specifies whether a database link is required to have the same name as the database to which it connects. To use Streams to share information between databases, set this parameter to `true` for each database in your Streams environment.
`JOB_QUEUE_PROCESSES=4`	Mandatory	Default: 0 Range: 0 to 1000 Modifiable?: Yes	Specifies the number of job queue processes for each instance (J000 ... J999). Job queue processes handle requests created by the `DBMS_JOB` PL/SQL package. Set this parameter to at least 2 at each database that propagates events in your Streams environment, and then set it to the same value as the maximum number of jobs that can run simultaneously, plus `2`.
`LOG_ARCHIVE_DEST_n`	Recommended	Default: None Range: None Modifiable?: Yes	Defines up to ten log archive destinations, where *n* is 1, 2, 3, ... 10. To use downstream capture and copy the redo log files to the downstream database using log transport services, have at least one log archive destination on the site that runs the downstream capture process. **See Also**: *Oracle Data Guard Concepts and Administration*

Table 12–5 (Cont.) Initialization Parameters to Be Configured for the Database Source

Parameter Name and Recommendation	Mandatory or Recommended Parameter	Default Value	Description
LOG_ARCHIVE_DEST_STATE_*n*	Recommended	Default: enable Range: One of the following: alternate reset defer enable Modifiable?: Yes	Specifies the availability state of the corresponding destination. The parameter suffix (1 through 10) specifies one of the ten corresponding LOG_ARCHIVE_DEST_n destination parameters. To use downstream capture and copy the redo log files to the downstream database using log transport services, ensure that the destination that corresponds to the LOG_ARCHIVE_DEST_*n* destination for the downstream database is set to enable.
OPEN_LINKS	Recommended	Default: 4 Range: 0 to 255 Modifiable?: No	Specifies the maximum number of concurrent open connections to remote databases in one session. These connections include database links, external procedures, and cartridges, each of which uses a separate process. In a Streams environment, set this parameter to the default value of 4 or higher.
PARALLEL_MAX_SERVERS Set this parameter to at least 20.	Mandatory	Default: Derived from the values of the following parameters: CPU_COUNT PARALLEL_ADAPTIVE_MULTI_USER PARALLEL_AUTOMATIC_TUNING Range: 0 to 3599 Modifiable?: Yes	Specifies the maximum number of parallel execution processes and parallel recovery processes for an instance. As demand increases, Oracle Database increases the number of processes from the number created at instance startup up to this value. In a Streams environment, each capture process and apply process can use multiple parallel execution servers. Set this initialization parameter to an appropriate value to ensure that there are enough parallel execution servers.
PROCESSES	Recommended	Default: Derived from PARALLEL_MAX_SERVERS Range: 6 to operating system dependent limit Modifiable?: No	Specifies the maximum number of operating system user processes that can simultaneously connect to Oracle Database. Ensure that the value of this parameter allows for all background processes, such as locks, job queue processes, and parallel execution processes. In Streams, capture processes and apply processes use background processes and parallel execution processes, and propagation jobs use job queue processes.
SESSIONS	Recommended	Default: Derived from: (1.1 * PROCESSES) + 5 Range: 1 to 231 Modifiable?: No	Specifies the maximum number of sessions that can be created in the system. To run one or more capture processes or apply processes in a database, then you may need to increase the size of this parameter. Each background process in a database requires a session.
SGA_MAX_SIZE Increase by at least 200M	Mandatory	Default: Initial size of SGA at startup Range: 0 to operating system dependent limit Modifiable?: No	Specifies the maximum size of SGA for the lifetime of a database instance. To run multiple capture processes on a single database, you may need to increase the size of this parameter.

Table 12–5 (Cont.) Initialization Parameters to Be Configured for the Database Source

Parameter Name and Recommendation	Mandatory or Recommended Parameter	Default Value	Description
SHARED_POOL_SIZE	Recommended	Default: 32-bit platforms: 32 MB, rounded up to the nearest granule size 64-bit platforms: 84 MB, rounded up to the nearest granule size Range: Minimum: the granule size Maximum: operating system dependent Modifiable?: Yes	Specifies (in bytes) the size of the shared pool. The shared pool contains shared cursors, stored procedures, control structures, and other structures. If you set the STREAMS_POOL_SIZE initialization parameter to zero, then Streams can use up to 10 percent of the shared pool.

Table 12–5 (Cont.) Initialization Parameters to Be Configured for the Database Source

Parameter Name and Recommendation	Mandatory or Recommended Parameter	Default Value	Description
STREAMS_POOL_SIZE>200M If using sga_target, also increase this value by at least 200M.	Mandatory	Default: 0 Range: Minimum: 0 Maximum: operating system dependent Modifiable?: Yes	Specifies (in bytes) the size of the Streams pool. The Streams pool contains captured events. In addition, Oracle Database uses the Streams pool for internal communications during parallel capture and apply. If the size of the Streams pool is greater than zero, then Oracle Database allocates any SGA memory used by Streams from the Streams pool. If you set the Streams pool size to zero, then Oracle Database allocates SGA memory used by Streams from the shared pool and can use up to 10 percent of the shared pool. You can modify this parameter. However, if you set this parameter to zero when a database instance starts, then increasing it beyond zero has no effect on the current instance because it is using the shared pool for Streams allocations. Also, if you set this parameter to a value greater than zero when an instance starts and is then reduce it to zero when the instance is running, then Streams processes and jobs will not run. You should increase the size of the Streams pool for each of the following factors: ■ 10 MB for each capture process parallelism ■ 1 MB for each apply process parallelism ■ 10 MB or more for each queue staging captured events For example, suppose you set parallelism to 3 for a capture process, and then increase the Streams pool by 30 MB. If you set parallelism to 5 for an apply process, then you must increase the Streams pool by 5 MB.
TIMED_STATISTICS	Recommended	Default: If STATISTICS_LEVEL is set to TYPICAL or ALL, then true If STATISTICS_LEVEL is set to BASIC, then false The default for STATISTICS_LEVEL is TYPICAL. Range: true or false Modifiable?: Yes	Specifies whether statistics related to time are collected. To collect elapsed time statistics in the data dictionary views related to Streams, set this parameter to true. The following views include elapsed time statistics: V$STREAMS_CAPTURE V$STREAMS_APPLY_COORDINATOR V$STREAMS_APPLY_READER V$STREAMS_APPLY_SERVER
UNDO_RETENTION=3600	Recommended	Default: 900 Range: 0 to 2^32-1 (max value represented by 32 bits) Modifiable?: Yes	Specifies (in seconds) the amount of committed undo information to retain in the database. For a database running one or more capture processes, set this parameter to specify an adequate undo retention period. If you are running one or more capture processes and you are unsure about the proper setting, then try setting this parameter to at least 3600. If you encounter "snapshot too old" errors, then increase the setting for this parameter until these errors cease. Ensure that the undo tablespace has enough space to accommodate the UNDO_RETENTION setting. **See Also:** *Oracle Database Administrator's Guide* for more information about the retention period and the undo tablespace

12.4 Oracle Database 10*g* Release 2 (10.2) Audit Source Parameter Recommendations

For best results in a REDO collector environment, set the following initialization parameters at each participating database: COMPATIBLE, GLOBAL_NAMES, _job_queue_interval, SGA_TARGET, STREAMS_POOL_SIZE.

At each participating source site, configure the initialization parameters for each database to include the following hidden parameters (see Table 12–6).

Table 12–6 Hidden Initialization Parameters to Be Configured for the Database Source

Parameter Name and Recommendation	Mandatory or Recommended Parameter	Default Value	Description
_job_queue_interval=1	Recommended	5	Scan rate interval (seconds) of job queue
_spin_count=5000	Recommended	2000	Controls the amount of time spent waiting (that is, "spinning") for a serialization latch to be released. Its default value is 2000. Set this parameter if Memory Queue and Memory Queue Subscriber latch sleeps are high.

At each participating source site, confirm that the following required initialization parameters are set appropriately for each database (see Table 12–7). Enable autotuning of the various pools within the SGA, by setting SGA_TARGET to a large nonzero value. Leave the STREAMS_POOL_SIZE value set to 0. The combination of these to parameters enables autotuning of the SGA and the Streams Pool size will be automatically adjusted to meet the workload requirements.

Table 12–7 Initialization Parameters to Be Configured for the Database Source

Parameter Name and Recommendation	Mandatory or Recommended Parameter	Default Value	Description
COMPATIBLE= 10.2.0	Mandatory	Default: 10.0.0 Range: 9.2.0 to Current Release Number Modifiable?: No	This parameter specifies the release with which the Oracle database must maintain compatibility. Oracle databases with different compatibility levels can interoperate. To use the new Streams features introduced in Oracle Database 10*g* release 1, set this parameter to 10.1.0 or higher. To use downstream capture, set this parameter 10.1.0 or higher for both the source database and the downstream database. To use the new Streams features introduced in Oracle Database 10*g* release 2, set this parameter to 10.2.0 or higher.
GLOBAL_NAMES=true	Recommended	Default: false Range: true or false Modifiable?: Yes	Specifies whether a database link is required to have the same name as the database to which it connects. To use Streams to share information between databases, set this parameter to true for each database that participates in your Streams environment.
JOB_QUEUE_ PROCESSES=4	Mandatory	Default: 0 Range: 0 to 1000 Modifiable?: Yes	Specifies the number of job queue processes for each instance (J000 ... J999). Job queue processes handle requests created by the DBMS_JOB PL/SQL package. Set this parameter to at least 2 for each database that propagates events in your Streams environment, and then set it to the same value as the maximum number of jobs that can run simultaneously, plus 2.

Table 12–7 (Cont.) Initialization Parameters to Be Configured for the Database Source

Parameter Name and Recommendation	Mandatory or Recommended Parameter	Default Value	Description
LOG_ARCHIVE_DEST_n	Recommended	Default: None Range: None Modifiable?: Yes	Defines up to ten log archive destinations, where *n* is 1, 2, 3, ... 10. To use downstream capture and copy the redo log files to the downstream database using log transport services, at least one log archive destination must be at the site running the downstream capture process. **See Also**: *Oracle Data Guard Concepts and Administration*
LOG_ARCHIVE_DEST_STATE_*n*	Recommended	Default: enable Range: One of the following: alternate reset defer enable Modifiable?: Yes	Specifies the availability state of the corresponding destination. The parameter suffix (1 through 10) specifies one of the ten corresponding LOG_ARCHIVE_DEST_*n* destination parameters. To use downstream capture and copy the redo log files to the downstream database using log transport services, ensure that the destination that corresponds to the LOG_ARCHIVE_DEST_*n* destination for the downstream database is set to enable.
OPEN_LINKS	Recommended	Default: 4 Range: 0 to 255 Modifiable?: No	Specifies the maximum number of concurrent open connections to remote databases in one session. These connections include database links, external procedures, and cartridges, each of which uses a separate process. In a Streams environment, set this parameter to the default value of 4 or higher.
PARALLEL_MAX_SERVERS Set this parameter to at least 20.	Mandatory	Default: Derived from the values of the following parameters: CPU_COUNT PARALLEL_ADAPTIVE_MULTI_USER PARALLEL_AUTOMATIC_TUNING Range: 0 to 3599 Modifiable?: Yes	Specifies the maximum number of parallel execution processes and parallel recovery processes for an instance. As demand increases, Oracle Database increases the number of processes from the number created at instance startup up to this value. In a Streams environment, each capture process and apply process can use multiple parallel execution servers. Set this initialization parameter to an appropriate value to ensure that there are enough parallel execution servers.
PROCESSES	Recommended	Default: Derived from PARALLEL_MAX_SERVERS Range: 6 to operating system dependent limit Modifiable?: No	Specifies the maximum number of operating system user processes that can simultaneously connect to an Oracle database. Ensure that the value of this parameter allows for all background processes, such as locks, job queue processes, and parallel execution processes. In Streams, capture processes and apply processes use background processes and parallel execution processes, and propagation jobs use job queue processes.
SESSIONS	Recommended	Default: Derived from: (1.1 * PROCESSES) + 5 Range: 1 to 231 Modifiable?: No	Specifies the maximum number of sessions that can be created in the system. To run one or more capture processes or apply processes in a database, then you may need to increase the size of this parameter. Each background process in a database requires a session.
SGA_MAX_SIZE Increase by at least 200M	Mandatory	Default: Initial size of SGA at startup Range: 0 to operating system dependent limit Modifiable?: No	Specifies the maximum size of SGA for the lifetime of a database instance. To run multiple capture processes on a single database, you may need to increase the size of this parameter. See the STREAMS_POOL_SIZE initialization parameter for more specific recommendations.

Table 12–7 (Cont.) Initialization Parameters to Be Configured for the Database Source

Parameter Name and Recommendation	Mandatory or Recommended Parameter	Default Value	Description
SGA_TARGET >0 Increase this parameter by at least 200M.	Mandatory	Default: 0 (SGA autotuning is disabled) Range: 64 to operating system-dependent Modifiable?: Yes	Specifies the total size of all System Global Area (SGA) components. If you set this parameter to a nonzero value, then the size of the Streams pool is managed by Automatic Shared Memory Management. See the STREAMS_POOL_SIZE initialization parameter for more specific recommendations.
SHARED_POOL_SIZE=0	Recommended	Default: 32-bit platforms: 32 MB, rounded up to the nearest granule size 64-bit platforms: 84 MB, rounded up to the nearest granule size Range: Minimum: the granule size Maximum: operating system-dependent Modifiable?: Yes	Specifies (in bytes) the size of the shared pool. The shared pool contains shared cursors, stored procedures, control structures, and other structures. If you set the SGA_TARGET and STREAMS_POOL_SIZE initialization parameters to zero, then Streams transfers an amount equal to 10 percent of the shared pool from the buffer cache to the Streams pool.

Table 12–7 (Cont.) Initialization Parameters to Be Configured for the Database Source

Parameter Name and Recommendation	Mandatory or Recommended Parameter	Default Value	Description
STREAMS_POOL_SIZE=200	Mandatory	Default: 0 Range: Minimum: 0 Maximum: operating system-dependent Modifiable?: Yes	Specifies (in bytes) the size of the Streams pool. The Streams pool contains captured events. In addition, Oracle Database uses the Streams pool for internal communications during parallel capture and apply. If you set the SGA_TARGET initialization parameter to a nonzero value, then the Streams pool size is set by Automatic Shared memory management, and STREAMS_POOL_SIZE specifies the minimum size. You should set the STREAMS_POOL_SIZE initialization parameter to 200 MB and, if necessary, increment the SGA_TARGET and SGA_MAX initialization parameters appropriately. For example, if the SGA_TARGET initialization parameter is already set to 2 GB, setting STREAMS_POOL_SIZE=200 MB does not require you to increase the SGA_TARGET initialization parameter setting. However, if the SGA_TARGET initialization parameter is set to 600 MB and the STREAMS_POOL_SIZE initialization parameter is increased to 200 MB, then you should increase the SGA_TARGET initialization parameter value similarly. This parameter is modifiable. If you reduce this parameter setting to zero when an instance is running, then Streams processes and jobs cannot run. You should increase the size of the Streams pool for each of the following factors: ■ 10 MB for each capture process parallelism ■ 10 MB or more for each buffered queue. The buffered queue is where the Logical Change Records (LCRs) are stored. ■ 1 MB for each apply process parallelism You can use the V$STREAMS_POOL_ADVICE dynamic performance view to determine an appropriate setting for this parameter. For example, if you set parallelism to 3 for a capture process, then increase the Streams pool by 30 MB. If you set parallelism to 5 for an apply process, then increase the Streams pool by 5 MB.
TIMED_STATISTICS	Recommended	Default: If STATISTICS_LEVEL is set to TYPICAL or ALL, then true If STATISTICS_LEVEL is set to BASIC, then false The default for STATISTICS_LEVEL is TYPICAL. Range: true or false Modifiable?: Yes	Specifies whether statistics related to time are collected. To collect elapsed time statistics in the data dictionary views related to Stream, set this parameter to true. The following views include elapsed time statistics: V$STREAMS_CAPTURE V$STREAMS_APPLY_COORDINATOR V$STREAMS_APPLY_READER V$STREAMS_APPLY_SERVER

Table 12–7 (Cont.) Initialization Parameters to Be Configured for the Database Source

Parameter Name and Recommendation	Mandatory or Recommended Parameter	Default Value	Description
UNDO_ RETENTION=3600	Recommended	Default: 900 Range: 0 to 2^32-1 (max value represented by 32 bits) Modifiable?: Yes	Specifies (in seconds) the amount of committed undo information to retain in the database. For a database running one or more capture processes, set this parameter to specify an adequate undo retention period. If you are running one or more capture processes and you are unsure about the proper setting, then try setting this parameter to at least 3600. If you encounter "snapshot too old" errors, then increase the setting for this parameter until these errors cease. Ensure that the undo tablespace has enough space to accommodate the UNDO_RETENTION setting. **See Also:** *Oracle Database Administrator's Guide* for more information about the UNDO_RETENTION parameter

12.5 Oracle Database 11*g* Release 1 (11.1) Audit Source Parameter Recommendations

For best results in a REDO collector environment, set the following initialization parameters at each participating database: compatible, GLOBAL_NAMES, _job_ queue_interval, SGA_TARGET, STREAMS_POOL_SIZE.

At each participating source site, configure the initialization parameters for each database to include the following hidden parameters (see Table 12–6).

Table 12–8 Hidden Initialization Parameters to Be Configured for the Database Source

Parameter Name and Recommendation	Mandatory or Recommended Parameter	Default Value	Description
_job_queue_interval=1	Recommended	5	Scan rate interval (seconds) of job queue
_spin_count=5000	Recommended	2000	Controls the amount of time spent waiting (that is, "spinning") for a serialization latch to be released. Its default value is 2000. Set this parameter if Memory Queue and Memory Queue Subscriber latch sleeps are high.

At each participating source site, confirm that the following required initialization parameters are set appropriately for each database (see Table 12–7). Enable autotuning of the various pools within the SGA, by setting SGA_TARGET to a large nonzero value. Leave the STREAMS_POOL_SIZE value set to 0. The combination of these to parameters enables autotuning of the SGA and the Streams Pool size will be automatically adjusted to meet the workload requirements.

