An Oracle White Paper
June 2013

Oracle Audit Vault and Database Firewall

ORACLE

Introduction

Cyber threats, privacy laws and well known regulations such as Sarbanes-Oxley (SOX) and Payment Card Industry Data Security Standard (PCI-DSS) have resulted in information protection becoming a top-level issue for the enterprise. The 2012 Data Breach Investigations Report by the Verizon RISK Team showed that 94% of all data compromised involved servers. This and various studies and surveys conducted by government and academic institutions have concluded that a sizeable percentage of data breaches have been perpetrated using SQL injection, stolen credentials or by insiders who are authorized access to the system and its data. Securing data on servers requires a defense-in-depth approach involving both technical and administrative functions that span preventive, detective, and administrative controls.

The principle of trust-but-verify not only applies to privileged users who have direct access to the host and database but also to applications accessing the database. Most applications today operate as highly trusted users, using one-big user account for communicating with the database, Oracle or non-Oracle. This application architecture, combined with the increasing number of attacks on databases via SQL injection or privileged user accounts, makes deploying detective controls a crucial part of the overall defense-in-depth security strategy.

When deploying a monitoring solution, it is important to note that the quality and accuracy of the information gathered will depend on the level of visibility the solution has into the activities of the target system. It is also important to understand the risk associated with individual systems so that you can determine the level of visibility required for activities on those systems. A good analogy to understand this concept is to consider the cameras or guards at the front entrance to buildings. Both can view what is going into the building, only one can stop what goes into the building, but neither can provide a complete view on what happened inside the building. If the building were a database, the camera or guard could monitor SQL statements before they reach the database, but the challenge is in finding out what happens after the SQL executes inside the database. Recursive SQL spawned by stored procedures, dynamic SQL, privileged user operations, scheduled jobs, trigger executions, application user names, as well as "before" and "after" data values are all examples of information that is largely invisible from outside the database, but are visible to the auditing system inside the database. As a result, the value of monitoring is directly linked to the level and quality of information gathered as well as available reporting and alerting functionality.

Oracle Audit Vault and Database Firewall Overview

Oracle Audit Vault and Database Firewall provides comprehensive and flexible monitoring through consolidation of audit data from Oracle and non-Oracle databases, operating systems, directory, file systems, and application specific audit data. At the same time Oracle Database Firewall can act as a first line of defense on the network, enforcing expected application behavior, helping prevent SQL injection, application bypass, and other malicious activity from reaching the database. Oracle Audit Vault and Database Firewall can consolidate audit data from thousands of databases and monitor SQL traffic at the same time, looking for, alerting on, and preventing unauthorized or out-of-policy SQL statements. Dozens of out of the box reports combined with a custom reporting interface provide a comprehensive view of database activity across the enterprise whether observed through the network, or through the audit logs. Oracle Audit Vault and Database Firewall supports Oracle database, Microsoft SQL Server, IBM DB2 for LUW, SAP Sybase ASE and Oracle MySQL databases.

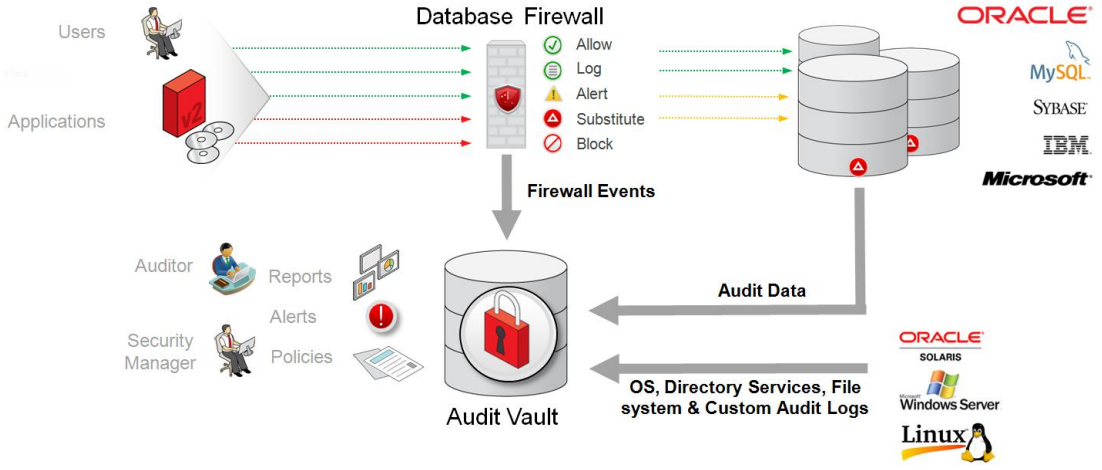

Figure 1: Oracle Audit Vault and Database Firewall

Auditing and Monitoring Overview

Auditing has become an important tool over the past 10 years for both compliance and forensic analysis of data breaches. Audit records provide an irrefutable record of actions taken whether they are generated by a database, directory, or operating system. Information such as the event type (create table, drop table, create procedure, truncate table, select, insert, update, delete) coupled with the context of the event such as the initiating IP address, event time, and actual SQL statement, are just a few examples of audit information that is commonly needed in compliance and forensic reports (see Figure 2). Oracle Audit Vault and Database Firewall can consolidate, report, and alert on audit information from databases, operating systems, file systems, and directories.

