ORACLE

ORACLE DOJO NR. 14

ORACLE BU CORE & CLOUD TECHNOLOGIES

Absicherung einer Oracle Datenbank

Cloud und On-Premises

Oracle Dojo ist eine Serie von Heften, die Oracle Deutschland B.V. zu unterschiedlichsten Themen aus der Oracle-Welt herausgibt.

Der Begriff Dojo [‘do:dʒo] kommt aus dem japanischen Kampfsport und bedeutet Übungshalle oder Trainingsraum. Als „Trainingseinheiten“, die unseren Anwendern helfen, ihre Arbeit mit Oracle zu perfektionieren, sollen auch die Oracle Dojos verstanden werden. Ziel ist es, Oracle-Anwendern mit jedem Heft einen schnellen und fundierten Überblick zu einem abgeschlossenen Themengebiet zu bieten.

Im *Oracle Dojo Nr. 14* beschäftigen sich Michael Fischer, Norman Sibbing, Volker Linz und Ralf Durben mit dem Thema Datenbank Sicherheit und erklären die Technologien zur Absicherung einer Datenbank in On-Premises und in Cloud Installationen.

ORACLE

Inhalt

ORACLE

ORACLE DOJO NR. 14

Absicherung einer Oracle Datenbank

Cloud und On-Premises

RALF DURBEN, MICHAEL FISCHER,
VOLKER LINZ, NORMAN SIBBING

VORWORT

Dieses Dojo hilft beim schrittweisen Herangehen an das Thema Datenbank Sicherheit und hat als Ziel, mit den vorhandenen Datenbank-Sicherheitstechnologien die benötigte Datensicherheit zu erreichen.

Dabei wird beiden Ansätzen Rechnung getragen: Bestehende Datenbankinstallationen zu untersuchen, als auch bei neu aufzusetzenden Umgebungen die erforderlichen Technologien zu identifizieren und einzusetzen. Nachdem der gewünschte Status erreicht ist, gilt es für beide Ansätze, die Installationen hinsichtlich erwarteter Nutzung und Konfiguration zu überwachen und sicherheitstechnisch aktuell zu halten.

Da die gleiche Oracle Datenbanksoftware in On-Premises Installationen, in Cloud Services (wie DBCS und DBaaS), in der Appliance Cloud@Customer und in den Autonomous Database Services (in ADW und ATP) eingesetzt wird, unterscheidet sich die Herangehensweise nur in den Konfigurationsmöglichkeiten. So wird beispielsweise in den Cloudumgebungen automatisch Verschlüsselung ohne Mehrkosten eingesetzt. Die unterschiedlichen Nutzungsmöglichkeiten sind in eigenen Kapiteln beschrieben.

Das Dojo folgt mit dem Aufbau einem Top-Down Ansatz. In diesem Dokument werden die Herangehensweise und Konzepte dargestellt. Ein zentraler Blogeintrag enthält weiterführenden Informationen zu den jeweiligen Kapiteln und Technologien. Auch finden sich dort Beispiele inklusive Hands-On Anleitungen. Den Blogeintrag finden Sie unter: *www.oraclecloud.de/dojo14*

Die länglichen Links im Dojo werden zur leichteren Verwendung verkürzt dargestellt.

Ich wünsche Ihnen viel Spaß beim Lesen, Testen und Ausprobieren der beschriebenen Technologien.

Ihr Michael Fischer
im Namen aller mitwirkenden Autoren

PS: Wir sind an Ihrer Meinung interessiert. Anregungen, Lob oder Kritik gerne an barbara.frank@oracle.com. Vielen Dank!

Zugriff auf alle Dojos: http://tinyurl.com/dojos-online

1 Oracle Datenbank und Security

Viele führende Unternehmen in den Bereichen Informationstechnologie, Datenbanken und Sicherheit stimmen heute damit überein, dass die Absicherung von Datenbanken eines ihrer wichtigsten Ziele ist. Schließlich sind es in den meisten Unternehmen Datenbanken, die die sensiblen Daten oder die Zugriffsmöglichkeit darauf enthalten.