Table 12–9 Initialization Parameters to Be Configured for the Database Source

Parameter Name and Recommendation	Mandatory or Recommended Parameter	Default Value	Description
`COMPATIBLE= 11.1.0`	Mandatory	Default: `11.1.0` Range: `10.1.0` to Current Release Number Modifiable?: No	This parameter specifies the release with which the Oracle server must maintain compatibility. Oracle servers with different compatibility levels can interoperate. To use the new Streams features introduced in Oracle Database 10*g* release 1, this parameter must be set to `10.1.0` or higher. To use downstream capture, this parameter must be set to `10.1.0` or higher at both the source database and the downstream database. To use the new Streams features introduced in Oracle Database 10*g* release 2, this parameter must be set to `10.2.0` or higher. To use the new Streams features introduced in Oracle Database 11*g* release 1, this parameter must be set to `11.1.0` or higher.
`GLOBAL_NAMES=true`	Recommended	Default: false Range: true or false Modifiable?: Yes	Specifies whether a database link is required to have the same name as the database to which it connects. To use Streams to share information between databases, set this parameter to `true` at each database that is participating in your Streams environment.
`JOB_QUEUE_ PROCESSES=4`	Mandatory	Default: 0 Range: 0 to 1000 Modifiable?: Yes	Specifies the number of Jnnn job queue processes for each instance (J000 ... J999). Job queue processes handle requests created by `DBMS_JOB`. This parameter must be set to at least 2 at each database that is propagating events in your Streams environment, and should be set to the same value as the maximum number of jobs that can run simultaneously plus two.
`LOG_ARCHIVE_DEST_n`	Recommended	Default: None Range: None Modifiable?: Yes	Defines up to ten log archive destinations, where n is 1, 2, 3, ... 10. To use downstream capture and copy the redo log files to the downstream database using log transport services, at least one log archive destination must be at the site running the downstream capture process. **See Also**: *Oracle Data Guard Concepts and Administration*

`LOG_ARCHIVE_DEST_ STATE_n`	Recommended	Default: enable
		Range: One of the following:
		`alternate`
		`reset`
		`defer`
		`enable`
		Modifiable?: Yes
			Specifies the availability state of the corresponding destination. The parameter suffix (1 through 10) specifies one of the ten corresponding `LOG_ARCHIVE_DEST_n` destination parameters.
			To use downstream capture and copy the redo log files to the downstream database using log transport services, ensure that the destination that corresponds to the `LOG_ARCHIVE_DEST_ n` destination for the downstream database is set to `enable`.

`OPEN_LINKS`	Recommended	Default: 4 Range: 0 to 255 Modifiable?: No	Specifies the maximum number of concurrent open connections to remote databases in one session. These connections include database links, as well as external procedures and cartridges, each of which uses a separate process. In a Streams environment, ensure that this parameter is set to the default value of 4 or higher.
`PROCESSES`	Recommended	Default: Derived from PARALLEL_ MAX_SERVERS Range: 6 to operating system dependent limit Modifiable?: No	Specifies the maximum number of operating system user processes that can simultaneously connect to Oracle. Ensure that the value of this parameter allows for all background processes, such as locks, job queue processes, and parallel execution processes. In Streams, capture processes and apply processes use background processes and parallel execution processes, and propagation jobs use job queue processes.

Table 12–9 (Cont.) Initialization Parameters to Be Configured for the Database Source

Parameter Name and Recommendation	Mandatory or Recommended Parameter	Default Value	Description
SESSIONS	Recommended	Default: Derived from: (1.1 * PROCESSES) + 5 Range: 1 to 231 Modifiable?: No	Specifies the maximum number of sessions that can be created in the system. To run one or more capture processes or apply processes in a database, then you may need to increase the size of this parameter. Each background process in a database requires a session.
SGA_MAX_SIZE Increase by at least 200M	Mandatory	Default: Initial size of SGA at startup Range: 0 to operating system dependent limit Modifiable?: No	Specifies the maximum size of SGA for the lifetime of a database instance. To run multiple capture processes on a single database, you may need to increase the size of this parameter. See the STREAMS_POOL_SIZE initialization parameter for more specific recommendations.
SGA_TARGET >0 Increase this parameter by at least 200M.	Mandatory	Default: 0 (SGA autotuning is disabled) Range: 64 to operating system-dependent Modifiable?: Yes	Specifies the total size of all System Global Area (SGA) components. If this parameter is set to a nonzero value, then the size of the Streams pool is managed by Automatic Shared Memory Management. See the STREAMS_POOL_SIZE initialization parameter for more specific recommendations.
SHARED_POOL_SIZE=0	Recommended	Default: 32-bit platforms: 32 MB, rounded up to the nearest granule size 64-bit platforms: 84 MB, rounded up to the nearest granule size Range: Minimum: the granule size Maximum: operating system-dependent Modifiable?: Yes	Specifies (in bytes) the size of the shared pool. The shared pool contains shared cursors, stored procedures, control structures, and other structures. If the SGA_TARGET and STREAMS_POOL_SIZE initialization parameters are set to zero, then Streams transfers an amount equal to 10% of the shared pool from the buffer cache to the Streams pool. The STREAMS_POOL_SIZE initialization parameter should be set to 200 MB and, if necessary, increment the SGA_TARGET and SGA_MAX initialization parameters appropriately. For example, if the SGA_TARGET initialization parameter is already set to 2 GB, setting STREAMS_POOL_SIZE=200 MB would not require that the SGA_TARGET initialization parameter be increased. However, if the SGA_TARGET initialization parameter is set to 600 MB and the STREAMS_POOL_SIZE initialization parameter is increased to 200 MB, then it is recommended that the SGA_TARGET initialization parameter value be increased similarly.

Table 12–9 (Cont.) Initialization Parameters to Be Configured for the Database Source

Parameter Name and Recommendation	Mandatory or Recommended Parameter	Default Value	Description
STREAMS_POOL_SIZE=200	Mandatory	Default: 0 Range: Minimum: 0 Maximum: operating system-dependent Modifiable?: Yes	Specifies (in bytes) the size of the Streams pool. The Streams pool contains captured events. In addition, the Streams pool is used for internal communications during parallel capture and apply. If the SGA_TARGET initialization parameter is set to a nonzero value, then the Streams pool size is set by Automatic Shared memory management, and STREAMS_POOL_SIZE specifies the minimum size. This parameter is modifiable. If this parameter is reduced to zero when an instance is running, then Streams processes and jobs will not run. You should increase the size of the Streams pool for each of the following factors: 10 MB for each capture process parallelism 10 MB or more for each buffered queue. The buffered queue is where the Logical Change Records (LCRs) are stored. 1 MB for each apply process parallelism You can use the V$STREAMS_POOL_ADVICE dynamic performance view to determine an appropriate setting for this parameter. For example, if parallelism is set to 3 for a capture process, then increase the Streams pool by 30 MB. If parallelism is set to 5 for an apply process, then increase the Streams pool by 5 MB.
TIMED_STATISTICS	Recommended	Default: If STATISTICS_LEVEL is set to TYPICAL or ALL, then true If STATISTICS_LEVEL is set to BASIC, then false The default for STATISTICS_LEVEL is TYPICAL. Range: true or false Modifiable?: Yes	Specifies whether statistics related to time are collected. To collect elapsed time statistics in the data dictionary views related to Stream, set this parameter to true. The views that include elapsed time statistics include: V$STREAMS_CAPTURE V$STREAMS_APPLY_COORDINATOR V$STREAMS_APPLY_READER V$STREAMS_APPLY_SERVER
UNDO_RETENTION=3600	Recommended	Default: 900 Range: 0 to 2^32-1 (max value represented by 32 bits) Modifiable?: Yes	Specifies (in seconds) the amount of committed undo information to retain in the database. For a database running one or more capture processes, ensure that this parameter is set to specify an adequate undo retention period. If you are running one or more capture processes and you are unsure about the proper setting, then try setting this parameter to at least 3600. If you encounter "snapshot too old" errors, then increase the setting for this parameter until these errors cease. Ensure that the undo tablespace has enough space to accommodate the UNDO_RETENTION setting. **See Also**: *Oracle Database Administrator's Guide* for more information about the UNDO_RETENTION parameter

13

DBMS_AUDIT_MGMT Data Dictionary Views

The DBMS_AUDIT_MGMT data dictionary views describe audit configuration settings that you create with the DBMS_AUDIT_MGMT PL/SQL package. Chapter 14 describes this package in detail.

Table 13–1 lists data dictionary views that are described in this section.

Table 13–1 DBMS_AUDIT_MGMT Data Dictionary Views

Data Dictionary View	Description
DBA_AUDIT_MGMT_CONFIG_PARAMS	Displays the currently configured audit trail properties that are used by the DBMS_AUDIT_MGMT PL/SQL package
DBA_AUDIT_MGMT_LAST_ARCH_TS	Displays the last archive timestamps that have been set for audit trail cleanup or purges.
DBA_AUDIT_MGMT_CLEANUP_JOBS	Displays the currently configured audit trail cleanup or purge jobs
DBA_AUDIT_MGMT_CLEAN_EVENTS	Displays the history of cleanup or purge events. Periodically, as user SYS connected with the SYSDBA privilege, you should delete the contents of this view so that it will not grow too large. For example: `DELETE FROM DBA_AUDIT_MGMT_CLEAN_EVENTS;`

13.1 DBA_AUDIT_MGMT_CONFIG_PARAMS

The DBA_AUDIT_MGMT_CONFIG_PARAMS data dictionary view displays the currently configured audit trail properties that are used by the DBMS_AUDIT_MGMT PL/SQL package.

Column	Datatype	Null	Description
PARAMETER_NAME	VARCHAR2(1024)	NOT NULL	Name of the property
PARAMETER_VALUE	VARCHAR2(4000)		Value of the property

Column	Datatype	Null	Description
AUDIT_TRAIL	VARCHAR2(28)		Audit trails for which the property is configured: ■ STANDARD AUDIT TRAIL ■ FGA AUDIT TRAIL ■ STANDARD AND FGA AUDIT TRAIL ■ OS AUDIT TRAIL ■ XML AUDIT TRAIL ■ OS AND XML AUDIT TRAIL ■ ALL AUDIT TRAILS

13.2 DBA_AUDIT_MGMT_LAST_ARCH_TS

The DBA_AUDIT_MGMT_LAST_ARCH_TS data dictionary view displays the last archive timestamps set for audit trail cleanup or purges.

Column	Datatype	Null	Description
AUDIT_TRAIL	VARCHAR2(20)		Audit trail for which the last archive timestamp applies: ■ STANDARD AUDIT TRAIL ■ FGA AUDIT TRAIL ■ OS AUDIT TRAIL ■ XML AUDIT TRAIL
RAC_INSTANCE	NUMBER	NOT NULL	Oracle RAC instance number for which the last archive timestamp applies. 0 implies "Not Applicable".
LAST_ARCHIVE_TS	TIMESTAMP(6) WITH TIMEZONE		Timestamp of the last audit record or audit file that has been archived

13.3 DBA_AUDIT_MGMT_CLEANUP_JOBS

The DBA_AUDIT_MGMT_CLEANUP_JOBS data dictionary view displays the currently configured audit trail cleanup or purge jobs.

Column	Datatype	Null	Description
JOB_NAME	VARCHAR2(100)	NOT NULL	Name of the audit trail purge job
JOB_STATUS	VARCHAR2(8)		Current status of the audit trail purge job (ENABLED) or (DISABLED)

Column	Datatype	Null	Description
AUDIT_TRAIL	VARCHAR2(28)		Audit trail for which the audit trail purge job is configured: ■ STANDARD AUDIT TRAIL ■ FGA AUDIT TRAIL ■ STANDARD AND FGA AUDIT TRAIL ■ OS AUDIT TRAIL ■ XML AUDIT TRAIL ■ OS AND XML AUDIT TRAIL ■ ALL AUDIT TRAILS
JOB_FREQUENCY	VARCHAR2(100)		Frequency at which the audit trail purge job runs

13.4 DBA_AUDIT_MGMT_CLEAN_EVENTS

The DBA_AUDIT_MGMT_CLEAN_EVENTS data dictionary view displays the history of cleanup or purge events.

Column	Datatype	Null	Description
AUDIT_TRAIL	VARCHAR2(28)		The audit trail that was cleaned at the time of the event: ■ STANDARD AUDIT TRAIL ■ FGA AUDIT TRAIL ■ STANDARD AND FGA AUDIT TRAIL ■ OS AUDIT TRAIL ■ XML AUDIT TRAIL ■ OS AND XML AUDIT TRAIL ■ ALL AUDIT TRAILS
RAC_INSTANCE	NUMBER	NOT NULL	Instance number indicating the Oracle RAC instance that was cleaned up at the time of the event. 0 implies "Not Applicable".
CLEANUP_TIME	TIMESTAMP(6) WITH TIME ZONE		Timestamp when the cleanup event completed
DELETE_COUNT	NUMBER		Number of audit records or audit files that were deleted at the time of the event
WAS_FORCED	VARCHAR2(3)		Indicates whether a forced cleanup occurred (YES) or (NO); forced cleanup bypasses the last archive timestamp set

14

DBMS_AUDIT_MGMT PL/SQL Package

This chapter contains:

- About Using the DBMS_AUDIT_MGMT PL/SQL Package
- DBMS_AUDIT_MGMT PL/SQL Package Security Model
- DBMS_AUDIT_MGMT PL/SQL Package Constants
- Summary of DBMS_AUDIT_MGMT PL/SQL Package Subprograms

See Also:

- Section 4.8 for the general steps you must take to purge audit trail data
- Chapter 13, "DBMS_AUDIT_MGMT Data Dictionary Views," for `DBMS_AUDIT_MGMT`-specific data dictionary views

14.1 About Using the DBMS_AUDIT_MGMT PL/SQL Package

The `DBMS_AUDIT_MGMT` PL/SQL package provides a set of subprograms that you can use to manage the Oracle Database audit trail data. It enables you to:

- Archive and purge (clean) the audit trail data for all of the supported audit trail formats.
- Move the database audit trail tables out of the `SYSTEM` tablespace to a different tablespace. This improves performance and enables you to dedicate an optimized tablespace for audit records.
- For the operating system audit trail, set a maximum size and age of the file before a new operating system audit trial file is created.
- For the database audit trail, set a record batch size in which records are deleted from audit trail tables.
- Set an archive timestamp for archived audit records, and then delete audit trail records based on the last archive timestamp. The last archive timestamp indicates when the audit records were last archived.
- Configure and schedule periodic purge jobs to delete audit records.
- Diagnose errors by using trace files.

14.2 DBMS_AUDIT_MGMT PL/SQL Package Security Model

All DBMS_AUDIT_MGMT subprograms require the user to have EXECUTE privilege on the DBMS_AUDIT_MGMT package. The SYSDBA privilege has EXECUTE privileges on the package by default.

Only audit administrators should have EXECUTE privileges over the DBMS_AUDIT_MGMT package.

14.3 DBMS_AUDIT_MGMT PL/SQL Package Constants

The DBMS_AUDIT_MGMT package defines several enumerated constants that should be used for specifying parameter values. Enumerated constants must be prefixed with the package name, for example, DBMS_AUDIT_MGMT.AUDIT_TRAIL_AUD_STD.

The DBMS_AUDIT_MGMT package uses the constants shown in the following tables:

- Table 14–1, " DBMS_AUDIT_MGMT Constants - Types of Audit Trails"
- Table 14–2, " DBMS_AUDIT_MGMT Constants - Audit Trail Properties"
- Table 14–3, " DBMS_AUDIT_MGMT Constants - Purge Job Status"
- Table 14–4, " DBMS_AUDIT_MGMT Constants - Trace Level Values"

Table 14–1 lists the audit trail type constants. Audit trails are classified by where Oracle Database writes the audit records: to database tables, operating system files, or XML files.

Table 14–1 DBMS_AUDIT_MGMT Constants - Types of Audit Trails

Constant	Data Type	Value	Description
AUDIT_TRAIL_ALL	PLS_INTEGER	15	All audit trail types. This includes the standard database audit trail (SYS.AUD$ and SYS.FGA_LOG$ tables), operating system (OS) audit trail, and XML audit trail.
AUDIT_TRAIL_AUD_STD	PLS_INTEGER	1	Standard database audit records in the SYS.AUD$ table
AUDIT_TRAIL_DB_STD	PLS_INTEGER	3	Both standard audit (SYS.AUD$) and FGA audit (SYS.FGA_LOG$) records
AUDIT_TRAIL_FGA_STD	PLS_INTEGER	2	Standard database fine-grained auditing (FGA) records in the SYS.FGA_LOG$ table
AUDIT_TRAIL_FILES	PLS_INTEGER	12	Both operating system (OS) and XML audit trails
AUDIT_TRAIL_OS	PLS_INTEGER	4	Operating system audit trail. This refers to the audit records stored in operating system files.
AUDIT_TRAIL_XML	PLS_INTEGER	8	XML audit trail. This refers to the audit records stored in XML files.

Table 14–2 lists the constants related to audit trail properties. Audit trail properties determine the audit configuration settings.

Table 14–2 DBMS_AUDIT_MGMT Constants - Audit Trail Properties

Constant	Type	Value	Description
CLEANUP_INTERVAL	PLS_INTEGER	21	Interval, in hours, after which the cleanup job is called to clear audit records in the specified audit trail
DB_DELETE_BATCH_SIZE	PLS_INTEGER	23	Specifies the batch size to be used for deleting audit records in database audit tables. The audit records are deleted in batches of size equal to DB_DELETE_BATCH_SIZE.
OS_FILE_MAX_AGE	PLS_INTEGER	17	Specifies the maximum number of days for which an operating system (OS) or XML audit file can be kept open before a new audit file gets created
OS_FILE_MAX_SIZE	PLS_INTEGER	16	Specifies the maximum size to which an operating system (OS) or XML audit file can grow before a new file is opened

Table 14–3 lists the constants related to purge job status values. The audit trail purge job cleans up the audit trail.

Table 14–3 DBMS_AUDIT_MGMT Constants - Purge Job Status

Constant	Type	Value	Description
PURGE_JOB_DISABLE	PLS_INTEGER	32	Disables a purge job
PURGE_JOB_ENABLE	PLS_INTEGER	31	Enables a purge job

Table 14–4 lists the constants related to trace level values. The DBMS_AUDIT_MGMT package enables you to trace operations for diagnostic purposes.