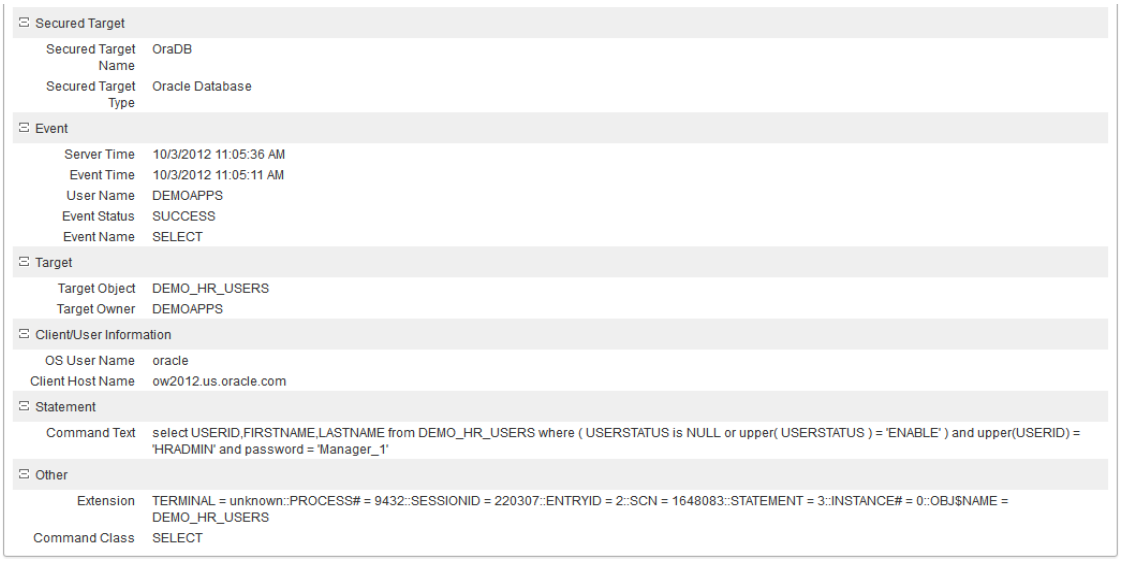

Figure 2: Sample Audit log in Oracle Audit Vault and Database Firewall

Monitoring includes the examination of the initiating events (SQL statements) that generate the audit data. Because monitoring of SQL traffic is done outside the database, Database Firewall can decide whether an event should be permitted, modified, blocked, or alerted on. Figure 3 shows an example of a report where a SQL injection attempt is blocked.

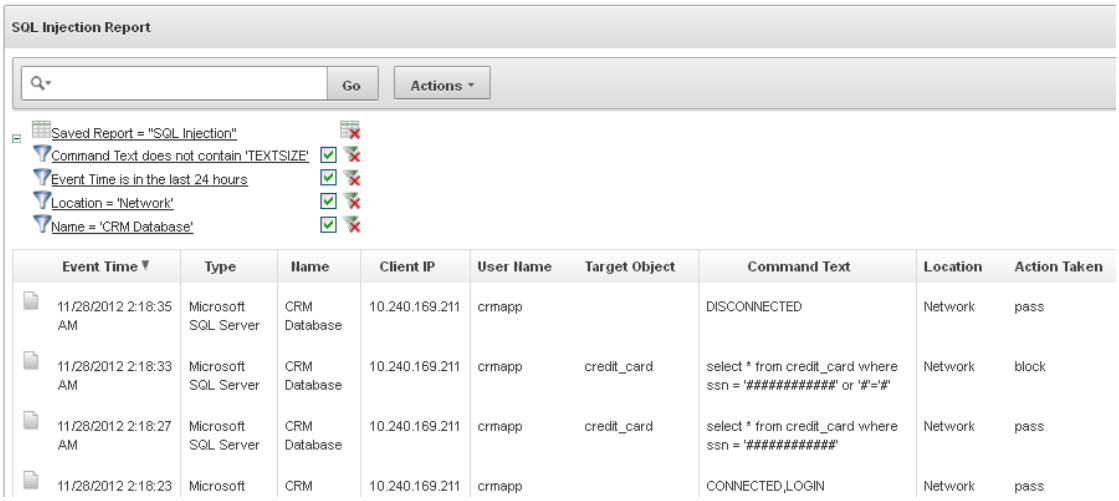

Figure 3: Database Firewall SQL Monitoring

Audit Vault

The Audit Vault is the central, highly scalable and secure repository that stores the consolidated audit data as well as event logs generated by the Database Firewall. The Audit Vault is the central platform for reporting, alerting, and policy management. Using lightweight agents, the audit data is transferred from the target system, and then optionally removed from the target system. The Audit Vault can

consolidate audit information from all database sources and can be extended to custom sources, including application tables/files on Oracle and non-Oracle databases that log custom audit data. As shown in Figure 4, report data can span across multiple databases and include information from the target system and the network.