Table 14–4 DBMS_AUDIT_MGMT Constants - Trace Level Values

Constant	Type	Value	Description
TRACE_LEVEL_DEBUG	PLS_INTEGER	1	Logs detailed debug messages
TRACE_LEVEL_ERROR	PLS_INTEGER	2	Logs only error messages

14.4 Summary of DBMS_AUDIT_MGMT PL/SQL Package Subprograms

Table 14–5 lists the DBMS_AUDIT_MGMT package subprograms.

Table 14–5 DBMS_AUDIT_MGMT Package Subprograms

Subprogram	Description
CLEAN_AUDIT_TRAIL Procedure on page 14-4	Deletes audit trail records that have been archived
CLEAR_AUDIT_TRAIL_PROPERTY Procedure on page 14-5	Clears the value for the audit trail property that you specify
CLEAR_LAST_ARCHIVE_TIMESTAMP Procedure on page 14-6	Clears the timestamp set by the SET_LAST_ARCHIVE_TIMESTAMP procedure
CREATE_PURGE_JOB Procedure on page 14-7	Creates a purge job for periodically deleting the audit trail records

Table 14–5 (Cont.) DBMS_AUDIT_MGMT Package Subprograms

Subprogram	Description
DEINIT_CLEANUP Procedure on page 14-8	Undoes the setup and initialization performed by the INIT_CLEANUP procedure
DROP_PURGE_JOB Procedure on page 14-9	Drops the purge job that was created using the CREATE_PURGE_JOB procedure
GET_AUDIT_COMMIT_DELAY Function on page 14-9	Returns the number of seconds allowed for a COMMIT operation to take place when an audit record is written to the database
INIT_CLEANUP Procedure on page 14-10	Sets up the audit management infrastructure and sets a default cleanup interval for the audit trail records
IS_CLEANUP_INITIALIZED Function on page 14-11	Checks to see if the INIT_CLEANUP procedure has been run for an audit trail type
SET_AUDIT_TRAIL_LOCATION Procedure on page 14-11	Moves the audit trail tables from their current tablespace to a user-specified tablespace
SET_AUDIT_TRAIL_PROPERTY Procedure on page 14-12	Sets the audit trail properties for the audit trail type that you specify
SET_DEBUG_LEVEL Procedure on page 14-14	Sets the trace level for the DBMS_AUDIT_MGMT package
SET_LAST_ARCHIVE_TIMESTAMP Procedure on page 14-15	Sets a timestamp indicating when the audit records were last archived
SET_PURGE_JOB_INTERVAL Procedure on page 14-16	Sets the interval at which the CLEAN_AUDIT_TRAIL is called for the purge job that you specify
SET_PURGE_JOB_STATUS Procedure on page 14-17	Enables or disables the purge job that you specify

14.4.1 CLEAN_AUDIT_TRAIL Procedure

The CLEAN_AUDIT_TRAIL procedure deletes audit trail records that have been archived.

Typically, you run the CLEAN_AUDIT_TRAIL procedure is after you run the SET_LAST_ARCHIVE_TIMESTAMP procedure, which sets the last archived timestamp for the audit records.

Syntax

```
DBMS_AUDIT_MGMT.CLEAN_AUDIT_TRAIL(
   audit_trail_type         IN PLS_INTEGER,
   use_last_arch_timestamp  IN BOOLEAN DEFAULT TRUE) ;
```

Parameters

Parameter	Description
audit_trail_type	Enter the audit trail type for which the cleanup operation needs to be performed. Table 14–1 on page 14-2 lists audit trail types.

Parameter	Description
use_last_arch_timestamp	Specify whether the last archived timestamp should be used to determine the records that should be deleted. A value of TRUE indicates that only audit records created before the last archive timestamp should be deleted. A value of FALSE indicates that all audit records should be deleted. The default value is TRUE.

Usage Notes

None

Examples

The following example calls the CLEAN_AUDIT_TRAIL procedure to clean up the operating system (OS) audit trail records that were created before the last archive timestamp.

```
BEGIN
DBMS_AUDIT_MGMT.CLEAN_AUDIT_TRAIL(
   audit_trail_type => DBMS_AUDIT_MGMT.AUDIT_TRAIL_OS,
   use_last_arch_timestamp => TRUE);
END;
/
```

14.4.2 CLEAR_AUDIT_TRAIL_PROPERTY Procedure

The CLEAR_AUDIT_TRAIL_PROPERTY procedure clears the value for the audit trail property that is specified. Audit trail properties are set using the SET_AUDIT_TRAIL_PROPERTY procedure.

The CLEAR_AUDIT_TRAIL_PROPERTY procedure can optionally reset the property value to its default value through the use_default_values parameter.

Syntax

```
DBMS_AUDIT_MGMT.CLEAR_AUDIT_TRAIL_PROPERTY(
   audit_trail_type        IN PLS_INTEGER,
   audit_trail_property    IN PLS_INTEGER,
   use_default_values      IN BOOLEAN DEFAULT FALSE) ;
```

Parameters

Parameter	Description
audit_trail_type	Enter the audit trail type for which the property needs to be cleared. Table 14–1 on page 14-2 lists the audit trail types.
audit_trail_property	Enter the audit trail property whose value needs to be cleared. You cannot clear the value for the CLEANUP_INTERVAL property. Table 14–2 on page 14-3 lists the audit trail properties.

Parameter	Description
use_default_values	Specify whether the default value of the audit_trail_property should be used in place of the cleared value. A value of TRUE causes the default value of the parameter to be used. A value of FALSE causes the audit_trail_property to have no value. The default value for this parameter is FALSE.

Usage Notes

- You can use this procedure to clear the value for an audit trail property that you do not wish to use. For example, if you do not want a restriction on the operating system audit file size, then you can use this procedure to reset the OS_FILE_MAX_SIZE property.

  You can also use this procedure to reset an audit trail property to its default value. You need to set use_default_values to TRUE when invoking the procedure.
- The DB_DELETE_BATCH_SIZE property needs to be individually cleared for the AUDIT_TRAIL_AUD_STD and AUDIT_TRAIL_FGA_STD audit trail types. You cannot clear this property collectively using the AUDIT_TRAIL_DB_STD and AUDIT_TRAIL_ALL audit trail types.
- You cannot clear the value for the CLEANUP_INTERVAL property.

Examples

The following example calls the CLEAR_AUDIT_TRAIL_PROPERTY procedure to clear the value for the audit trail property OS_FILE_MAX_SIZE because the procedure uses a value of FALSE for the USE_DEFAULT_VALUES parameter. This means that the OS_FILE_MAX_SIZE property will no longer determine the size of the operating system (OS) audit files.

```
BEGIN
DBMS_AUDIT_MGMT.CLEAR_AUDIT_TRAIL_PROPERTY(
   AUDIT_TRAIL_TYPE        =>  DBMS_AUDIT_MGMT.AUDIT_TRAIL_OS,
   AUDIT_TRAIL_PROPERTY    =>  DBMS_AUDIT_MGMT.OS_FILE_MAX_SIZE,
   USE_DEFAULT_VALUES      =>  FALSE );
END;
/
```

14.4.3 CLEAR_LAST_ARCHIVE_TIMESTAMP Procedure

The CLEAR_LAST_ARCHIVE_TIMESTAMP procedure clears the timestamp set by the SET_LAST_ARCHIVE_TIMESTAMP procedure.

Syntax

```
DBMS_AUDIT_MGMT.CLEAR_LAST_ARCHIVE_TIMESTAMP(
   audit_trail_type     IN PLS_INTEGER,
   rac_instance_number  IN PLS_INTEGER DEFAULT 0) ;
```

Parameters

Parameter	Description
audit_trail_type	Enter the audit trail type for which the timestamp needs to be cleared. Table 14–1 on page 14-2 lists the audit trail types.

Parameter	Description
rac_instance_number	Enter the instance number for the Oracle Real Application Clusters (Oracle RAC) instance. The default value is 0, which is used for the database audit trail type. The rac_instance_number is not relevant for the database audit trail type, as the database audit trail tables are shared by all Oracle RAC instances.

Usage Notes

None

Example

The following example calls the CLEAR_LAST_ARCHIVE_TIMESTAMP procedure to clear the timestamp value for the operating system (OS) audit trail type.

```
BEGIN
DBMS_AUDIT_MGMT.CLEAR_LAST_ARCHIVE_TIMESTAMP(
   audit_trail_type     =>  DBMS_AUDIT_MGMT.AUDIT_TRAIL_OS,
   rac_instance_number  =>  1 /* single instance database */);
END;
/
```

14.4.4 CREATE_PURGE_JOB Procedure

The CREATE_PURGE_JOB procedure creates a purge job for periodically deleting the audit trail records. The procedure can use the timestamp value set by the SET_LAST_ARCHIVE_TIMESTAMP procedure to determine the records to be deleted.

This procedure performs the cleanup operation on an intervals that you specify. It calls the CLEAN_AUDIT_TRAIL procedure to perform the cleanup operation.

The SET_PURGE_JOB_INTERVAL procedure is used to modify the frequency of the purge job.

The SET_PURGE_JOB_STATUS procedure is used to enable or disable the purge job.

The DROP_PURGE_JOB procedure is used to drop a purge job created with the CREATE_PURGE_JOB procedure.

Syntax

```
DBMS_AUDIT_MGMT.CREATE_PURGE_JOB(
   audit_trail_type            IN PLS_INTEGER,
   audit_trail_purge_interval  IN PLS_INTEGER,
   audit_trail_purge_name      IN VARCHAR2,
   use_last_arch_timestamp     IN BOOLEAN DEFAULT TRUE) ;
```

Parameters

Parameter	Description
audit_trail_type	Enter the audit trail type for which the purge job needs to be created. Table 14–1 on page 14-2 lists the audit trail types.
audit_trail_purge_interval	Enter the interval, in hours, at which the clean up procedure is called. A lower value means that the cleanup is performed more often.
audit_trail_purge_name	A name to identify the purge job

Parameter	Description
use_last_arch_timestamp	Specify whether the last archived timestamp should be used to determine the records that should be deleted. A value of TRUE indicates that only audit records created before the last archive timestamp should be deleted. A value of FALSE indicates that all audit records should be deleted. The default value is TRUE.

Usage Notes

Use this procedure to schedule the CLEAN_AUDIT_TRAIL procedure for your audit records.

Examples

The following example calls the CREATE_PURGE_JOB procedure to create a cleanup job called CLEANUP for all audit trail types. It sets the audit_trail_purge_interval parameter to 100 to invoke that the cleanup job every 100 hours. It also sets the use_last_arch_timestamp parameter value to TRUE, so that all audit records older than the last archive timestamp are deleted.

```
BEGIN
DBMS_AUDIT_MGMT.CREATE_PURGE_JOB(
  audit_trail_type => DBMS_AUDIT_MGMT.AUDIT_TRAIL_ALL,
  audit_trail_purge_interval => 100 /* hours */,
  audit_trail_purge_name => 'CLEANUP',
  use_last_arch_timestamp => TRUE);
END;
/
```

14.4.5 DEINIT_CLEANUP Procedure

The DEINIT_CLEANUP procedure undoes the setup and initialization performed by the INIT_CLEANUP procedure. The DEINIT_CLEANUP procedure clears the value of the default_cleanup_interval parameter. However, it does not move the audit trail tables back to their original location.

Syntax

```
DBMS_AUDIT_MGMT.DEINIT_CLEANUP(
   audit_trail_type  IN PLS_INTEGER) ;
```

Parameters

Parameter	Description
audit_trail_type	Enter the audit trail type for which the procedure needs to be called. Table 14–1 on page 14-2 lists the audit trail types.

Usage Notes

You can change the default_cleanup_interval later using the SET_AUDIT_TRAIL_PROPERTY procedure.

Examples

The following example clears the `default_cleanup_interval` parameter setting for the standard database audit trail:

```
BEGIN
DBMS_AUDIT_MGMT.DEINIT_CLEANUP(
  AUDIT_TRAIL_TYPE  => DBMS_AUDIT_MGMT.AUDIT_TRAIL_AUD_STD);
END;
/
```

14.4.6 DROP_PURGE_JOB Procedure

The `DROP_PURGE_JOB` procedure drops the purge job that was created using the `CREATE_PURGE_JOB` procedure. The name of the purge job is passed as an argument.

Syntax

```
DBMS_AUDIT_MGMT.DROP_PURGE_JOB(
   audit_trail_purge_name    IN VARCHAR2) ;
```

Parameter

Parameter	Description
`audit_trail_purge_name`	Enter the name of the purge job to be deleted. This is the purge job name that you specified with the `CREATE_PURGE_JOB` procedure.

Usage Notes

None

Examples

The following example calls the `DROP_PURGE_JOB` procedure to drop the purge job called `CLEANUP`.

```
BEGIN
DBMS_AUDIT_MGMT.DROP_PURGE_JOB(
  AUDIT_TRAIL_PURGE_NAME  => 'CLEANUP');
END;
/
```

14.4.7 GET_AUDIT_COMMIT_DELAY Function

The `GET_AUDIT_COMMIT_DELAY` function returns the number of seconds allowed for a `COMMIT` operation to take place when an audit record is written to the database. The default time is 5 seconds. If the `COMMIT` operation exceeds this time, then Oracle Database writes each audit record to an operating system file, even if the `AUDIT_TRAIL` initialization parameter is set to `DB` or `DB, EXTENDED`.

Syntax

```
DBMS_AUDIT_MGMT.GET_AUDIT_COMMIT_DELAY
  RETURN NUMBER;
```

Parameters

None

Usage Notes

None

Examples

None

14.4.8 INIT_CLEANUP Procedure

The INIT_CLEANUP procedure sets up the audit management infrastructure and sets a default cleanup interval for the audit trail records. The procedure also moves the audit trail tables out of the SYSTEM tablespace.

Moving the audit trail tables out of the SYSTEM tablespace enhances overall database performance. The INIT_CLEANUP procedure moves the audit trail tables to the SYSAUX tablespace. If the SET_AUDIT_TRAIL_LOCATION procedure has already moved the audit tables elsewhere, then they are not moved back to the SYSAUX tablespace.

The SET_AUDIT_TRAIL_LOCATION procedure enables you to specify an alternate target tablespace for the database audit tables.

The INIT_CLEANUP procedure is currently not relevant for the AUDIT_TRAIL_OS, AUDIT_TRAIL_XML, and AUDIT_TRAIL_FILES audit trail types. No preliminary set up is required for these audit trail types.

This procedure also sets a default cleanup interval for the audit trail records.

Syntax

```
DBMS_AUDIT_MGMT.INIT_CLEANUP(
   audit_trail_type          IN PLS_INTEGER,
   default_cleanup_interval  IN PLS_INTEGER);
```

Parameters

Parameter	Description
audit_trail_type	Enter the audit trail type for which the clean up operation needs to be initialized. Table 14–1 on page 14-2 lists audit trail types.
default_cleanup_interval	Enter the default time interval, in hours, after which the cleanup procedure should be called. The minimum value is 1 and the maximum is 999.

Usage Notes

- This procedure may involve data movement across tablespaces. This can be a resource-intensive operation, especially if your database audit trail tables are already populated. Oracle recommends that you invoke the procedure during nonpeak hours.
- You should ensure that the SYSAUX tablespace, into which the audit trail tables are being moved, has sufficient space to accommodate the audit trail tables. You should also optimize the SYSAUX tablespace for frequent write operations.
- You can change the default_cleanup_interval later using the SET_AUDIT_TRAIL_PROPERTY procedure.

Examples

The following example calls the INIT_CLEANUP procedure to set a default_cleanup_interval of 12 hours for all audit trail types:

```
BEGIN
DBMS_AUDIT_MGMT.INIT_CLEANUP(
              audit_trail_type => DBMS_AUDIT_MGMT.AUDIT_TRAIL_ALL,
      default_cleanup_interval => 12 /* hours */);
END;
/
```

14.4.9 IS_CLEANUP_INITIALIZED Function

The IS_CLEANUP_INITIALIZED function checks if the INIT_CLEANUP procedure has been run for an audit trail type. The IS_CLEANUP_INITIALIZED function returns TRUE if the procedure has already been run for the audit trail type. It returns FALSE if the procedure has not been run for the audit trail type.

This function does not apply to the AUDIT_TRAIL_OS, AUDIT_TRAIL_XML, and AUDIT_TRAIL_FILES audit trail types. The function always returns TRUE for these audit trail types. No preliminary set up is required for these audit trail types.

Syntax

```
DBMS_AUDIT_MGMT.IS_CLEANUP_INITIALIZED(
   audit_trail_type  IN PLS_INTEGER)
 RETURN BOOLEAN;
```

Parameters

Parameter	Description
audit_trail_type	Enter the audit trail type for which the function needs to be called. Table 14–1 on page 14-2 lists the audit trail types.

Usage Notes

None

Examples

The following example checks if the standard database audit trail type has been initialized for a cleanup operation. If the audit trail type has not been initialized, then the example calls the INIT_CLEANUP procedure to initialize the audit trail type.

```
BEGIN
 IF
   NOT DBMS_AUDIT_MGMT.IS_CLEANUP_INITIALIZED(DBMS_AUDIT_MGMT.AUDIT_TRAIL_AUD_STD)
 THEN
   DBMS_AUDIT_MGMT.INIT_CLEANUP(
      audit_trail_type => DBMS_AUDIT_MGMT.AUDIT_TRAIL_AUD_STD,
      default_cleanup_interval => 12 /* hours */);
 END IF;
END;
/
```

14.4.10 SET_AUDIT_TRAIL_LOCATION Procedure

The SET_AUDIT_TRAIL_LOCATION procedure moves the audit trail tables from their current tablespace to a user-specified tablespace.

The SET_AUDIT_TRAIL_LOCATION procedure does not apply to the AUDIT_TRAIL_OS, AUDIT_TRAIL_XML, and AUDIT_TRAIL_FILES audit trail types. The AUDIT_FILE_DEST initialization parameter can be used to specify the destination directory for these audit trail types.

Syntax

```
DBMS_AUDIT_MGMT.SET_AUDIT_TRAIL_LOCATION(
   audit_trail_type           IN PLS_INTEGER,
   audit_trail_location_value  IN VARCHAR2) ;
```

Parameters

Parameter	Description
audit_trail_type	Enter the audit trail type for which the audit trail location needs to be set. Table 14–1 on page 14-2 lists audit trail types.
audit_trail_location_value	Enter the target location or tablespace for the audit trail records

Usage Notes

- This procedure involves data movement across tablespaces. This can be a resource-intensive operation, especially if your database audit trail tables are already populated. Oracle recommends that you run the procedure during nonpeak hours.
- You should ensure that the target tablespace, into which the audit trail tables are being moved, has sufficient space to accommodate the audit trail tables. You should also optimize the target tablespace for frequent write operations. For more information, see *Oracle Database Performance Tuning Guide* and *Oracle Database Administrator's Guide*.

Examples

The following example moves the database audit trail tables, AUD$ and FGA_LOG$, from the current tablespace to a user-created tablespace called RECORDS:

```
BEGIN
DBMS_AUDIT_MGMT.SET_AUDIT_TRAIL_LOCATION(
       audit_trail_type => DBMS_AUDIT_MGMT.AUDIT_TRAIL_DB_STD,
       audit_trail_location_value =>  'RECORDS');
END;
/
```

14.4.11 SET_AUDIT_TRAIL_PROPERTY Procedure

The SET_AUDIT_TRAIL_PROPERTY procedure sets the audit trail properties for the audit trail type that you specify.

The procedure sets properties such as OS_FILE_MAX_SIZE and OS_FILE_MAX_AGE for operating system (OS) and XML audit trail types. These properties determine the maximum size and age of an audit trail file before a new audit trail file is created.

The procedure sets properties such as DB_DELETE_BATCH_SIZE and CLEANUP_INTERVAL for the database audit trail type. DB_DELETE_BATCH_SIZE specifies the batch size in which records are deleted from audit trail tables. This ensures that if a cleanup operation is interrupted midway, the process does not need to start afresh the

next time it is invoked. This is because all batches before the last processed batch are already deleted.

The CLEANUP_INTERVAL property value specifies the frequency, in hours, with which the cleanup procedure is called.

Syntax

```
DBMS_AUDIT_MGMT.SET_AUDIT_TRAIL_PROPERTY(
   audit_trail_type            IN PLS_INTEGER,
   audit_trail_property        IN PLS_INTEGER,
   audit_trail_property_value  IN PLS_INTEGER) ;
```

Parameters

Parameter	Description
audit_trail_type	Enter the audit trail type for which the property needs to be set. Table 14–1 on page 14-2 lists audit trail types.
audit_trail_property	Enter the audit trail property that is being set. Table 14–2 on page 14-3 lists audit trail properties.
audit_trail_property_value	Enter the value of the property specified using audit_trail_property. The following are valid values for audit trail properties: ■ OS_FILE_MAX_SIZE can have a minimum value of 1 and maximum value of 2000000. The default value is 10000. OS_FILE_MAX_SIZE is measured in kilobytes (KB). ■ OS_FILE_MAX_AGE can have a minimum value of 1 and a maximum value of 497. The default value is 5. OS_FILE_MAX_AGE is measured in days. ■ DB_DELETE_BATCH_SIZE can have a minimum value of 100 and a maximum value of 1000000. The default value is 10000. DB_DELETE_BATCH_SIZE is measured as the number of audit records that are deleted in one batch. ■ CLEANUP_INTERVAL can have a minimum value of 1 and a maximum value of 999. The default value is set using the INIT_CLEANUP procedure. CLEANUP_INTERVAL is measured in hours.

Usage Notes

- The audit trail properties for which you do not explicitly set values use their default values.
- If you have set both the OS_FILE_MAX_SIZE and OS_FILE_MAX_AGE properties for an operating system (OS) or XML audit trail type, then a new audit trail file is created, depending on which of these two limits is reached first.

  For example, suppose OS_FILE_MAX_SIZE is 10000 and OS_FILE_MAX_AGE is 5. If the operating system audit file is already more than 5 days old and has a size of 9000 KB, then a new audit file is opened, because one of the limits has been reached.
- You must individually set the DB_DELETE_BATCH_SIZE property for the AUDIT_TRAIL_AUD_STD and AUDIT_TRAIL_FGA_STD audit trail types. You cannot set this property collectively using the AUDIT_TRAIL_DB_STD and AUDIT_TRAIL_ALL audit trail types.

Examples

The following example calls the SET_AUDIT_TRAIL_PROPERTY procedure to set the OS_FILE_MAX_SIZE property for the operating system (OS) audit trail. It sets this property value to 102400, so that a new audit file is created every time the current audit file size reaches 100 MB.

```
BEGIN
DBMS_AUDIT_MGMT.SET_AUDIT_TRAIL_PROPERTY(
       audit_trail_type => DBMS_AUDIT_MGMT.AUDIT_TRAIL_OS,
       audit_trail_property  =>  DBMS_AUDIT_MGMT.OS_FILE_MAX_SIZE,
       audit_trail_property_value =>  102400 /* 100MB*/ );
END;
/
```

The following example calls the SET_AUDIT_TRAIL_PROPERTY procedure to set the OS_FILE_MAX_AGE property for the operating system (OS) audit trail. It sets this property value to 5, so that a new audit file is created every sixth day.

```
BEGIN
DBMS_AUDIT_MGMT.SET_AUDIT_TRAIL_PROPERTY(
       audit_trail_type => DBMS_AUDIT_MGMT.AUDIT_TRAIL_OS,
       audit_trail_property  =>  DBMS_AUDIT_MGMT.OS_FILE_MAX_AGE,
       audit_trail_property_value  =>  5 /* days */);
END;
/
```

The following example calls the SET_AUDIT_TRAIL_PROPERTY procedure to set the DB_DELETE_BATCH_SIZE property for the AUDIT_TRAIL_AUD_STD audit trail. It sets this property value to 100000. Thus, during a cleanup operation, audit records are deleted from the SYS.AUD$ table in batches of 100000 records.

```
BEGIN
DBMS_AUDIT_MGMT.SET_AUDIT_TRAIL_PROPERTY(
       audit_trail_type => DBMS_AUDIT_MGMT.AUDIT_TRAIL_AUD_STD,
       audit_trail_property => DBMS_AUDIT_MGMT.DB_DELETE_BATCH_SIZE,
       audit_trail_property_value => 100000 /* delete batch size */);
END;
/
```

14.4.12 SET_DEBUG_LEVEL Procedure

The SET_DEBUG_LEVEL procedure sets the trace level for the DBMS_AUDIT_MGMT package. The default trace level, TRACE_LEVEL_ERROR, logs only the error messages as trace messages. The debug trace level, TRACE_LEVEL_DEBUG, logs detailed debug messages.

Syntax

```
DBMS_AUDIT_MGMT.SET_DEBUG_LEVEL(
   debug_level IN PLS_INTEGER DEFAULT TRACE_LEVEL_ERROR);
```

Parameters

Parameter	Description
debug_level	Enter the trace level. TRACE_LEVEL_ERROR logs only the error messages as trace messages. TRACE_LEVEL_DEBUG logs detailed debug messages.

Usage Notes

None

Examples

The following example calls the SET_DEBUG_LEVEL procedure to enable enhanced debugging.

```
BEGIN
DBMS_AUDIT_MGMT.SET_DEBUG_LEVEL(
  debug_level   => DBMS_AUDIT_MGMT.TRACE_LEVEL_DEBUG);
END;
/
```

14.4.13 SET_LAST_ARCHIVE_TIMESTAMP Procedure

The SET_LAST_ARCHIVE_TIMESTAMP procedure sets a timestamp indicating when the audit records were last archived. The audit administrator provides the timestamp to be attached to the audit records. The CLEAN_AUDIT_TRAIL procedure uses this timestamp to determine the audit records to be deleted.

Syntax

```
DBMS_AUDIT_MGMT.SET_LAST_ARCHIVE_TIMESTAMP(
   audit_trail_type     IN PLS_INTEGER,
   last_archive_time    IN TIMESTAMP,
   rac_instance_number  IN PLS_INTEGER DEFAULT 0) ;
```

Parameters

Parameter	Description
audit_trail_type	Enter the audit trail type for which the timestamp needs to be set. Table 14–1 on page 14-2 lists audit trail types.
last_archive_time	Enter the TIMESTAMP value to be attached to the audit records. This indicates the last time when the audit records were archived.
rac_instance_number	Enter the instance number for the Oracle Real Application Clusters (Oracle RAC) instance. The default value is 0, which is used for the database audit trail type. The rac_instance_number parameter is not relevant for the database audit trail type, as the database audit trail tables are shared by all Oracle RAC instances.

Usage Notes

- You must set the last_archive_time parameter using Coordinated Universal Time (UTC) for the AUDIT_TRAIL_AUD_STD and AUDIT_TRAIL_FGA_STD audit trail types. This is because the database audit trails store the timestamps in UTC. UTC is also known as Greenwich Mean Time (GMT).
- You must set the last_archive_time parameter using the local time zone time for the AUDIT_TRAIL_OS and AUDIT_TRAIL_XML audit trail types. This is because the operating system audit records are stored as files that use the local time zone for their last modification timestamps.
- If you are using and Oracle RAC database, Oracle recommends that you use the Network Time Protocol (NTP) to synchronize individual Oracle RAC nodes.

Examples

The following example uses the SET_LAST_ARCHIVE_TIMESTAMP procedure to set the last archive timestamp for the operating system (OS) audit trail type. It uses the TO_TIMESTAMP function to convert a character string into a timestamp value.

```
BEGIN
DBMS_AUDIT_MGMT.SET_LAST_ARCHIVE_TIMESTAMP(
   audit_trail_type => DBMS_AUDIT_MGMT.AUDIT_TRAIL_OS,
   last_archive_time => TO_
TIMESTAMP('10-SEP-0714:10:10.0','DD-MON-RRHH24:MI:SS.FF'),
   rac_instance_number => 1 /* single instance database */);
END;
/
```

14.4.14 SET_PURGE_JOB_INTERVAL Procedure

The SET_PURGE_JOB_INTERVAL procedure sets the interval at which the CLEAN_AUDIT_TRAIL procedure is called for the purge job specified. The purge job must have already been created using the CREATE_PURGE_JOB procedure.

Syntax

```
DBMS_AUDIT_MGMT.SET_PURGE_JOB_INTERVAL(
   audit_trail_purge_name      IN VARCHAR2,
   audit_trail_interval_value  IN PLS_INTEGER) ;
```

Parameters

Parameter	Description
audit_trail_purge_name	Enter the name of the purge job for which the interval is being set. This is the purge job name that you specified with the CREATE_PURGE_JOB procedure.
audit_trail_interval_value	Enter the interval, in hours, at which the cleanup procedure should be called. This value modifies the audit_trail_purge_interval parameter set using the CREATE_PURGE_JOB procedure.

Usage Notes

Use this procedure to modify the audit_trail_purge_interval parameter set using the CREATE_PURGE_JOB procedure.

Examples

The following example calls the SET_PURGE_JOB_INTERVAL procedure to change the frequency at which the purge job called CLEANUP is invoked. The new interval is set to 24 hours.

```
BEGIN
DBMS_AUDIT_MGMT.SET_PURGE_JOB_INTERVAL(
  AUDIT_TRAIL_PURGE_NAME       => 'CLEANUP',
  AUDIT_TRAIL_INTERVAL_VALUE   => 24 );
END;
/
```

14.4.15 SET_PURGE_JOB_STATUS Procedure

The SET_PURGE_JOB_STATUS procedure enables or disables the specified purge job. The purge job must have already been created using the CREATE_PURGE_JOB procedure.

Syntax

```
DBMS_AUDIT_MGMT.SET_PURGE_JOB_STATUS(
   audit_trail_purge_name    IN VARCHAR2,
   audit_trail_status_value  IN PLS_INTEGER) ;
```

Parameters

Parameter	Description
audit_trail_purge_name	Enter the name of the purge job for which the status is being set. This is the purge job name that you specified with the CREATE_PURGE_JOB procedure.
audit_trail_status_value	Enter one of the values specified in Table 14–3 on page 14-3. The value PURGE_JOB_ENABLE enables the specified purge job. The value PURGE_JOB_DISABLE disables the specified purge job.

Usage Notes

None

Examples

The following example calls the SET_PURGE_JOB_STATUS procedure to enable the CLEANUP purge job.

```
BEGIN
DBMS_AUDIT_MGMT.SET_PURGE_JOB_STATUS(
  audit_trail_purge_name      => 'CLEANUP',
  audit_trail_status_value    => DBMS_AUDIT_MGMT.PURGE_JOB_ENABLE);
END;
/
```

A

Troubleshooting an Oracle Audit Vault System

This appendix contains:

- Location of Audit Vault Server Log and Error Files
- Location of Audit Vault Collection Agent Log and Error Files
- Troubleshooting Tips

A.1 Location of Audit Vault Server Log and Error Files

Table A–1 describes the Audit Vault Server log and error files. These files are located in the Audit Vault Server `$ORACLE_HOME/av/log` directory. They contain important information about the return status of commands and operations. Use this information to diagnose problems.

Table A–1 Name and Description of Audit Vault Server Log and Error Files

File Name	Description
`agent.err`	Contains a log of errors encountered in collection agent initialization. You can delete this file at any time.
`agent.out`	Contains a log of all primary collection agent-related operations and activity. You can delete this file at any time.
`avca.log`	Contains a log of all `AVCA` and `AVCTL` commands that have been run and the results of running each command. You can only delete this file only after you have shut down the Audit Vault Server.
`av_client-%g.log.n`	Contains a log of the collection agent operations and any errors returned from those operations. You can delete this file at any time. The *%g* is a generation number that starts from 0 (zero) and increases once the file size reaches the 100 MB limit. A concurrent existence of this file is indicated by a *.n* suffix appended to the file type name, such as `av_client-%g.log.n`, where *n* is an integer issued in sequence (for example, `av_client-0.log.1`).
`avorcldb.log`	Contains a log of all `AVORCLDB` commands that have been run and the results of running each command. You can delete this file at any time.

Table A–1 (Cont.) Name and Description of Audit Vault Server Log and Error Files

File Name	Description

DB2DB-%g.log	Contains a log of all AVDB2DB commands that have been run and the results of running each command. You can delete this file at any time. The *%g* is a generation number that starts from 0 (zero) and increases once the file size reaches the 100 MB limit. To enable detailed logging of AVDB2DB commands, restart Oracle Audit Vault from the Audit Vault Server with the log level set to debug, as follows:
	`avctl stop_av -loglevel debug`
	`avctl start_av -loglevel debug`

MSSQLDB-%g.log	Contains a log of all AVMSSQLDB commands that have been run and the results of running each command. You can delete this file at any time. The *%g* is a generation number that starts from 0 (zero) and increases once the file size reaches the 100 MB limit. To enable detailed logging of AVMSSQLDB commands, restart Oracle Audit Vault from the Audit Vault Server with the log level set to debug, as follows:
	`avctl stop_av -loglevel debug`
	`avctl start_av -loglevel debug`

SYBDB-%g.log	Contains a log of all AVSYBDB commands that have been run and the results of running each command. You can delete this file at any time. The *%g* is a generation number that starts from 0 (zero) and increases once the file size reaches the 100 MB limit. To enable detailed logging of AVSYBDB commands, restart Oracle Audit Vault from the Audit Vault Server with the log level set to debug, as follows:
	`avctl stop_av -loglevel debug`
	`avctl start_av -loglevel debug`

If you need to troubleshoot the Audit Vault Console, enable Oracle Enterprise Manager logging. To do so, modify the emomslogging.properties file (located in the *ORACLE_HOME*/sysman/config directory) in the Audit Vault Server home. Add the following lines to this file:

```
log4j.appender.avAppender=org.apache.log4j.RollingFileAppender
log4j.appender.avAppender.File=$ORACLE_HOME/oc4j/j2ee/OC4J_DBConsole__/log/av-application.log
log4j.appender.avAppender.Append=true
log4j.appender.avAppender.MaxFileSize =20000000
log4j.appender.avAppender.Threshold = DEBUG
log4j.appender.avAppender.layout=org.apache.log4j.PatternLayout
log4j.appender.avAppender.layout.ConversionPattern=%d [%t] %-5p %c{2} %M.%L - %m\n
log4j.category.oracle =DEBUG, avAppender
```

You can use this information to debug communication problems between the server and the collection agents.

A.2 Location of Audit Vault Collection Agent Log and Error Files

Table A–2 lists the names and a description of the Audit Vault collection agent log and error files located in the $ORACLE_HOME/av/log directory. These files contain important information about the return status of commands and operations. Use this information to diagnose problems.

Table A–2 Name and Description of Audit Vault Collection Agent Log and Error Files

File Name	Description
`agent.err`	Contains a log of all errors encountered in collection agent initialization and operation. You can delete this file at any time.
`agent.out`	Contains a log of all primary collection agent-related operations and activity. You can delete this file at any time.
`avca.log`	Contains a log of all `AVCA` and `AVCTL` commands that have been run and the results of running each command. You can delete this file at any time.
`avorcldb.log`	Contains a log of all `AVORCLDB` commands that have been run and the results of running each command. You can delete this file at any time.
`DBAUD-and-OSAUD-collector-name_source-name_source-id``.log`	Contains a log of collection operations for the Oracle Database DBAUD and OSAUD collectors. This file has no maximum size limit. To delete this log, shut down the collector, delete the log, and then restart the collector.

`non-Oracle_collector-name_source-name_collector_name-``%g.log`	Contains a log of collection operations for the MSSQLDB, SYBDB, and DB2DB collectors. The `%` symbol refers to the generation number of the log file. The maximum log file size is 100 MB.
	
	You can only delete this file after you have shut down OC4J. For example, to delete the log where `%g` is 0, you must stop OC4J. To delete the logs where`%g` is higher than 0, you can do so while OC4J is running.
	
	To increase the log level, restart OC4J on the collection agent side with the appropriate debug level, as in the following example:
	
	`avctl stop_oc4j`
	`avctl start_oc4j -loglevel error`
	
	See Section 7.12 and Section 7.16 for information about these commands, including the available log levels for OC4J.

`agent_client-%g.log.n`	Contains a log of the collection agent operations and any errors returned from those operations. The `%g` is a generation number that starts from 0 (zero) and increases once the file size reaches the 100 MB limit. A concurrent existence of this file is indicated by a `.n` suffix appended to the file type name, such as `av_client-%g.log.n`, where *n* is an integer issued in sequence, for example, `av_client-0.log.1`. You can delete this file at any time.