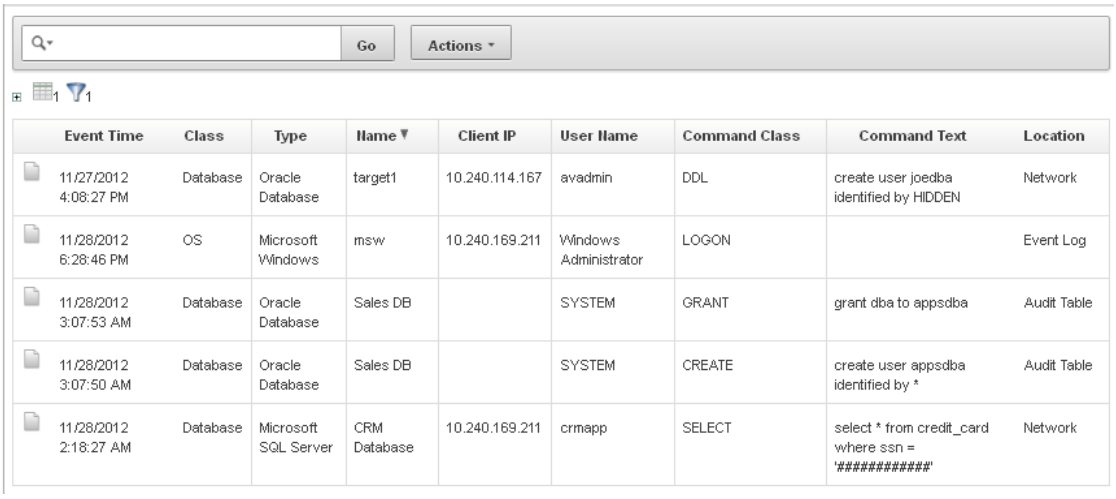

Figure 4: Consolidation from Network, Database Audit and OS Event Logs

Using the Audit Vault console, multiple targets are identified (Figure 5). The console is then used to manage the Database Firewall policies, schedule and customize the reports, set up the report attestation, and configure the alerts.

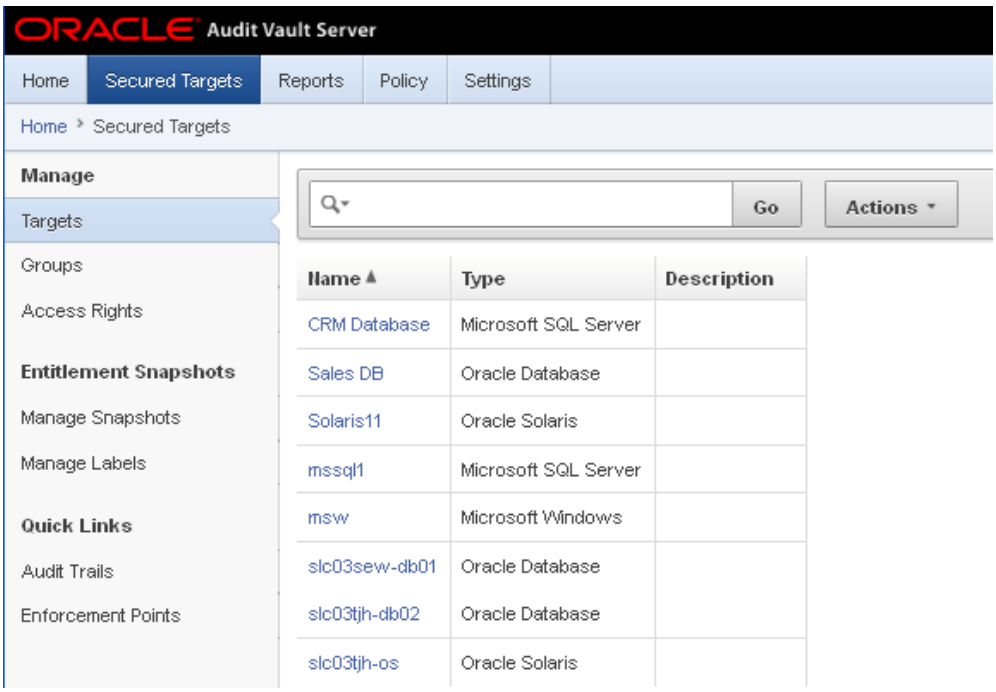

Figure 5: Oracle Audit Vault and Database Firewall Console showing Secured Targets