`DB2DB-%g.log`	Contains a log of all `AVDB2DB` commands that have been run and the results of running each command. You can delete this file at any time. The `%g` is a generation number that starts from 0 (zero) and increases once the file size reaches the 100 MB limit. To enable detailed logging of `AVDB2DB` commands, restart OC4J on the collection agent side with the log level set to `debug`, as follows:
	
	`avctl stop_oc4j`
	`avctl start_oc4j -loglevel debug`

Table A–2 (Cont.) Name and Description of Audit Vault Collection Agent Log and Error Files

File Name	Description

`MSSQLDB-%g.log`	Contains a log of all `AVMSSQLDB` commands that have been run and the results of running each command. You can delete this file at any time. The *%g* is a generation number that starts from 0 (zero) and increases once the file size reaches the 100 MB limit. To enable detailed logging of `AVMSSQLDB` commands, restart OC4J on the collection agent side with the log level set to debug, as follows:
	`avctl stop_oc4j`
	`avctl start_oc4j -loglevel debug`

`SYBDB-%g.log`	Contains a log of all `AVSYBDB` commands that have been run and the results of running each command. You can delete this file at any time. The `%g` is a generation number that starts from 0 (zero) and increases once the file size reaches the 100 MB limit. To enable detailed logging of `AVSYBDB` commands, restart OC4J on the collection agent side with the log level set to `debug`, as follows:
	`avctl stop_oc4j`
	`avctl start_oc4j -loglevel debug`

`sqlnet.log`	Contains a log of SQL*Net information.

The Oracle Audit Vault collection agent `$ORACLE_HOME/oc4j/j2ee/home/log` contains the logs generated by the collection agent OC4J. In this directory, the file `AVAgent-access.log` contains a log of requests that the collection agent receives from the Audit Vault Server. Use this information to debug communication problems between the server and the collection agent.

Failed configuration commands are located in the Audit Vault collection agent `$ORACLE_HOME/cfgtoollogs` directory, which includes the file, `configToolFailedCommands`. This file contains only the name of the failed command. See the `avca.log` or `avorcldb.log` file for additional information, including any associated errors and error messages.

A.3 Troubleshooting Tips

This section contains:

- Checking Trace Files for Detailed Information About Oracle Database Errors
- Troubleshooting Audit Vault Server
- Troubleshooting Audit Vault Collection Agent
- Troubleshooting the Audit Vault Collector
- Troubleshooting Oracle Audit Vault Console
- Troubleshooting Oracle Audit Vault in an Oracle Real Application Clusters Environment

A.3.1 Checking Trace Files for Detailed Information About Oracle Database Errors

For detailed information about the cause of an Oracle Database error message, check the trace files. The trace files also indicate problems that may have occurred with the Audit Vault data warehouse, alert, and some configuration issues. The `USER_DUMP_DEST` initialization parameter specifies the current location of the trace files. You can find the value of this parameter by issuing `SHOW PARAMETER USER_DUMP_DEST` in SQL*Plus. See *Oracle Database Performance Tuning Guide* for more information about trace files

A.3.2 Troubleshooting Audit Vault Server

Problem: Need to find the best way to tune the Audit Vault Server performance for the REDO collector.

The Audit Vault Server installation process sets the `STREAMS_POOL_SIZE` initialization parameter to 150 MB. If you plan to use the REDO collector, you must tune this parameter to maximize REDO collector performance. In an Oracle Real Application Clusters (Oracle RAC) environment, tune this parameter on all nodes, because it is uncertain where the queue will be after a database instance starts.

Solution:

Typically, after you have configured and started the REDO collector, let it run for a while. This enables the Oracle Database autotuning feature to allocate memory for the best database performance for the `STREAMS_POOL_SIZE` parameter. Using Automatic Workload Repository (AWR), check if Streams AQ has a flow control problem, such as `enqueue` being blocked. If the performance is slow (for example, only 500 records are applied per second), you may need to tune the `STREAMS_POOL_SIZE` parameter.

If you have at least 1 GB of physical memory in your Audit Vault Server system, set the `STREAMS_POOL_SIZE` parameter to 200 MB using the `ALTER SYSTEM` SQL statement, as follows:

```
ALTER SYSTEM SET STREAMS_POOL_SIZE=200;
```

The record apply rate should be 2000 records per second, which is a typical maximum rate for the REDO collector. Usually, setting the value to 200 MB is sufficient. If you are using Oracle Audit Vault in an Oracle RAC environment, set this parameter value accordingly on all nodes in the cluster, as follows:

```
ALTER SYSTEM SET STREAMS_POOL_SIZE=200 SID=avn;
```

Replace *avn* with the SID for each node in the cluster.

A.3.3 Troubleshooting Audit Vault Collection Agent

Problem: Audit Vault agent status is blank on the Windows Services Panel.

After installing Audit Vault Agent for Microsoft Windows (32-bit), configuring a source and collectors, then starting the agent on the Audit Vault Server side, you notice that the Services Panel on the Windows system where the Audit Vault collection agent resides shows that the status is blank, rather than **Started**.

Solution:

This is normal behavior for the Audit Vault collection agent on Microsoft Windows systems because the service is a short-lived process. Once the Agent service process completes its task, it exits, so the status of the service will not show as Started. However, the Audit Vault collection agent is running without problems.

Run the `avctl show_agent_status` command to check the status of the Audit Vault Agent, as follows:

```
C:\ORACLE_HOME\agent_dir\bin\avctl show_agent_status -agentname agent1
AVCTL started
Getting agent metrics...
--------------------------------
Agent is running
--------------------------------
```

```
Metrics retrieved successfully.
```

Problem: Need to debug a collection agent problem.

You want to enable debug logging while trying to diagnose an Audit Vault collection agent problem.

Solution:

1. Run the following set of AVCTL commands on the command line:

   ```
   avctl stop_oc4j
   avctl start_oc4j -loglevel debug
   ```

2. Check the log output in the Audit Vault collection agent $ORACLE_HOME/av/log directory.
3. Because debugging creates more logging and writing overhead, remember to disable it when you no longer need it, as follows:

   ```
   avctl stop_oc4j
   avctl start_oc4j
   ```

See Section 7.12 and Section 7.16 for more information about the avctl stop_oc4j and avctl start_oc4j commands.

Problem: The Agent OC4J or Audit Vault Console OC4J fails to start.

After you run the avctlstart_oc4j command, an avctl show_oc4j_status command shows that OC4J is not running. Or, after you issue avctl start_av command, an avctl show_av_status command shows that OC4J is not running.

Solution:

Go to $ORACLE_HOME/av/log/agent.err log file and check the error message that appears in the log.

Or, go to $ORACLE_HOME/oc4j/j2ee/home and issue the following command to find the error message that appears on the console:

```
java -jar oc4j.jar
```

This problem is most likely caused by a port conflict. For example, if the problem is caused by an RMI port conflict, you would see a message similar to the following:

```
C:\oracle\product\10.2\avagentrc3_01\oc4j\j2ee\home>java -jar oc4j.jar

08/05/16 10:39:51 Error starting ORMI-Server.  Unable to bind socket: Address
already in use: JVM_Bind
```

The RMI, JMS, and HTTP ports are necessary for starting OC4J or the Audit Vault Console OC4J. The agent OC4J and Audit Vault Console OC4J can fail to start or the agent service of the Audit Vault Console can become unavailable if these ports have a conflict. If there is a port conflict, you can modify the port settings in the following files at $ORACLE_HOME/oc4j/j2ee/config by selecting a port number not in use:

- rmi.xml
- jms.xml
- http-web-site.xml or (av-agent-web-site.xml)

Problem: The setup command returned an error message that the connection to the source database using the credentials in the wallet was not successful.

This problem is most likely due to entering an incorrect user name or password or both when issuing the `setup` command using the `AVORCLDB`, `AVMSSQLDB`, `AVSYBDB`, or `AVDB2DB` command-line utility.

Solution:

Reissue the `setup` command again using the correct credentials.

A.3.4 Troubleshooting the Audit Vault Collector

Problem: Cannot start the DBAUD collector and the log file shows an error.

The DBAUD collector log file (in the Audit Vault collection agent home directory) shows the following entry:

```
INFO @ '17/08/2008 15:05:48 02:00':
Could not call Listener, NS error 12541, 12560, 511, 2, 0
```

Solution:

Ensure that you completed the last step for configuring the source database and collectors, as described in Section 2.3.6, which describes how to run the `avorcldb setup` command in the Audit Vault collection agent home. See also Section 2.2.3 and Section 2.2.4.

Follow these steps:

1. Change directories to the `network/admin` directory:

   ```
   $ cd $ORACLE_HOME/network/admin
   ```

2. Perform the `cat` command on your `tnsnames.ora` file.

   There should be an entry similar to `SRCDB1`. If there is no `SRCDB1` entry in your `tnsnames.ora` file, then run the `avorcldb setup` command as shown in Section 2.3.6.

3. Try to connect to the source database with the following command, assuming your tnsnames.ora file has an `SRCDB2` entry.

   For example:

   ```
   $ sqlplus /@SRCDB1
   ```

   If the connection is successful, then your source database is set up correctly.

4. Try starting the DBAUD collector using the `avctl start_colletor` command.

   For example:

   ```
   $ avctl start_collector -collname REDO_Collector -srcname ORCLSRC1.EXAMPLE.COM
   ```

   See Section 7.11 for more information about the `avctl start_collector` command.

Problem: Not sure if the DBAUD and OSAUD collectors are collecting from the AUD$ table and the OS file, respectively.

After you set up both the DBAUD and OSAUD collectors, you want to check that they are collecting from the `AUD$` table and OS file, respectively.

Solution:

To determine if the DBAUD collector is collecting from the `AUD$` table, check the contents of the DBAUD log file, located in the `$ORACLE_HOME/av/log` directory.

To determine if the OSAUD collector is collecting from the OS file, check the contents of the `osaud_`*`collector-name_source-name_source-id`*`.log` file in the Audit Vault collection agent- home `$ORACLE_HOME/av/log` directory.

Check each file for entries that indicate that the collector is collecting audit records.

For example, the DBAUD collector log file would have entries similar to the following:

```
     ***** Started logging for 'AUD$ Audit Collector' *****
.
.
.
INFO @ '25/10/2008 19:08:42 -8:00':
     ***** SRC connected OK

INFO @ '25/10/2008 19:08:53 -8:00':
     ***** SRC data retireved OK
.
.
.
```

The OSAUD collector log file could have an entry as follows:

```
File opened for logging source "DBS1.REGRESS.RDBMS.DEV.US.ORACLE.COM"
INFO @ '24/10/2008 18:16:18 -8:00':
***** Started logging for 'OS Audit Collector' *****
```

If the log files look correct, then refresh the data warehouse using the `avctl refresh_warehouse` command in the Audit Vault Server shell. When this operation completes, log in to the Audit Vault Console as the Audit Vault auditor. Examine the graphical summary named **Activity by Audit Event Category** on the **Overview** page for the appearance of additional audit records in the various event categories. Increased counts for the various event categories indicate that these collectors are collecting audit records.

Problem: ORA-01017:invalid username/password; logon denied error when starting up the DBAUD_Collector or setting up the REDO_Collector.

When you try to start the DBAUD collector or configure the REDO collector, the following error message appears:

```
ORA-01017: invalid username/password; logon denied
```

Solution:

There may be a problem with your user name or password in the password file, or a problem with the wallet credentials. Try re-creating the user name and password. If the problem persists, re-create the password file. If this does not correct the problem, add the source user to the wallet again using the `avorcldb setup` command. Ensure that this user is the same user name and password that you are using on the source database.

Problem: Collector log for the MSSQLDB, SYBDB, or DB2DB collector indicates that a jar file is missing.

If the following JDBC driver jar files are missing from the Audit Vault collection agent `$ORACLE_HOME/jlib` library, this error appears in the collector log of the respective collector being used.

- **SQL Server:** `sqljdbc.jar`
- **Sybase ASE:** `jconn3.jar`
- **IBM DB2:** `db2jcc.jar`

Under other circumstances, such as when you use either the `AVMSSQLDB`, `AVSYBDB`, or `AVDB2DB` command-line utilities, the following error is appears when the JDBC driver is not in this directory:

```
JDBC Driver is missing. Please make sure that the JDBC jar exists in the location
specified in Audit Vault documentation.
```

Solution:

See the following sections:

- **SQL Server:** Section 2.4.1 for information about the `sqljdbc.jar` file
- **Sybase ASE:** Section 2.5.1 for information about the `jconn3.jar` file
- **IBM DB2:** Section 2.6.1 for information about the `db2jcc.jar` files

After you download and copy these JDBC drivers in place, restart OC4J. See Section 7.16 and Section 7.12 for more information about stopping and starting the OC4J agent.

Problem: Unable to connect to source database.

When you try to verify the ORCLDB, MSSQLDB, SYBDB, or DB2DB collector using the `verify` command, the following error message appears:

```
Unable to connect to source database
```

Solution:

This error appears if the source user that you specified in the `verify` command for the source database does not have sufficient privileges to connect to the source database. Check if the source user has sufficient privileges to connect to the respective database. See the following sections for information about creating a source user with sufficient privileges:

- Section 2.3.2 for Oracle databases
- Section 2.4.2 for Microsoft SQL Server databases
- Section 2.5.2 for Sybase ASE databases
- Section 2.6.2 for IBM DB2 databases

A.3.5 Troubleshooting Oracle Audit Vault Console

Problem: Audit Vault Console does not appear in the Web browser.

When you try to access the Audit Vault Console in a Web browser, it appears to hang, or after a while it times out.

Solution:

This may be happening because the Audit Vault Console is down. To check the status of the Audit Vault Console, issue an `avctl show_av_status` command in the Audit Vault Server shell. If the status indicates that the Audit Vault Console is down, issue the `avctl start_av` command in the Audit Vault Server shell to restart it. Then start the Audit Vault Console in the Web browser. The Audit Vault Console

should appear and let you log in to the management system of the Audit Vault auditor administrator.

Problem: Need to debug an Audit Vault Console problem.

You want to enable debug logging while trying to diagnose an Audit Vault Console problem.

Solution:

Run the following commands on the command-line:

```
avctl stop_av
avctl start_av -loglevel debug
```

Then check the log output in the Audit Vault Server `$ORACLE_HOME/av/log` directory.

Because debugging creates more logging and writing overhead, remember to disable it when you no longer need it, as follows:

```
avctl stop_av
avctl start_av
```

See Section 7.10 and Section 7.14 for more information about these commands.

A.3.6 Troubleshooting Oracle Audit Vault in an Oracle Real Application Clusters Environment

Problem: In an Oracle RAC environment, the avca drop_agent operation fails with an error when this command is issued from one of the Oracle RAC nodes.

When you try to run the `avca add_agent` command from one of the Oracle RAC nodes, the command fails.

Solution:

In an Oracle RAC environment, you must run the `AVCA` commands from the node on which Oracle Enterprise Manager resides. This is the same node on which the `av.ear` file is deployed.

To find where the `av.ear` file is deployed, locate the `$ORACLE_HOME/oc4j/j2ee/oc4j_applications/applications/av/av/WEB-INF/classes/av.properties` file is located.

Once you locate the node, run the `AVCA` and `AVCTL` commands from that node.

If the node on which the `av.ear` file is deployed is down, deploy the `av.ear` file to another node using the `avca deploy_av` command. See Section 6.4 for more information about this command.

When you run the `avca deploy_av` command, on the other node Oracle Database creates a wallet containing the default `avadmin` entries. You must add the other entries, such as the source user credentials, to the wallet by using the `setup` command for the appropriate utility (`AVORCLDB`, `AVMSSQLDB`, `AVSYBDB`, or `AVDB2DB`), depending on the collectors being used.

To access the Audit Vault Console from this other node, enter the following URL in the Web browser:

`http://host:port/av`

In this specification:

- *host* is the host name or IP address of the other Oracle RAC node
- *port* is the port number for the Oracle RAC node

B

Oracle Audit Vault Error Messages

The following sections describe the Oracle Audit Vault error messages:

- Audit Vault Server Error Messages
- Oracle Audit Vault Client Error Messages

B.1 Audit Vault Server Error Messages

This section describes the following Audit Vault Server-side error message codes:

- Generic Error Codes
- Source Database and Event Error Codes
- Collector Error Codes
- Attribute Definition Error Codes
- Alert Error Codes
- Server-Side Audit Service Error Messages
- Data Warehouse Error Messages
- Other Audit Vault Policy Error Messages

B.1.1 Generic Error Codes

This section describes the generic error codes.

46501, invalid %s

Cause: Invalid value specified.

Action: Provide a valid non-NULL value with a valid length.

46502, NULL in %s

Cause: NULL value specified.

Action: Provide a non-NULL value.

46503, object %s already exists

Cause: Object specified was already present in the system.

Action: Provide a different value. Remember that even if you drop an object from Oracle Audit Vault, such as a source database, the name of the dropped object is stored internally.

46504, duplicate %s

Cause: Value was repeated in the input.

Action: Remove the duplicates.

46505, object %s does not exist

Cause: Object specified was not present in the system.

Action: Provide a different value.

46612, invalid number of years %s for audit data retention; must be positive

Cause: Invalid number of years was specified for audit data retention.

Action: Specify a valid number, and ensure that this number is positive.

46626, Invalid number of years %s for audit data retention; must be positive

Cause: Invalid number of years was specified for audit data retention

Action: Specify valid number, the number should be positive.

46966, Function AV_TRUNCATE_CLOB does not exist in source database

Cause: The latest version of script `zarsspriv.sql` was not run. This can happen if you had configured the source database using a release earlier than the latest release of Oracle Audit Vault. The agent from the earlier Oracle Audit Vault release could contain a `zarsspriv.sql` script that is not compatible with the latest installed release of Oracle Audit Vault. You can find the `zarsspriv.sql` script in the `$ORACLE_HOME/av/scripts/streams/source` directory in the Oracle Audit Vault collection agent home directory.

Action: None. Function created automatically.

B.1.2 Source Database and Event Error Codes

This section describes the source database and event error codes.

46521, NULL value passed for a mandatory attribute

Cause: A mandatory attribute was set to a `NULL` value.

Action: Provide a non-`NULL` value for the mandatory attribute.

46522, mandatory attribute %s missing in the input

Cause: Mandatory attribute name was missing in the attribute value list.

Action: Provide the value for mandatory attribute.

46523, attempting to drop Event Category with active Events

Cause: Event category specified had active Events.

Action: Drop the active events before dropping this event category.

46524, active Collectors exist for the Source

Cause: Source database specified had collectors which were active.

Action: Drop active collectors for the given source database.

46525, Sourcetype-specific extension for Category already exists

Cause: Event category was specified which already has a format extension for the given source database type.

Action: Provide an event category that does not have a source database type-specific extension.

46526, attempting to drop an in-use Event mapping

Cause: Event mapping specified was in use.

Action: Provide an event mapping that is not being used.

46527, attempting to change an immutable attribute

Cause: An immutable attribute was specified.

Action: Provide a mutable attribute.

46528, attempting to drop system-defined Event

Cause: Event specified was system-defined.

Action: Provide a user-defined event.

46529, attempting to drop Event with active mappings

Cause: Event specified had active event mappings.

Action: Drop the active mappings before dropping this event.

46530, attempting to drop Sourcetype with active Sources

Cause: Source type specified had active source databases.

Action: Drop the active source databases before dropping this source type.

46531, unsupported Source version

Cause: Version specified for the source database was not supported.

Action: Provide a source database version that is equal to or greater than the minimum supported version for the corresponding source database type. See Section 1.3.1 for a listing of supported database versions.

B.1.3 Collector Error Codes

This section describes the collector error codes.

46506, attribute %s exists in %s

Cause: Attribute specified was already present.

Action: Provide a different attribute.

46507, invalid data or type name for attribute %s

Cause: Data type of the value specified was different from the type name of the attribute.

Action: Change the type name or the type of the value for the attribute.

46541, attempting to drop Collector Type with active Collectors

Cause: One or more collectors for this collector type were active.

Action: Drop all active collectors for this collector type.

46542, attempting to drop an Agent with active Collectors

Cause: One or more collectors for this agent were active.

Action: Drop all active collectors for this agent.

46543, attempting to drop a Collector before disabling the collection

Cause: The collection for the collector specified was not disabled.

Action: Disable the collection before dropping the collector.

46544, attempting to drop an Agent before disabling it

Cause: The agent specified was not disabled.

Action: Disable the agent before dropping it.

46964, Connector was not able to reconnect to Source Database

Cause: Maximum number of attempts to reconnect was exceeded.

Action: Verify connectivity and that the database is started. You can use the `lsnrctl status` command to check the status of the database.

B.1.4 Attribute Definition Error Codes

This section describes the attribute definition error codes.

46508, too many attributes of type %s specified

Cause: Specified number of attributes of this type exceeded the maximum number supported.

Action: Specify a fewer number of attributes of this type.

46551, attempting to change the type of an attribute currently in use

Cause: Attribute specified was in use.

Action: Provide an attribute that is not being used.

46552, attempting to drop an attribute currently in use

Cause: Attribute specified was in use.

Action: Provide an attribute that is not being used.

46553, attempting to change the type of an attribute without providing a new default value

Cause: Current type of the default value did not match with the new type specified.

Action: Provide a new default value for the attribute.

46965, Attribute %s is longer than 4000 bytes and was clipped

Cause: When the attribute was converted to UTF8 encoding, it became longer than 4000 bytes.

Action: None. It was clipped automatically after conversion.

B.1.5 Alert Error Codes

This section describes the alert error codes.

46561, no Format defined for the Source Type and Category

Cause: Format for the specified source type and category pair was not present in the system.

Action: Provide source type and category pair which already has a format defined.

46562, error in Alert condition

Cause: Invalid alert condition was specified.

Action: Correct the alert condition.

46563, attempting to drop a nonuser-defined Alert

Cause: Nonuser-defined alert was specified.

Action: Provide a user-defined alert.

46599, Internal error %s

Cause: Internal error occurred in Oracle Audit Vault.

Action: Contact Oracle Support Services.

B.1.6 Server-Side Audit Service Error Messages

This section describes the server-side audit service error codes.

46601, The authenticated user is not authorized with audit source

Cause: User is not authorized to send audit data on behalf of this audit source.

Action: Connect as the user who is associated with the source. Or grant this user appropriate authorization by changing the properties of the source database.

46602, Error on audit record insert as RADS partition full

Cause: RADS partition table is full.

Action: Purge the RADS partition table through archive.

46603, Error on audit record insert as RADS_INVALID table full

Cause: `RADS_INVALID` table is full.

Action: Need to purge `RADS_INVALID` table or make its size larger.

46604, Error on insert as Error table full

Cause: Error table is full.

Action: Need to purge the error table.

46605, There are more recovery entries than the maximum member can be returned

Cause: There are more recovery entries for this collector.

Action: Need to purge the old entries from the recovery table.

46606, There is no recovery entry for the given name

Cause: There was no recovery context matching to the given name.

Action: Need to check if the name was correct or if the recovery context was saved for this name.

46607, There are more configuration entries than the maximum member can be returned

Cause: There were more configuration entries for this collector.

Action: Need to reduce the configuration entries for this collector.

B.1.7 Data Warehouse Error Messages

This section describes messages from the data warehouse.

46620, invalid interval %s for data warehouse duration; must be positive

Cause: Invalid interval was specified for data warehouse duration.

Action: Specify valid interval, the interval should be positive.

46621, invalid start date %s for data warehouse operation; must be less than %s

Cause: Invalid start date was specified for data warehouse load/purge operation.

Action: Specify valid start date, the start date must be less than current date - warehouse duration.

46622, invalid number of days %s for data warehouse operation; must be greater than 0

Cause: Invalid number of days was specified for data warehouse load/purge operation.

Action: Specify valid number of days, the number of days must be positive

46623, cannot execute warehouse operation; another operation is currently running

Cause: A warehouse operation was executed while another operation is currently running.

Action: Wait for the operation to complete before reissuing the command.

46624, invalid schedule %s for data warehouse refresh schedule

Cause: Invalid schedule was specified for data warehouse refresh.

Action: Specify valid non-null schedule.

46625, invalid repeat interval %s for data warehouse refresh schedule

Cause: Invalid schedule was specified for data warehouse refresh.

Action: Specify valid non-null repeat interval.

B.1.8 Other Audit Vault Policy Error Messages

This section describes Oracle Audit Vault policy error messages.

46640, specified source name %s was not found

Cause: Invalid source name was specified.

Action: Specify a valid source name.

46641, archive does not exist

Cause: Invalid archive id was specified.

Action: Specify valid archive ID.

46642, database audit type invalid

Cause: Invalid database audit type specified.

Action: Database audit type must be S for standard or F for FGA.

46643, audit frequency invalid

Cause: Invalid audit frequency specified.

Action: Audit frequency must be A for "by access" or S for "by session".

46644, return type invalid

Cause: Return type was invalid.

Action: Return type must be S for "success", F for "failure", or B for "both".

46645, privilege flag invalid

Cause: Privilege flag is invalid.

Action: The privilege flag must be Y or N.

46646, specified Agent name %s was not found

Cause: Invalid Agent name was specified.

Action: Specify a valid Agent name.

B.2 Oracle Audit Vault Client Error Messages

This section describes the following Oracle Audit Vault client error messages:

- General Error Messages
- CSDK Error Messages
- OSAUD Collector Error Messages
- DBAUD Collector Error Messages

B.2.1 General Error Messages

This section describes the general error messages.

46800, Normal, successful completion

Cause: Normal exit.

Action: None.

46801, Out of memory

Cause: The process ran out of memory.

Action: Increase the amount of memory on the system.

B.2.2 CSDK Error Messages

This section describes the CSDK error messages.

46821, generic CSDK error (line %d)

Cause: There was a generic error in CSDK.

Action: Contact Oracle Support Services.

46822, no collector details for collector %s

Cause: Collector is not properly set up in AV tables.

Action: Configure collector.

46823, attribute %s is not valid for category

Cause: Collector attempted to set invalid attribute.

Action: Contact collector owner.

46824, type is not valid for attribute %s

Cause: Collector attempted to set value of wrong type to attribute.

Action: Contact collector owner.

46825, invalid record

Cause: Collector attempted to pass invalid record.

Action: Contact collector owner.

46826, invalid parameter %s (line %d)

Cause: Collector attempted to pass invalid parameter.

Action: Contact collector owner.

46827, invalid context

Cause: Collector attempted to pass invalid context.

Action: Contact collector owner.

46828, OCI layer error %d

Cause: OCI layer returned error.

Action: Contact collector owner.

46829, category %s unknown

Cause: Collector attempted to pass category not configured in AV.

Action: Contact collector owner.

46830, null pointer (line %d)

Cause: Collector attempted to pass null pointer.

Action: Contact collector owner.

46831, invalid source event id (%s)

Cause: Collector passed source event id not suitable for category.

Action: Contact collector owner.

46832, internal error (line %d)

Cause: Internal error occurred in CSDK.

Action: Contact Oracle Support Services.

46833, invalid error record

Cause: Collector attempted to pass invalid error record.

Action: Contact collector owner.

46834, missing attribute in error record

Cause: One or more attributes of error record is missing.

Action: Contact collector owner.

46835, duplicate error attribute

Cause: Collector attempted to set already set attribute.

Action: Contact collector owner.

46836, error record in use

Cause: Attempt to create a new error record before sending or dropping the previous one.

Action: Contact collector owner.

46837, missing eventid attribute in audit record

Cause: Event ID attributes of audit record are missing.

Action: Contact collector owner.

46838, Internal Error: Failed to insert %s into %s hash table

Cause: Core hash table insertion function failed.

Action: Contact collector owner.

B.2.3 OSAUD Collector Error Messages

This section describes the OSAUD collector error messages.

46901, internal error, %s

Cause: There was a generic internal exception for OS Audit Collector.

Action: Contact Oracle Support Services.

46902, process could not be started, incorrect arguments

Cause: Wrong number of arguments or invalid syntax used.

Action: Please verify that all the required arguments are provided. The required arguments are Host name, Source name, Collector name, and the Command.

46903, process could not be started, operating system error

Cause: The process could not be spawned because of an operating system error.

Action: Please consult the log file for detailed operating system error.

46904, collector %s already running for source %s

Cause: Collector specified was already running.

Action: Provide a different collector or source name.

46905, collector %s for source %s does not exist

Cause: Collector specified was not running.

Action: Provide a different collector or source name.

46906, could not start collector %s for source %s, reached maximum limit

Cause: No more collectors could be started for the given source.

Action: None.

46907, could not start collector %s for source %s, configuration error

Cause: Some collector parameters were not configured correctly.

Action: Check the configuration parameters added during ADD_COLLECTOR.

46908, could not start collector %s for source %s, directory access error for %s

Cause: Access to specified directory was denied.

Action: Verify the path is correct and the collector has read permissions on the specified directory.

46909, could not start collector %s for source %s, internal error: [%s], [%d]

Cause: An internal error occurred while starting the collector.

Action: Contact Oracle Support Services.

46910, error processing collector %s for source %s, directory access error for %s

Cause: Access to specified directory was denied.

Action: Verify the path is correct and the collector has read permissions on the specified directory.

46911, error processing collector %s for source %s, internal error: [%s], [%d]

Cause: An internal error occurred while processing the collector.

Action: Contact Oracle Support Services.

46912, could not stop collector %s for source %s

Cause: An error occurred while closing the collector.

Action: None.

46913, error in recovery of collector %s for source %s: %s

Cause: An error occurred while accessing the file.

Action: Verify the path is correct and the collector has read permissions on the specified directory.

46914, error in recovery of collector %s for source %s, internal error: [%s], [%d]

Cause: An internal error occurred while getting recovery information for collector.

Action: Contact Oracle Support Services.

46915, error in parsing of collector %s for source %s: %s

Cause: An error occurred while accessing the file.

Action: Verify the path is correct and the collector has read permissions on the specified directory.

46916, error in parsing of collector %s for source %s, internal error [%s], [%d]

Cause: An internal error occurred while parsing data for collector.

Action: Contact Oracle Support Services.

46917, error processing request, collector not running

Cause: OS Audit Collector was not running and a command was issued.

Action: Start the collector using command START.

46918, could not process the command; invalid command

Cause: An invalid value was passed to the command argument.

Action: Please verify that a valid value is passed to command argument. The valid values are `START`, `STOP` and `METRIC`.

46919, error processing METRIC command; command is not in the required format

Cause: METRIC command was not in the required METRIC:XYZ format.

Action: Please verify that the metric passed is in METRIC:XYZ format where XYZ is the type of metric (Example: METRIC:ISALIVE).

46920, could not start collector %s for source %s, directory or file name %s is too long

Cause: The name of directory or file was too long.

Action: Verify the length of the path is less than the system-allowed limit.

46921, error processing collector %s for source %s, directory or file name %s is too long

Cause: The name of directory or file was too long.

Action: Verify the length of the path is less than the system-allowed limit.

46922, could not start collector %s for source %s, cannot open Windows event log

Cause: Windows event log could not be opened.

Action: Verify event log exists.

46923, OCI error encountered for source database %s access, audit trail cleanup support disabled.

Cause: An error was encountered while attempting to connect to or execute SQL statements on the source database.

Action: Verify source database and listener are up and connect information is correct.

46924, Corrupted recovery information detected for collector %s for source %s

Cause: Corrupted recovery information detected.

Action: Contact Oracle Support Services.

46925, error in parsing XML file %s for collector %s and source database %s : error code %u.

Cause: An internal error occurred while parsing data for collector.

Action: Verify that collector has read permissions on the file and the file is in proper XML format. Contact Oracle Support Services for patch set.

46926, error in recovery of XML file %s for collector %s and source database %s : error code %u.

Cause: An internal error has occurred while parsing data for collector.

Action: Verify that collector has read permissions on the file and the file is in proper XML format. Contact Oracle Support Services for patch set.

46927, Syslog is not configured or error in getting audit files path for syslog for collector %s and source database %s.

Cause: One of the following occurred:

- `facility.priority` was not valid.
- There was no corresponding path for `facility.priority` setting.
- Source database was only returning facility and there was no corresponding path for `facility.*` setting.

Action: Configure syslog auditing to a valid `facility.priority` setting and corresponding valid path. If source database only returning the facility, then contact Oracle Support Services for patch set.

46928, Collector %s for source %s cannot read complete file %s

Cause: File size is more than 2GB.

Action: File size should be less than 2GB. Please use log rotation to limit the file size to less then 2GB.

B.2.4 DBAUD Collector Error Messages

This section describes the DBAUD collector error messages.

46941, internal error, on line %d in file ZAAC.C, additional information %d

Cause: There was a generic internal exception for `AUD$` Audit Collector.

Action: Contact Oracle Support Services.

46942, invalid AUD Collector context

Cause: The AUD Collector context passed to collector was invalid.

Action: Make sure that context passed is the context returned by `ZAAC_START`.

46943, NULL AUD Collector context

Cause: The pointer to AUD Collector context passed to Collector was `NULL`.

Action: Make sure that context passed is the context returned by `ZAAC_START`.

46944, conversion error in column %s for <%s>

Cause: The `VARCHAR` retrieved from `AUD$` or `FGA_LOG$` table could not be converted to `ub4`.

Action: Correct value in source database.

46945, bad recovery record

Cause: The recovery record retrieved from Audit Vault was damaged.

Action: None. The record will be corrected automatically.

46946, too many active sessions

Cause: The number of active sessions exceeded the specified number in the `GV$PARAMETER` table.

Action: Contact Oracle Support Services.

46947, CSDK layer error

Cause: CSDK layer returned error indication.

Action: Action should be specified in CSDK error report.

46948, already stopped

Cause: AUD collector already stopped because of previous fatal error.

Action: Restart Collector.

46949, log level

Cause: Specified log level was invalid.

Action: Use a legal log level (1,2,3).

46950, log file

Cause: An error occurred during the opening of the log file.

Action: Make sure that the log directory exists, and that the directory and log file are writable.

46951, bad value for AUD collector attribute

Cause: Specified collector attribute was invalid.

Action: Correct the attribute value in the Audit Vault table `AV$ATTRVALUE`.

46952, bad name for AUD collector metric

Cause: The specified metric name was undefined.

Action: Use a correct metric name.

46953, unsupported version

Cause: The specified version of the source database is not supported.

Action: Update to supported version.

46954, recovery context of 10.x

Cause: Source database (9.x) was incompatible with 10.x recovery context.

Action: Clean up `AUD$` and `FGA_LOG$` tables and recovery context.

46955, recovery context of 9.x

Cause: Source database (10.x) was incompatible with 9.x recovery context.

Action: Clean up `AUD$` and `FGA_LOG$` tables and recovery context.

46956, FGA_LOG$ table of 9.x

Cause: Source database (10.x) was incompatible with 9.x rows of `FGA_LOG$`.

Action: Clean up `FGA_LOG$` table.

46957, RAC recovery context

Cause: Non-RAC source database was incompatible with RAC recovery context.

Action: Clean up AUD$ and FGA_LOG$ tables and recovery context.

46958, Non-RAC recovery context

Cause: RAC source database was incompatible with non-RAC recovery context

Action: Clean up AUD$ and FGA_LOG$ tables and recovery context.

46959, bad authentication information

Cause: Incorrect format of authentication information in the column COMMENT$TEXT.

Action: Contact Oracle Support Services.

46960, bad metric request

Cause: Unknown metric name (%s) was provided in metric request.

Action: Contact Oracle Support Services.

46961, internal error, on line %d in file ZAAC.C, additional info |%s|

Cause: There was a generic internal exception for AUD$ Audit Collector.

Action: Contact Oracle Support Services.

46962, Database Vault audit table is not accessible

Cause: Database Vault was not set up properly or proper role was not granted to user used by collector.

Action: Set up Database Vault and make sure that DVSYS.AUDIT_TRAIL$ is accessible to the user used by collector.

46963, Some rows may have been missed by Audit Vault or may be duplicated

Cause: Collector encountered rows in the SYS.AUD$ or FGA_LOG$ tables with SESSIONID <= 0.

Action: Contact Oracle Support Services.

Glossary

alert

An indicator signifying that a particular metric condition has been encountered. The following conditions trigger alerts:

- A metric threshold is reached.
- The availability of a monitored service changes. For example, the availability of the host changes from up to down.
- A metric-specific condition occurs. For example, an error message is written to a database alert log file.

alert rule

A rule in an audit policy setting that specifies an audit condition or other abnormal condition that raises an alert. An alert rule is based on the data in a single audit record.

audit data source

See **source database**.

audit data warehouse

A data store within Oracle Audit Vault that stores processed audit data from the **raw audit data store**. Auditors can access this data by generating the Oracle Audit Vault reports.