Database Firewall

The Database Firewall is the network monitoring component outside the database that monitors the inbound SQL traffic and serves as a first line of defense against SQL injection threats and other unauthorized SQL statements. Database Firewall monitors data access, enforces access policies, highlights anomalies and helps protect against network based attacks originating from outside or inside the organization. Unlike traditional SQL firewalls that rely on identifying out-of-policy SQL using regular expressions, the Oracle Database Firewall enforces policies using a sophisticated grammar analysis engine that delivers the required scalability, accuracy, and management simplicity.

Organizations can choose to deploy Database Firewall in active monitoring mode to protect their database assets or in passive monitoring mode to alert security operations personnel of unexpected activity, and/or supplemental auditing to address compliance requirements. In passive monitoring mode, Database Firewall observes database traffic and analyzing SQL interactions. Information from Database Firewall is logged to the Audit Vault, enabling reports to span information observed on the network alongside audit information from the database, operating systems, and directories.

In active monitoring mode, Database Firewall transparently intercepts SQL traffic coming from database clients acting as an application layer firewall, analyzes the security of the SQL payload in TCP packets before forwarding it on to the database. Attacks including SQL injection can be blocked by comparing incoming SQL against the approved white list of application SQL. Support for white list, black list, and exception list based policies, provides a high degree of deployment flexibility.

White List Policy Enforcement

The white list policy enforces security using a set of approved SQL statements along with the conditions under which they were executed including the username, IP address, time of day, and program name. Database Firewall compares SQL traffic with the approved white list and then based upon the policy, it chooses to alert, substitute, or block the SQL statement. The approved SQL or white list is learned over time by monitoring database traffic. The monitoring period needed to establish the white list will vary depending on the application and business cycle.

Black List Policy Enforcement

In addition to the white list based positive security enforcement model, Database Firewall also supports a black list model that blocks specific SQL statements. As with white list policies, black list policies can evaluate various factors such as username, IP address, time of day and program, before making the decision.

Exception List Policy Enforcement

Exception lists policies override white list and black list policies by allowing custom bypass policies to be created for specific activities. For example, exception list policies could be used to enable a specific remote administrator coming from a predetermined IP address to diagnose a particular application performance issue without being bound by the white list or the blacklist.

Handling Unauthorized SQL

When Oracle Database Firewall finds an unauthorized statement, it handles the statement in the one of the following ways:

- Terminate the connection: This blocks all traffic from that specific database connection to go to the server as it kills the connection. This is the most aggressive action and if the application is using connection pooling, this may impact all the users using the pool.
- Block the SQL statement: This specific statement is stopped from reaching the database server. The actual end-user experience would depend upon how application handles this case where the server does not respond. Client connection to the database can be configured to either maintain or terminate.
- Modify the request using SQL statement substitution by replacing an out-of-policy statement with a new harmless statement that does not return any data or returns an error (as shown in Table 1 below). Statement substitution is more transparent to the existing application.
- Alert on all out of policy SQL statements, in addition to or instead of blocking

ORIGINAL STATEMENT (FRAUDULENT)	SUBSTITUTED STATEMENT	DATABASE RESPONSE (RESULT)
SELECT * FROM tbl_users;	SELECT * FROM tbl_users WHERE 'a' = 'b';	No record found
DROP TABLE tbl_accounts;	SELECT * FROM aaabbbccc;	Error. Table not known
UPDATE tbl_accounts SET accounts = '123' WHERE user = 'Fred';	SELECT DUAL SET 'Fred';	Error. Incorrect Syntax.

Table 1: Oracle Audit Vault and Database Firewall SQL Statement Substitution Examples

Reports

Oracle Audit Vault and Database Firewall reports can be used to monitor a wide range of activity including privileged user activity on the database server, changes to database structures, and data on inbound SQL statements on the network. Reports can display consolidated audit information from databases, operating systems, and directories, providing a holistic picture of activities across the enterprise. In addition, reports can include information on database account management, roles and privileges, object management, and stored procedure changes.

Auditors can access reports interactively through an Auditor Console web interface, or through PDF or XLS report files. The easy-to-use interactive browsing is built on Oracle Application Express technology with the ability to create color-coded charts and graphs. Report columns can be sorted, filtered, re-ordered, added, or removed. Rules can automatically highlight specific rows so that users can quickly spot suspicious or unauthorized activity.

PDF and XLS report definitions can be used to schedule automatic report generation. Reports can be scheduled and delivered via e-mail attachments or URLs. Reports can also be defined to require attestation by multiple auditors. Users can use Oracle BI Publisher to create new or customize PDF and XLS report templates to meet specific compliance and security requirements. Furthermore the Audit Vault repository schema is documented, enabling integration with third-party reporting solutions.

Compliance Reports

Standard out-of-the-box audit assessment reports are categorized to help meet standard regulations such as Payment Card Industry Data Security Standard (PCI-DSS), Gramm-Leach-Bliley Act (GLBA), Health Insurance Portability and Accountability Act (HIPAA), Sarbanes-Oxley Act (SOX), and European Union Data Protection Act (DPA).

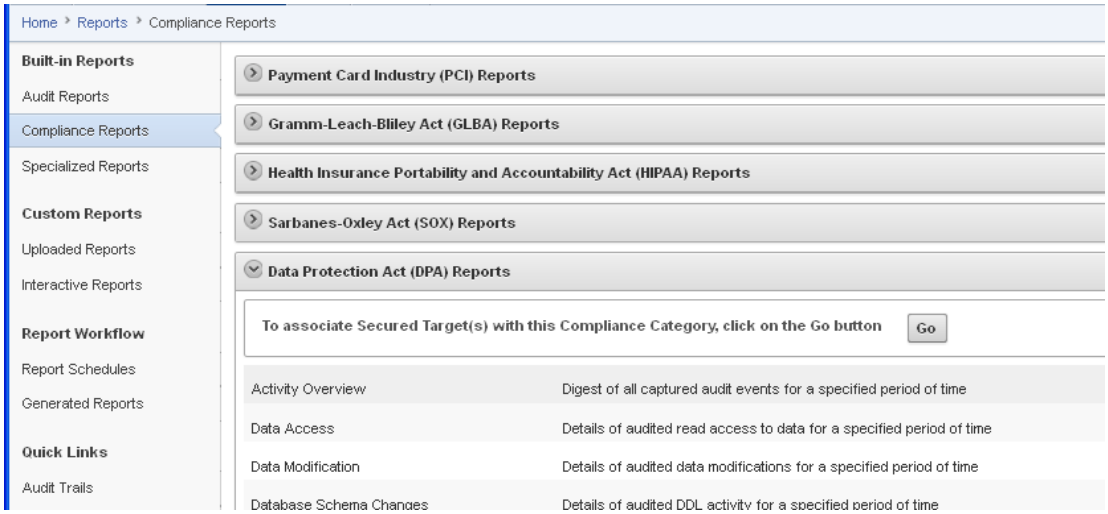

Figure 6: Oracle Audit Vault and Database Firewall Built-in Compliance Reports

Activity Reports

Activity Reports cover topics such as failed logins, changes to application tables, database schema changes, or user entitlements (see Figure 7). For example, if you want to audit each time a user performs data definition language (DDL) SQL statement such as DROP or ALTER, the pre-built "Database Schema Changes Report" highlights rows of that particular user and drills down to individual event details. You can also get an overview of all audit events which can be filtered further by target system, user, operation, time, and so on.