See also **data warehouse**.

audit rule

A rule in a audit setting that specifies the action to be audited (for example, a logon attempt or a user accessing a table).

audit setting

A set of rules that specifies which audit events should be collected in Oracle Audit Vault, and how each audit event should be evaluated after it is inserted into the **raw audit data store**. The types of rules in an audit setting include alert rules, audit rules, and capture rules. An audit setting can be composed of two or more sets of rules known as a *composite audit setting*.

See also **alert rule**; **audit rule**; and **capture rule**.

Audit Vault administrator user

A user granted the `AV_ADMIN` role, and is the audience for this manual. This user configures and manages collectors, collection agents, and warehouse settings and

scheduling. This user also configures sources, enables and disables systemwide alerts, views audit event categories, and monitors audit errors.

Audit Vault agent user

A user account granted the AV_AGENT role. This is an internal user only.

Audit Vault auditor user

A user granted the AV_AUDITOR role. This user monitors audit event categories for alert activity to detect security risks, creates detail and summary reports of events across systems, and manages the reports. This user also manages audit policies that create alerts and evaluate alert scenarios, and manage audit settings. This user can use the data warehouse services to further review the audit data and look for trends, intrusions, anomalies, and other items of interest. See *Oracle Audit Vault Auditor's Guide* for more information about the auditor's duties.

Audit Vault Configuration Assistant (AVCA)

See **AVCA**.

Audit Vault Control (AVCTL)

See **AVCTL**.

Audit Vault IBM DB2 Database (AVDB2DB)

See **AVDB2DB**.

Audit Vault Microsoft SQL Server Database (AVMSSQLDB)

See **AVMSSQLDB**.

Audit Vault Oracle Database (AVORCLDB)

See **AVORCLDB**.

Audit Vault Sybase ASE Database (AVSYBDB)

See **AVSYBDB**.

AVCA

Audit Vault Configuration Assistant, a command-line utility that you use to manage various Oracle Audit Vault components, manage collection agents (adding, altering, or dropping), secure communication between the Audit Vault Server and Audit Vault collection agent, set warehouse scheduling and audit data retention settings, and create a wallet and certificates for the collection agent, as needed. See Chapter 6, "Audit Vault Configuration Assistant (AVCA) Reference," for more information.

AVCTL

Audit Vault Control, a command-line utility that you use to manage the Oracle Audit Vault components, such as starting and stopping collection agents, collectors, the Audit Vault Console, and OC4J. See Chapter 7, "Audit Vault Control (AVCTL) Reference," for more information.

AVDB2DB

Audit Vault IBM DB2 Database, a command-line utility that you use to configure Oracle Audit Vault to retrieve audit data from an IBM DB2 database. The process entails adding the source database and configuring the **DB2DB collector**. See Chapter 11, "Audit Vault IBM DB2 (AVDB2DB) Utility Commands," for more information.

AVMSSQLDB

Audit Vault Microsoft SQL Server Database, a command-line utility that you use to configure Oracle Audit Vault to retrieve audit data from a SQL Server database The configuration process entails adding the source database and configuring the **MSSQLDB collector**. See Chapter 9, "Audit Vault Microsoft SQL Server (AVMSSQLDB) Utility Commands," for more information.

AVORCLDB

Audit Vault Oracle Database, a command-line utility that you use to configure Oracle Audit Vault to retrieve audit data from an Oracle database. The configuration process entails adding the source database and configuring the appropriate collector (**DBAUD collector**, **OSAUD collector**, or **REDO collector**). See Chapter 8, "Audit Vault Oracle Database (AVORCLDB) Utility Commands," for more information.

AVSYBDB

Oracle Audit Vault Sybase ASE Database, a command-line utility that you use to configure Oracle Audit Vault to retrieve audit data from a Sybase ASE database. The configuration process entails adding the source database and configuring the **SYBDB collector**. See Chapter 10, "Audit Vault Sybase ASE (AVSYBDB) Utility Commands," for more information.

capture rule

A rule in an audit policy setting that specifies an audit event that is sent to Oracle Audit Vault.

certificate

A digitally signed statement by a certificate authority (CA), saying that it has certified the identity of an entity in some way. Upon request, the CA verifies the identity of the entity, and signs and grants a certificate, with a private key. This indicates that the certificate has been checked for data integrity and authenticity, where integrity means that data has not been modified or tampered with, and authenticity means that data comes from the entity claiming to have created and signed it.

A certificate is a digital identification of an entity that contains the following:

- SSL public key of the server
- Information about the server
- Expiration date
- Digital signature by the issuer of the certificate, used to verify the authenticity of the certificate

collection agent

A process in which **collector**s run. A collection agent defines the connection between the collector and the audit service, and interacts with the management service to manage and monitor collectors. See Section 1.3.3 for detailed information about collection agents.

collector

A component that collects audit data for a source and sends the audit records to Audit Vault. Each of the supported source database products has one or more associated collectors. See Table 1–4 on page 1-6 for detailed information about the available collectors.

See also **DB2DB collector**, **DBAUD collector**, **MSSQLDB collector**, **OSAUD collector**; **REDO collector**; and **SYBDB collector**.

composite audit setting

See **audit setting**.

configuration data

The Oracle Audit Vault metadata (stored within Oracle Audit Vault) that describes how to process and control the audit data as it passes through the Oracle Audit Vault system.

data warehouse

A relational database that is designed for query and analysis rather than transaction processing. A data warehouse usually contains historical data that is derived from transaction data, but it can include data from other sources. It separates the analysis workload from the transaction workload and enables a business to consolidate data from several sources. In Oracle Audit Vault, the data warehouse stores audit data that has been inserted into the data warehouse tables. From there, an Oracle Audit Vault auditor can see this data by generating the Oracle Audit Vault reports. See *Oracle Audit Vault Auditor's Guide* for more information.

See also **audit data warehouse** and **raw audit data store**.

DB2DB collector

IBM DB2 audit log collector. This collector extracts and collects IBM DB2 (releases 8 and 9.5) audit records from the audit trail logged in the ASCII text files generated by the source database. The DB2DB collector belongs to the DB2DB collector type.

DBAUD collector

Oracle Database DB audit log collector. This collector collects audit data from the Oracle Database `SYS.AUD$` table and the Oracle Database Vault audit trail `DVSYS.AUDIT_TRAIL$` table. The DBAUD collector belongs to the ORCLDB_DBAUD collector type.

digital certificate

See **certificate**.

fact table

A table in a **star schema** that contains facts. A fact table typically has two types of columns: columns that contain facts and columns that are foreign keys to dimension tables. The primary key of a fact table is usually a composite key composed of all of its foreign keys.

A fact table might contain either detail level facts or facts that have been aggregated (fact tables that contain aggregated facts are often called summary tables). A fact table usually contains facts with the same level of aggregation.

In Oracle Audit Vault, the **audit data warehouse** tables are in a star schema.

HTTPS

Hypertext Transmission Protocol, Secure. The use of Secure Sockets Layer (SSL) as a sublayer under the regular HTTP application layer.

Hypertext Transmission Protocol, Secure

See **HTTPS**.

keystore

A repository that includes the following:

- Certificates identifying trusted entities. When a keystore contains only certificates of trusted entities, it can be called a *trust store*.
- Private key and the matching certificate. This certificate is sent as a response to SSL authentication challenges.

keytool

A key and certificate management utility that Oracle Audit Vault uses to generate the keystore. It enables users to self-authenticate by administering their own public and private key pairs and associated certificates or data integrity and authentication services, using digital signatures. The `keytool` utility is located at `$ORACLE_HOME/jdk/bin`.

For Oracle Audit Vault, you must run the `keytool` utility to generate a keystore file if you want to configure HTTPS communication for Audit Vault. See Section 5.5 for more information.

LCR

Logical change record. This is a message with a specific format that describes a database change.

logical change record (LCR)

See **LCR**.

mapping

The definition of the relationship and data flow between source database and target objects.

metric

Unit of measurement used to report the health of the system.

MSSQLDB collector

Microsoft SQL Server Database audit log collector. This collector extracts and collects Microsoft SQL Server Database (SQL Server 2000 and SQL Server 2005) (for Windows platforms) audit records from the Windows Event logs, Server-side Traces, and C2 auditing logs. The MSSQLDB collector belongs to the MSSQLDB collector type.

Oracle Database DB audit logs collector (DBAUD)

See **DBAUD collector**.

Oracle Database OS audit logs collector (OSAUD)

See **OSAUD collector**.

Oracle Database redo logs collector (REDO)

See **REDO collector**.

OSAUD collector

Oracle Database OS audit log collector. This collector parses operating system (OS) log file entries into audit records. The OSAUD collector belongs to the ORCLDB_OSAUD collector type.

On Microsoft Windows, the OS audit trail depends on the `AUDIT_TRAIL` parameter setting:

- If the setting is `OS`, the OS audit trail is the Windows event log.
- If the setting is XML, then the OS audit trail is the XML file.

The OSAUD collector automatically extracts and collects audit records from either audit trail.

PKI

Public key infrastructure. This information security technology uses the principles of public key cryptography to encrypt and decrypt information using a shared public and private key pair. It provides for secure, private communications within a private network.

public key infrastructure (PKI)

See **PKI**.

raw audit data store

The first location in which Oracle Audit Vault places audit data it collects from a source database. It stores this unprocessed audit data in partitioned tables based on timestamp, and in unpartitioned tables based on source ID. Oracle Audit Vault then sends this data to the **data warehouse**, where it is organized into tables. Auditors access this data by generating audit reports.

REDO collector

Oracle Database redo log collector. This collector translates logical change records (LCRs) into audit records. The REDO collector belongs to the ORCLDB_REDO collector type.

source database

A database instance that has been configured to send audit data to Oracle Audit Vault.

The audit data source consists of databases, applications, or systems that generate audit data. For the current release of Oracle Audit Vault, the following database products are audit data sources:

- Oracle Database
- Microsoft SQL Server
- Sybase ASE
- IBM DB2

These databases can run on the same or different computers, potentially resulting in multiple source databases on the same system. Audit data from audit sources represent a variety of audit formats. Source types represent a class of audit sources. For example, Oracle Database audit sources with the same audit formats, audit events, and collection mechanisms represent an audit source type. Table 1–4 on page 1-6 lists the collectors that are associated with these database products.

See also **DB2DB collector**, **DBAUD collector**, **MSSQLDB collector**, **OSAUD collector**; **REDO collector**; and **SYBDB collector**.

star schema

A relational schema whose design represents a multidimensional data model. The star schema consists of one or more **fact table**s and one or more dimension tables that are related through foreign keys.

SYBDB collector

Sybase ASE Database audit log collector. This collector extracts and collects Sybase ASE (ASE 12.5.4 and ASE 15.0.2) audit records from the audit trail logged in audit tables in the `sybsecurity` database. The SYBDB collector belongs to the SYBDB collector type.

trust store

See keystore.

X.509

A widely used standard for defining digital certificates. X.509 defines a standard certificate format for public key certificates and certificate validation.

Index

A

add_agent command (AVCA utility), 6-2
add_collector command
  IBM DB2 databases, 11-2
  Oracle databases, 8-2
  SQL Server databases, 9-2
  Sybase ASE databases, 10-2
add_source command
  IBM DB2 databases, 11-3
  Oracle databases, 8-4
  SQL Server databases, 9-3
  Sybase ASE databases, 10-3
adding
  Audit Vault collection agents, 6-2
  collectors for IBM DB2 databases, 11-2
  collectors for Oracle databases, 8-2
  collectors for SQL Server databases, 9-2
  collectors for Sybase ASE databases, 10-2
administrators
  general steps for using Oracle Audit Vault, 1-2
  main tasks, 1-1
  managing security, 5-1 to 5-10
alert settings
  disabling globally, 3-3
  enabling globally, 3-3
  error messages, B-4
alter_collector command
  IBM DB2 databases, 11-3
  Oracle databases, 8-5
  SQL Server databases, 9-3
  Sybase ASE databases, 10-3
alter_source command
  IBM DB2 databases, 11-4
  Oracle databases, 8-8
  SQL Server databases, 9-5
  Sybase ASE databases, 10-4
altering
  DB2DB collector attributes, 11-3
  DBAUD collector attributes, 8-5
  MSSQLDB collector attributes, 9-3
  OSAUD collector attributes, 8-5
  REDO collector attributes REDO collector altering, 8-5
  source database collector attributes, 3-6 to 3-7
  SYBDB collector attributes, 10-3
archive log disk space
  monitoring in Audit Vault Server, 4-2
attributes
  collectors
    about, 3-6
    altering Audit Vault Console, 3-6
    altering in shell, 3-7
  source databases
    about, 3-15
    altering in Audit Vault Console, 3-15
    altering in shell, 3-16
audit data
  loading to Audit Vault, 3-12
  loading to warehouse, 7-3
  purging from warehouse, 7-4
  refreshing data repository, 7-5
  scheduling refreshes, 6-13
  setting a retention period, 6-12
  See also purging audit data
audit events
  error codes, B-2
  viewing categories, 3-3
Audit Vault administrator roles
  AV_ADMIN
    about, 1-10
    roles and privileges granted, 5-3
  AV_AGENT
    about, 1-10
    roles and privileges granted, 5-3
  AV_AUDITOR
    about, 1-10
    roles and privileges granted, 5-3
  See also Oracle Database Vault, administrator roles
Audit Vault collection agents
  about, 1-5
  adding, 6-2
  checking status, 2-25
  checking status of collection agent, 7-6
  checking status of OC4J agent, 7-8
  configuring connectivity for Oracle RAC nodes, 4-4
  configuring for Oracle RAC nodes, 4-3
  debugging advice, A-6
  dropping, 6-5
  errors log file
    Audit Vault collection agent, A-3

Audit Vault Server, A-1
log file
Audit Vault collection agent, A-3
Audit Vault Server, A-1
redeploying av.ear or AVAgent.ear file, 6-9
running command in Windows environment, 2-4
securing, 6-10
starting, 7-9
status blank on Windows Services Panel, A-5
stopping, 7-12
troubleshooting tips, A-5 to A-7
UNIX environment variable settings, 2-4
using on Windows, 2-4
wallet credentials not successful, A-7
Audit Vault collectors
about, 1-5
adding for DB2, 11-2
adding for Oracle database, 8-2
adding for SQL Server, 9-2
adding to Sybase ASE, 10-2
adding, general steps, 2-1
attributes
about, 3-6
altering in Audit Vault Console, 3-6
altering in shell, 3-7
DB2DB collector, 11-3
DBAUD collector, 8-5, 8-6
MSSQLDB collector, 9-3
OSAUD collector, 8-5, 8-6
REDO collector, 8-7
SYBDB collector, 10-3
checking status of, 2-27, 7-7
collector not starting, 7-10
dropping
from IBM DB2, 11-5
from Oracle databases, 8-9
from SQL Server databases, 9-6
from Sybase ASE databases, 10-5
starting, 7-10
stopping, 7-13
troubleshooting tips, A-7 to A-9
Audit Vault Configuration Assistant (AVCA) utility
commands
add_agent command, 6-2
create_credential command, 6-2
create_wallet command, 6-3
deploy_av command, 6-4
drop_agent command, 6-5
generate_csr command, 5-9, 6-6
-help command, 6-6
import_cert command, 5-9, 6-8
redeploy command, 6-9
remove_cert command, 6-9
secure_agent command, 6-10
secure_av command, 6-11
set_warehouse_retention command, 6-12
set_warehouse_schedule command, 6-13
table of, 6-1
log file
Audit Vault collection agent, A-3
Audit Vault Server, A-1
Audit Vault Console
checking status, 3-2
checking status of, 7-7
debugging advice, A-10
logging in, 3-2
starting, 3-2
stopping, 3-3
troubleshooting tips, A-9
viewing
Audit Vault errors, 3-5
Web browser hanging, A-9
Audit Vault Control (AVCTL) utility
commands
-help command, 7-2
load_warehouse command, 7-3
purge_warehouse command, 7-4
refresh_warehouse command, 7-5
set_warehouse command, 6-13
show_agent_status command, 7-6
show_av_status command, 7-7
show_collector_status command, 2-27, 7-7
show_oc4j_status command, 7-8
start_agent command, 2-25, 7-9
start_av command, 3-2, 7-9
start_collector command, 2-26, 7-10
start_oc4j command, 7-11
stop_agent command, 7-12
stop_av command, 3-3, 7-12
stop_collector command, 7-13
stop_oc4j command, 7-14
table of, 7-1
log file
Audit Vault collection agent, A-3
Audit Vault Server, A-1
Audit Vault data warehouse
about, 3-7
error messages, B-5
loading data
about, 3-12
using Audit Vault Console, 3-12
using shell, 3-13
manually refreshing
about, 3-11
using Audit Vault Console, 3-11
using shell, 3-12
purging data
about, 3-14
using Audit Vault Console, 3-14
using shell, 3-14
scheduling audit data refreshes
creating schedule, 3-9 to 3-11
example, 3-8
scheduling audit refreshes
how it works, 3-8
setting data retention period, 6-12
Audit Vault IBM DB2 (AVDB2DB) utility
commands
add_collector command, 11-2
add_source command, 11-3

alter_collector command, 11-3
alter_source command, 11-4
drop_collector command, 11-5
drop_source command, 11-6
-help command, 11-7
setup command, 11-7
table of, 11-1
verify command, 11-8
log file
Audit Vault collection agent, A-3
Audit Vault Server, A-2
syntax, 11-1
See also IBM DB2 databases
Audit Vault Microsoft SQL Server Database (AVMSSQLDB) utility
commands
add_collector command, 9-2
add_source command, 9-3
alter_collector command, 9-3
alter_source command, 9-5
drop_collector command, 9-6
drop_source command, 9-6
-help command, 9-7
setup command, 9-8
table of, 9-1
verify command, 9-9
log file
Audit Vault collection agent, A-4
Audit Vault Server, A-2
syntax, 9-1
See also Microsoft SQL Server databases
Audit Vault Oracle Database (AVORCLDB) utility
commands
add_collector command, 8-2
add_source command, 8-4
alter_collector command, 8-5
alter_source command, 8-8
drop_collector command, 8-9
drop_source command, 8-9
-help command, 8-10
setup command, 8-11
table of, 8-1
verify command, 8-12
log file, A-4
Audit Vault collection agent, A-3
Audit Vault Server, A-1
syntax, 8-1
See also Oracle databases
Audit Vault Server
about, 1-4
administrative tasks
archive log disk space, 4-2
backup and recovery operations, 4-2
changing user passwords, 5-4
configuring collection agent connectivity for RAC, 4-4
configuring collection agent to listen to RAC nodes, 4-3
flash recovery area, 4-2
SYSAUX tablespace usage, 4-1
alert settings, managing, 3-3
audit event categories, viewing, 3-3
Audit Vault Console status, checking, 3-2
checking errors, 3-5
components, 1-4
error codes
alert errors, B-4
attribute definition errors, B-4
collector errrors, B-3
data warehouse errors, B-5
event errors, B-2
generic errors, B-1
policy errors, B-6
service-side audit service errors, B-5
performance tuning, A-5
securing, 5-9, 6-11
starting console, 7-9
troubleshooting tips, A-5
UNIX environment variable settings, 2-2
Audit Vault Sybase ASE Database (AVSYBDB) utility
commands
add_collector command, 10-2
add_source command, 10-3
alter_collector command, 10-3
alter_source command, 10-4
drop_collector command, 10-5
drop_source command, 10-6
-help command, 10-7
setup command, 10-7
table of, 10-1
verify command, 10-8
log file
Audit Vault collection agent, A-4
Audit Vault Server, A-2
syntax, 10-1
See also Sybase ASE databases
audited records, purging
See purging audit records
authentication
configuring HTTPS communication, 5-9
securing Audit Vault, 5-9, 6-11
securing Audit Vault collection agent, 5-9, 6-10
AV_ADMIN role
about, 1-10
roles and privileges granted, 5-3
AV_AGENT role
about, 1-10
roles and privileges granted, 5-3
AV_AUDITOR role
about, 1-10
roles and privileges granted, 5-3
AVDB2DB command-line utility
See Audit Vault IBM DB2 (AVDB2DB) utility
AVMSSQLDB command-line utility
See Audit Vault Microsoft SQL Server Database (AVMSSQLDB) utility
AVORCLDB command-line utility, 8-1
See Audit Vault Oracle Database (AVORCLDB) utility
AVSYBDB command-line utility