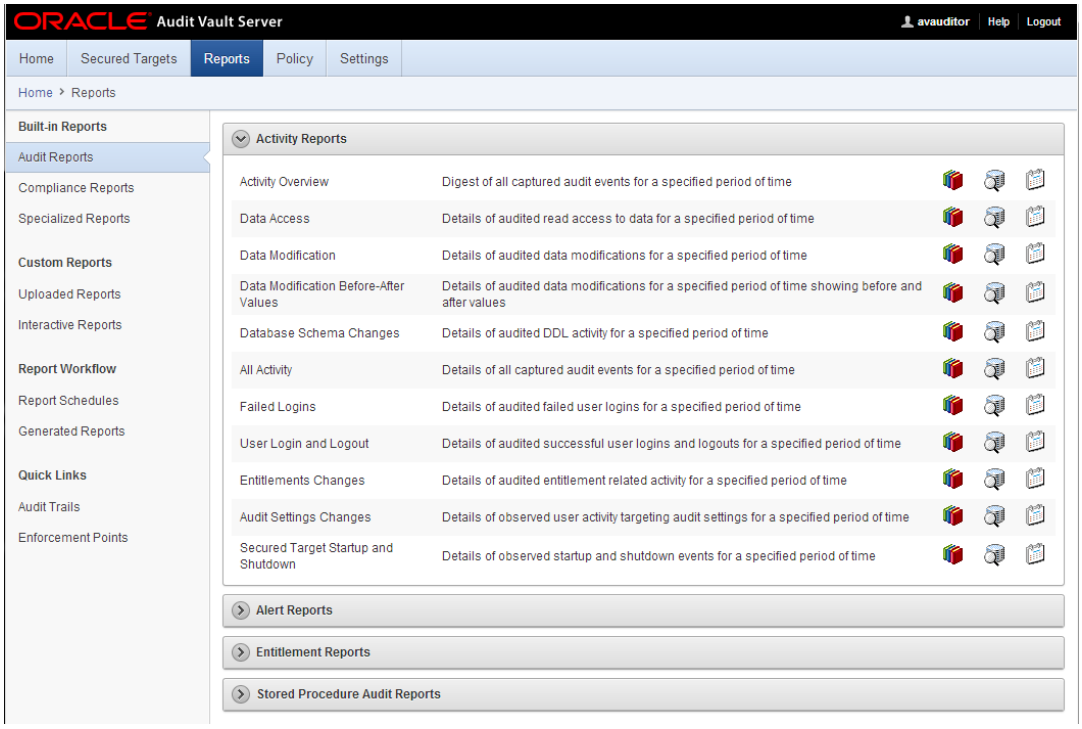

Figure 7: Oracle Audit Vault and Database Firewall Built-in Activity Reports

Entitlements Reports

Entitlement reports describe the types of access that users have to an Oracle database. It provides information about the users, roles, profiles, and privileges used. These reports are useful for tracking unnecessary access to data, finding duplicate privileges, and simplifying privilege grants. After you generate an entitlement snapshot, you can compare different snapshots to find how the entitlement information has changed over time. This is particularly useful to identify the drift from an already approved database entitlement baseline.

Stored Procedure Audit Reports

For many organizations, stored procedures form the bulk of the application logic for many applications and may contain flaws that can be exploited for malicious attacks including SQL injection. DBAs often write custom triggers or stored procedures to automate the jobs or to improve security. It is important that these stored procedures once defined are not tampered with. With Audit Vault and Database Firewall, you can run Stored Procedure Auditing Report to monitor any changes made to the Stored Procedures on the protected databases. This report shows all stored procedure operations, deleted and created procedures, as well as modification history.

Alerts and Notifications

Audit Vault provides the ability to detect and alert on activities that may indicate attempts to gain unauthorized access and/or abuse system privileges. Database Firewall policies can be configured to generate alerts on network activity, providing an early warning detective control for potential malicious activity. Furthermore Audit Vault continuously monitors the events collected, evaluating the activities

against defined alert conditions. Alerts can be associated with any database event including system events such as changes to application tables, creating privileged users, or events when someone attempts to access sensitive business information and is blocked by an Oracle Database Vault policy.

As shown in Figure 8, alerts can also be configured to be threshold and time based. For example, if 10 login failures occur within a 1 minute window, possibly indicating a brute force attack, then an alert is raised.

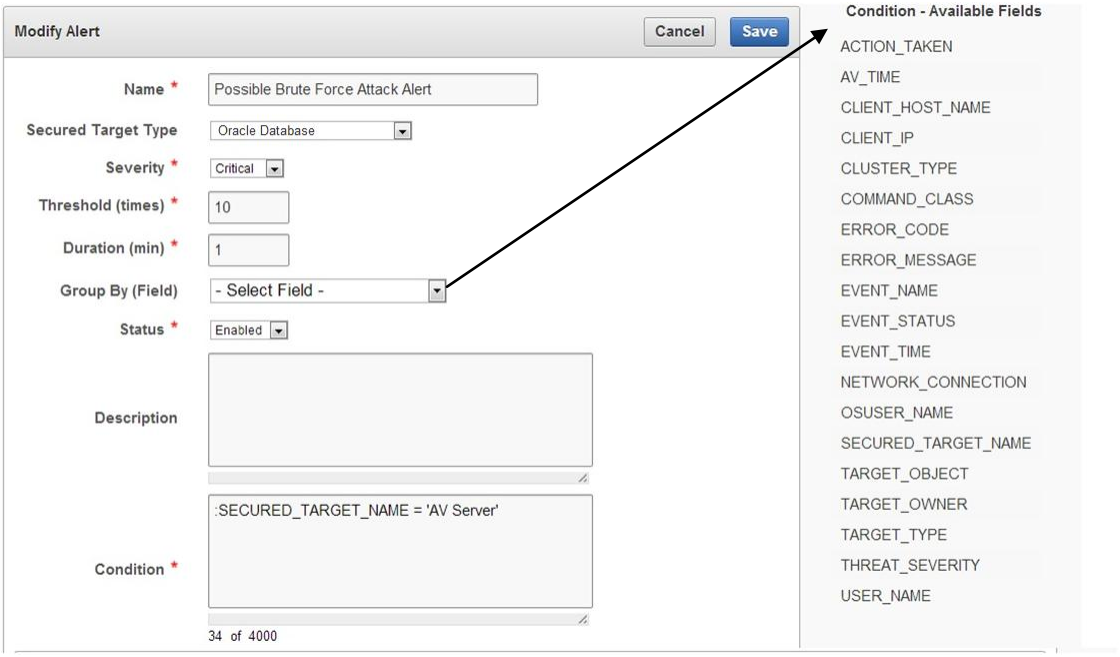

Figure 8: Oracle Audit Vault and Database Firewall Alert Definition

Audit Vault interface provides graphical summaries of alerts. These include a summary of alert activity and top sources by number of alerts. Users can click on the summary graphs and drill down to a more detailed report. For the purpose of reporting, alerts can be grouped by source, event category, and severity (warning or critical).