See Audit Vault Sybase ASE Database (AVSYBDB) utility

B

back-up and recovery for Audit Vault Server, 4-2

C

certificates
See Oracle wallets
CLEAN_AUDIT_TRAIL procedure, 14-4
CLEAR_AUDIT_TRAIL_PROPERTY procedure, 14-5
CLEAR_LAST_ARCHIVE_TIMESTAMP procedure, 14-6
collection agents
See Audit Vault collection agents
collectors
See Audit Vault collectors
commands
log file, A-4
coraenv UNIX script, 2-2
create_credential command (AVCA utility), 6-2
CREATE_PURGE_JOB procedure, 14-7
create_wallet command, 6-3

D

data dictionary views
DBMS_AUDIT_MGMT views, 13-1 to 13-3
data warehouse
See Audit Vault data warehouse
DB2DB collector
about, 1-6
DB2DB collector attributes, 11-3
db2jcc.jar file, 2-19
DBA_AUDIT_MGMT_CLEAN_EVENTS data dictionary view, 13-3
DBA_AUDIT_MGMT_CLEANUP_JOBS data dictionary view, 13-2
DBA_AUDIT_MGMT_CONFIG_PARAMS data dictionary view, 13-1
DBA_AUDIT_MGMT_LAST_ARCH_TS data dictionary view, 13-2
DBAUD collector
about, 1-6
attributes, 8-6
DBAUD collector attributes, 8-5
DBMS_AUDIT_MGMT package
about, 14-1
CLEAN_AUDIT_TRAIL procedure, 14-4
CLEAR_AUDIT_TRAIL_PROPERTY procedure, 14-5
CLEAR_LAST_ARCHIVE_TIMESTAMP procedure, 14-6
CREATE_PURGE_JOB procedure, 14-7
data dictionary views, 13-1
DEINIT_CLEANUP procedure, 14-8
DROP_PURGE_JOB procedure, 14-9
finding information about
See data dictionary view names beginning with DBA_AUDIT_MGMT_
GET_AUDIT_COMMIT_DELAY function, 14-9
INIT_CLEANUP procedure, 14-10
IS_CLEANUP_INITIALIZED function, 14-11
SET_AUDIT_TRAIL_LOCATION procedure, 14-11
SET_AUDIT_TRAIL_PROPERTY procedure, 14-12
SET_DEBUG_LEVEL procedure, 14-14
SET_LAST_ARCHIVE_TIMESTAMP procedure, 14-15
SET_PURGE_JOB_INTERVAL procedure, 14-16
SET_PURGE_JOB_STATUS procedure, 14-17
DBSNMP account
how Oracle Audit Vault handles, 5-3
DEINIT_CLEANUP procedure, 14-8
deploy_av command (AVCA utility), 6-4
drop_agent command (AVCA utility), 6-5
drop_collector command
IBM DB2 databases, 11-5
Oracle databases, 8-9
SQL Server databases, 9-6
Sybase ASE databases, 10-5
DROP_PURGE_JOB procedure, 14-9
drop_source command
IBM DB2 databases, 11-6
Oracle databases, 8-9
SQL Server databases, 9-6
Sybase ASE databases, 10-6
dropping
Audit Vault collection agents, 6-5
collectors from IBM DB2 databases, 11-5
collectors from Oracle Database, 8-9
collectors from SQL Server, 9-6
collectors from Sybase ASE, 10-5
DV_ACCTMGR role
about, 1-10
DV_ACCTMGR roles and privileges granted
about, 5-3
DV_OWNER role
about, 1-10
roles and privileges granted, 5-3

E

environment variables
LANG, 2-3
LD_LIBRARY_PATH, 2-2
LIBPATH, 2-2
ORACLE_HOME, 2-2
ORACLE_SID, 2-2
PATH, 2-2
SHLIB_PATH, 2-2
error and log files
Audit Vault collection agent
{collector-name}{source-name}{source-id}.log, A-2
agent.err, A-3
agent.out, A-3

av_client-0.log, A-3
avca.log, A-3
avorcldb.log, A-2
sqlnet.log, A-2
Audit Vault Server
agent.out, A-1
av_client-0.log, A-1
avca.log, A-1
log file location, A-1
debugging advice for Audit Vault collection agents, A-6
debugging Audit Vault Console, A-10
error messages, B-1 to B-13
failed commands
configToolFailedCommands, A-4
OC4J
AVAgent-access.log, A-4
operational errors, 3-5
Oracle Enterprise Manager
logging, A-2
sqlnet.log, A-4
See also troubleshooting tips
examples
scheduling audit data refresh, 3-8
See also reference chapters for utilities

F

flash recovery area
monitoring in Audit Vault Server, 4-2

G

generate_csr command (AVCA utility), 5-9, 6-6
GET_AUDIT_COMMIT_DELAY function, 14-9
granting required privileges
for policy management, 2-6
to source database user
for the REDO collector, 2-6
to source database user for the DBAUD collector, 2-6
to source database user for the OSAUD collector, 2-6

H

-help command
AVCA utility, 6-6
AVCTL utility, 7-2
AVDB2DB utility, 11-7
AVMSSQLDB utility, 9-7
AVORCLDB utility, 8-10
AVSYBDB utility, 10-7
HTTPS communication, 5-8, 5-9

I

IBM DB2 databases
adding collector to Oracle Audit Vault, 2-21
compatibility with collector, 2-20
converting binary audit file to ASCII text file, 2-22
creating user account, 2-20
IBM Data Server Driver for JDBC and SQLJ, 2-19
jar file missing, A-8
modifying collector attributes, 3-6 to 3-7
modifying source attributes, 3-15 to 3-17
planning configuration, 1-13
registering with Oracle Audit Vault, 2-19 to 2-23
removing from Oracle Audit Vault, 3-17 to 3-18
setting up in collection agent home, 11-7
source database errors, B-2
unable to connect to source database, A-9
verifying compatibility with collector, 11-8
See also
Audit Vault IBM DB2 (AVDB2DB) utility
import_cert command (AVCA), 5-9, 6-8
INIT_CLEANUP procedure, 14-10
initialization parameters
hidden
redo log audit source release 10.1, 12-5
redo log audit source release 10.2, 12-9, 12-13
redo log audit source release 9.2, 12-1
redo log
audit source release 10.1, 12-5
audit source release 10.2, 12-9, 12-14
audit source release 9.2, 12-2
init.ora parameters
See initialization parameters
IS_CLEANUP_INITIALIZED function, 14-11

K

keytool utility, 5-9

L

LANG environment variable, 2-3
languages supported, 2-3
LD_LIBRARY_PATH environment variable, 2-2
LIBPATH environment variable, 2-2
load_warehouse command (AVCTL utility), 7-3
log file locations and descriptions
Audit Vault collection agent, A-2
Audit Vault failed commands, A-4
Audit Vault Server, A-1

M

managing collection agents and collectors
starting
collection agents, 7-9
collectors, 7-10
stopping
collection agents, 7-12
collectors, 7-13
Microsoft SQL Server databases
checking collector status, 2-27
collection agent credentials, adding, 2-15
collector, adding to Oracle Audit Vault, 2-15
compatibility with collector, 2-13
creating user account, 2-13

downloading SQL Server 2005 Driver, 2-12
jar file missing, A-8
modifying collector attributes, 3-6 to 3-7
modifying source attributes, 3-15 to 3-17
planning configuration, 1-12
registering with Oracle Audit Vault, 2-12 to 2-16
removing from Oracle Audit Vault, 3-17 to 3-18
setting up in collection agent home, 9-8
source database errors, B-2
unable to connect to source database, A-9
verifying compatibility with collector, 9-9
See also Audit Vault Microsoft SQL Server Database (AVMSSQLDB) utility
MSSQLDB collector
about, 1-6
MSSQLDB collector attributes, 9-3

O

OC4J agent
about, 1-4
checking status, 7-8
collection agent components, 1-5
failing to start, A-6
how it fits in general process flow, 1-8
log file locations, A-4
requirements for IBM DB2, 2-19
requirements for SQL Server, 2-13
requirements for Sybase ASE, 2-16
starting, 7-11
stopping, 7-12
using in Windows, 2-4
when starting Audit Vault Console, 3-2
when starting collection agents, 2-24
operating system audit trail
age, controlling, 4-15
size, controlling, 4-14
Oracle Audit Vault
administrative tasks for high volume systems, 4-1 to 4-16
components, 1-3
configuring source databases, 2-1 to 2-28
how administrators use, 1-1
how components work together, 1-7
how it is secured, 5-1 to 5-10
languages supported, 2-3
maintenance tasks, 3-1 to 3-18
roles, 1-9
tools, 1-9
See also entries beginning with Audit Vault
Oracle Audit Vault clients
error messages, B-7 to B-13
Oracle Container for Java
See OC4J agent
Oracle Database Vault
how it implements security, 5-2
password file requirement for source database, 2-5
Oracle Database Vault administrator roles
DV_ACCTMGR
about, 1-10
DV_ACCTMGR role
roles and privileges granted, 5-3
DV_OWNER
about, 1-10
roles and privileges granted, 5-3
Oracle databases
adding collectors to Oracle Audit Vault, 2-9
checking collector status, 2-27
collection agent credentials, adding, 2-11
collectors not working, A-7
compatibility with collectors, 2-7
creating user account, 2-5
database audit trail
batch size for records during purging, 4-9
tablespace, moving to one other than SYSTEM, 4-6
modifying collector attributes, 3-6 to 3-7
modifying source attributes, 3-15 to 3-17
password file for Oracle Database Vault, 2-5
planning configuration, 1-11
purging audit data, 4-4 to 4-16
registering with Oracle Audit Vault, 2-5 to 2-12
removing from Oracle Audit Vault, 3-17 to 3-18
setting up in collection agent home, 8-11
source database errors, B-2
unable to connect to source database, A-9
verifying compatibility with collectors, 8-12
See also Audit Vault Oracle Database (AVORCLDB) utility
Oracle Enterprise Manager Database Control console, 5-1
Oracle Real Application Clusters
avca add_agent command failing on node, A-10
configuring collection agent connectivity for, 4-4
configuring collection agents for, 4-3
deploying av.ear file to nodes, 6-4
troubleshooting tips, A-10
Oracle wallets
creating, 6-3
creating credentials, 6-2
credentials not successful, A-7
generating certificate requests, 5-8, 6-6
how Oracle Audit Vault uses, 5-1
importing certificate requests, 6-8
locations, 5-1
removing certificates, 6-9
ORACLE_HOME environment variable, 2-2
ORACLE_SID environment variable, 2-2
oraenv UNIX script, 2-2
orapwd utility, 2-5
OSAUD collector
about, 1-6
attributes, 8-6
OSAUD collector attributes, 8-5

P

passwords
guidelines for changing, 5-4

PATH environment variable, 2-2
performance tuning
Audit Vault Server, A-5
database audit trail, moving to different tablespace, 4-6
purge_warehouse command (AVCTL utility), 7-4
purging Oracle source database audit records
all audit trail types
creating purge job, 4-7
database audit trail
batch size, clearing, 4-12
batch size, setting, 4-9
DBMS_AUDIT_MGMT package, downloading, 4-5
general steps, 4-4
initializing
canceling, 4-12
checking if done, 4-10
cleanup operation, 4-7
purge jobs, managing
deleting, 4-13
enabling or disabling, 4-10
time interval for all purge jobs, 4-11
time interval for named purge job, 4-12
tracing debug levels, setting, 4-13

R

redeploy command (AVCA utility), 6-9
REDO collector
about, 1-6
attributes, 8-7
refresh_warehouse command (AVCTL utility), 7-5
remove_cert command (AVCA utility), 6-9
removing
source databases from Audit Vault, 3-17 to 3-18
roles used with Oracle audit Vault, 1-9

S

scheduling audit collections
See Audit Vault data warehouse
secure_agent command (AVCA), 6-10
secure_av command (AVCA), 6-11
securing
Audit Vault by mutual authentication, 6-11
Audit Vault Collection Agent by mutual authentication, 6-10
Audit Vault collection agents, 6-10
Audit Vault Server, 5-9, 6-11
server.xml file, 5-1
SET_AUDIT_TRAIL_LOCATION procedure, 14-11
SET_AUDIT_TRAIL_PROPERTY procedure, 14-12
SET_DEBUG_LEVEL procedure, 14-14
SET_LAST_ARCHIVE_TIMESTAMP procedure, 14-15
SET_PURGE_JOB_INTERVAL procedure, 14-16
SET_PURGE_JOB_STATUS procedure, 14-17
set_warehouse command (AVCTL utility), 6-13
set_warehouse_retention command (AVCA utility), 6-12
set_warehouse_schedule command (AVCA utility), 6-13
setup command
IBM DB2 databases, 11-7
Oracle databases, 8-11
SQL Server databases, 9-8
Sybase databases, 10-7
SHLIB_PATH environment variable, 2-2
show_agent_status command (AVCTL utility), 7-6
show_av_status command (AVCTL utility), 7-7
show_collector_status command (AVCTL utility), 2-27, 7-7
show_oc4j_status command (AVCTL), 7-8
source databases
about, 1-3
altering collector attributes, 3-6 to 3-7
altering source database attributes, 3-15 to 3-17
general steps for adding to Oracle Audit Vault, 2-1
removing from Audit Vault
about, 3-17
using Audit Vault Console, 3-17
using shell, 3-18
supported database products, 1-3
UNIX environment variable settings, 2-2
See also IBM DB2 databases, Microsoft SQL Server databases, Oracle databases, and Sybase ASE databases
SQL*Net
log file, A-4
start_agent command (AVCTL utility), 2-25, 7-9
start_av command (AVCTL utility), 3-2, 7-9
start_collector command (AVCTL utility), 2-26, 7-10
start_oc4j command (AVCTL utility), 7-11
starting
Audit Vault collection agent, 7-9
Audit Vault Console, 3-2
collectors, 7-10
state of
Audit Vault collection agents, 7-6
Audit Vault Console, 3-2, 7-7
collectors, 7-7
OC4J, 7-8
stop_agent command (AVCTL utility), 7-12
stop_av command (AVCTL utility), 3-3, 7-12
stop_collector command (AVCTL utility), 7-13
stop_oc4j command (AVCTL utility), 7-14
stopping
Audit Vault collection agents, 7-12
Audit Vault Console, 3-3
collectors, 7-13
Sybase ASE databases
adding collector to Oracle Audit Vault, 2-18
collector status, checking, 2-27
compatibility with collector, 2-17
creating user account, 2-17
jar file missing, A-8
jConnect for JDBC driver, 2-16
modifying collector attributes, 3-6 to 3-7

modifying source attributes, 3-15 to 3-17
planning configuration, 1-13
registering with Oracle Audit Vault, 2-16 to 2-19
removing from Oracle Audit Vault, 3-17 to 3-18
setting up in collection agent home, 10-7
source database errors, B-2
unable to connect to source database, A-9
verifying compatibility with collector, 10-8
See also
Audit Vault Sybase ASE Database (AVSYBDB) utility
SYBDB collector
about, 1-6
SYBDB collector attributes, 10-3
SYS account
how Oracle Audit Vault handles, 5-3
SYSAUX tablespace
monitoring in Audit Vault Server, 4-1
moving Oracle database audit trail tables to, 4-6
SYSDBA privilege
how Oracle Audit Vault handles, 5-3
SYSMAN account
how Oracle Audit Vault handles, 5-3
SYSOPER privilege
how Oracle Audit Vault handles, 5-4
SYSTEM account
how Oracle Audit Vault handles, 5-3

T

tnsnames.ora file
updated by avorcldb setup command, 8-11
troubleshooting
collector not starting, 7-10
troubleshooting tips
Audit Vault collector, A-7 to A-9
Audit Vault Console, A-9
Audit Vault in an Oracle RAC environment, A-10
Audit Vault Server, A-5
error messages, B-1 to B-13
finding detailed information about an error, A-4
finding Oracle Database trace files, A-4
viewing operational errors, 3-5
See also errors and log files

U

UNIX environment variable settings, 2-4

V

verify command
IBM DB2 databases, 11-8
Oracle databases, 8-12
SQL Server databases, 9-9
Sybase ASE databases, 10-8
verifying source database compatibility
with DB2 collector, 11-8
with Oracle Database collectors, 8-12
with SQL Server collector, 9-9
with Sybase ASE collector, 10-8
viewing
Audit Vault errors, 3-5

W

wallets
See Oracle wallets