Scalability and Security

Audit data is an important record of business activity, and it must be protected against modification to ensure the integrity of reports and investigations. Audit Vault stores audit data in a secure repository built using Oracle's industry leading database technology. Timely transfer of audit data from source systems to Audit Vault Server is critical to close the window on intruders who may attempt to modify audit data and cover their tracks. Audit Vault can be configured to transfer audit data on a near real time basis. Audit Vault can also be configured to encrypt data during transmission.

The repository is built on an embedded Oracle Enterprise Edition database that includes numerous Oracle technologies, including compression, partitioning, encryption, and privileged user controls. The use of compression is particularly important for optimized storage of the consolidated data. The

combination of these technologies and the Oracle Enterprise Edition database results in a repository with massive scalability.

A single Audit Vault can scale to support hundreds of Audit Agents and Database Firewalls, each of which can in turn host multiple audit trails and hundreds of databases correspondingly. The integrated administrator console can configure the entire system, monitor the deployment, startup/shutdown Database Firewalls and Agents, configure Database Firewall HA operation, and manage the backup and restore operations.

The Audit Vault interface supports two broad categories of users: Auditors and Administrators. Auditors configure auditing and monitoring policies, define, generate and access audit reports and alerts. Administrators configure basic network and host settings for the secured targets, start and stop agents and Database Firewalls, and configure and monitor Audit Vault Server operation. Administrators do not have access to audit information. Within the two role categories, further separation of duties can be defined. A subset of protected assets can be assigned to individual auditors and administrators, ensuring that a single repository can be deployed to support an entire enterprise spanning multiple organizations, subsidiaries, or geographic regions. Fine grained authorizations are particularly important when information may span multiple countries with different privacy regulations and safe harbor requirements.

Deployment

Database Firewall Network Deployment

Database Firewall can be deployed as a transparent network bridge, simply inserted into the network in a segment that lies between database clients/application servers and the databases being protected (as shown in figure 10). This 'in line' bridge architecture requires no configuration changes to database clients, applications or the database itself, and provides the flexibility for both active and passive monitoring. If you are looking only for passively monitoring the database activity, then it is also possible to forward the traffic to the database firewall using the span-port.

In scenarios where it is difficult to add a network bridge, or if the database servers are in some remote places, Database Firewall can also be configured as a proxy such that all traffic to the database server is routed through the database firewall. This requires the Database Server IP address/port on the database client or application to be changed to the IP address/port for the Database Firewall proxy, along with changes to the database listener to reject direct connections. Most enterprise network switches and traditional firewalls can also be used to redirect database traffic to an Oracle Database Firewall proxy port, allowing SQL traffic to be protected without any changes to database clients or applications. A given database firewall can operate as a transparent bridge for some databases and a proxy for others.

Database Firewall supports a local server-side, monitor-only agent to ensure flexibility in the choice of the network point at which the traffic is monitored. Host Monitor, part of the Audit Agent, captures SQL traffic reaching the database server and securely forwards it to the Database Firewall. It can be used to remotely monitor database servers running on Linux and Windows platforms.

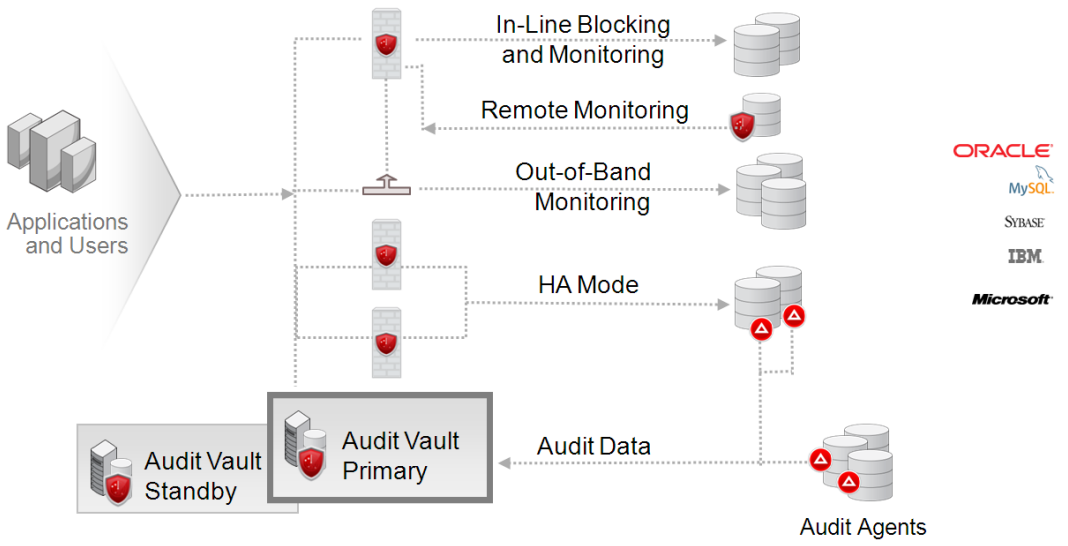

Figure 9: Oracle Audit Vault and Database Firewall Deployment

Audit Agents Deployment

The audit agents are distributed as JAR files to the target systems and require no additional manual configuration or updates once they have been distributed. The agents collect the audit data from the various sources including Oracle, non-Oracle, operating systems and directories. For Oracle databases SE or EE, the agents work independently of how the auditing is configured. For example, auditing can be configured to write audit data to the operating system or database. In addition, for Oracle databases, the agents can consolidate the "before" and "after" values for specific fields using the transaction or REDO logs and database entitlement information.

Policy Authoring and Management

Audit Vault provides integrated management interface for Database Firewall policies. Users can define a white list, black list, or exception list of SQL statements for a given database. The Database Firewall Policy Authoring interface can analyze all the captured SQL statements within a time period so that appropriate policies can be specified. It also allows factors such as user names, IP addresses, client programs, and time of day to be associated with policies for SQL statements.

Audit Vault can centrally define and provision audit settings for Oracle databases. This provides both internal auditors and IT security a much easier way to manage audit settings across the enterprise and demonstrate compliance and repeatable controls to external auditors.

Custom Audit Collection Plug-ins

Developers and third-party vendors can build custom collection plug-ins to collect audit data from a new secured target type or a new audit trail where audit data is stored in database tables and XML files. Secured target type can be relational databases, operating systems, mid-tier systems, or enterprise applications. No coding is required as you can easily define a template-based XML mapper file to

describe the audit data being collected and whether to store the audit data either in database tables or XML files. Example 1 below shows a sample manifest file for an XML file collection plug-in.

```
<?xml version="1.0"?>
<plugin xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
          xsi:schemaLocation="http://xmlns.oracle.com/av/plugin plugin-manifest.xsd"
          xmlns="http://xmlns.oracle.com/av/plugin"
           name="Oracle-XML-Template">
 <targetVersion min="11.1.0.0.0"/>
 <extensionSet>
          <extensionPoint type= "securedTargetType">
 <fileList>
 <templates>
           <include file="XMLSource-Mapper.xml"/>
 </templates>
 </fileList>
          <securedTargetTypeInfo name="oracle"/>
          <trailInfo>
          <trailType>DIRECTORY</trailType>
          <className name="oracle.av.platform.agent.collfwk.Collector.xml.XMLFileCollector"/>
          </trailInfo>
          <eventPatch name="p6753288_11.1.2.0.0_GENERIC.zip" order="2"/>
          </extensionPoint>
 </extensionSet>
</plugin>
```

Example 1 : Sample Manifest File for an XML file collection plug-in

Integration with 3rd Party Solutions

Oracle Audit Vault and Database Firewall integrates with F5 BIG-IP Application Security Manager. The combination of Database Firewall and F5 BIG-IP Application Security Manager (ASM) enables security and monitoring for both application servers and databases within an enterprise. If an attack originates from the web user, Database Firewall reports provide the actual IP address and application user obtained from BIG-IP ASM enabling you to pinpoint the source of the attack.

The HP ArcSight Security Information Event Management (SIEM) system is a centralized system for logging, analyzing, and managing log messages from different sources. HP ArcSight can pull security alerts from Oracle Audit Vault and Database Firewall.

Conclusion

Oracle Audit Vault and Database Firewall helps organizations increase security by proactively monitoring database activity on the network and inside the database, protecting against SQL injection threats and automating the consolidation of audit data into a secure and scalable repository. Extensive reporting and alerting capabilities provide auditors and security personnel with access to detailed information and early warning alerts on potential malicious activity. Sources beyond databases can be monitored, with out-of-the-box support for consolidation of audit data from various operating systems and directory services. An extensible plug-in architecture enables custom audit sources to be added to the collection framework, enabling application specific audit data to be aggregated and reported together with other event data in the repository. Detective and preventive controls for databases can be dialed-up based on the security requirements of Oracle and non-Oracle databases.

Oracle Audit Vault and Database Firewall
June 2013
Author: Oracle
Contributing Authors: Oracle

Oracle Corporation
World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065
U.S.A.

Worldwide Inquiries:
Phone: +1.650.506.7000
Fax: +1.650.506.7200

oracle.com

Oracle is committed to developing practices and products that help protect the environment

Hardware and Software, Engineered to Work Together