Verizon veröffentlicht jedes Jahr einen sogenannten „Data Breach" Report. Dieser bestätigt die oben genannte Einschätzung der Wichtigkeit des Schutzes der Datenbanken. Datenbanken sind im Report das Ziel Nummer 1 und leider beweisen die entdeckten Datenabflüsse, dass Lücken vorhanden sind. Große Probleme stellen dabei ungepatchte Datenbanken dar. Laut dem Status Quo bei Datenbanken (Oracle und KPMG Cloud Threat Report 2018) nutzt lediglich die Hälfte der Befragten die seit längerem verfügbaren Datenbanksicherheitstechnologien. Im Gegensatz dazu setzt etwa ein Viertel der Befragten beim Monitoring weiterentwickelte Technologien wie Machine Learning ein oder evaluiert die Einführung eines automatisierten Betriebs.

In den letzten 20 Jahren hat sich die Art und Weise, wie Datenbanken kompromittiert werden, stark verändert. Als Antwort darauf hat Oracle mehrere **Sicherheitstechnologien zur Sicherung von Daten „an der Quelle"** (also

innerhalb der Datenbank) entwickelt. Die Technologien sind unterteilt in: Assessment (auch Überprüfung des Datenschutzes), Prevent (auch Schutzfunktionen) und Detect (auch kontinuierliches Monitoring inklusive der Möglichkeit automatisiert zu reagieren). Analysten wie Gartner und Kuppinger bestätigen die umfassenden Absicherungsmöglichkeiten bei Oracle Datenbanken.

Manche Sicherheitstechnologien oder auch das Nutzungsrecht sind bereits im Standardumfang der Datenbank enthalten oder kostenfrei beziehbar, andere als Zusatzlizenz erwerbbar. Die Zusammenstellung der Zusatzlizenzen ist bei On-Premises und Clouddatenbanken verschieden.

Mit den Sicherheitstechnologien der Datenbank können die genutzten Datenbanken durch das eigene Betriebsteam fortwährend abgesichert und überwacht werden. Zusätzlich stellt Oracle weitere Technologien speziell im Bereich **Sicherheitsmonitoring** (SIEM), Verhaltensanomalien (UEBA) und Konfigurationsabweichungen (IT Compliance) zur Verfügung. Diese gehen über das Auditing der Datenbanken hinaus und beziehen auch weitere Systeme in die Betrachtung mit ein. Um auf die sich ständig ändernden Konstellationen und Problemmuster reagieren zu können, sind dabei Mechanismen wie Machine Learning und Aktualisierung von Threat Informationen enthalten.

Der Trend geht auch bei Datenbanken zur Nutzung von Cloud Services. Im Prinzip gibt es dabei zwei Arten von Anforderungen oder Zielrichtungen: die erste möchte möglichst wenig oder kein Datenbank Know-How vorhalten, die zweite möchte die Aufwände einfach nach hinten oder an andere Verantwortliche verschieben. Dabei soll nicht nur der Aufwand für das Aufsetzen der Datenbanken zum Cloud Anbieter verschoben werden, sondern auch möglichst viele betriebsrelevante Aufgaben wie Backup und Restore bis hin zum Performanceoptimierungen und Patching. Oracle's Antwort darauf sind die sogenannten **cloud-basierten** Datenbankservices und je nach dem Grad der Übernahme der Betriebsthemen die **„autonomous"** Datenbank Services. Autonomous Systeme weisen einen höheren Automatisierungsgrad vor, der mit Self-Driving, Self-Securing und Self-Repairing gekennzeichnet ist. Bei beiden Typen, cloud-basiert und autonomous, kommt die gleiche Oracle Software zum Ansatz, so dass gewonnenes Know-How nahtlos in den anderen Umgebungen weiter genutzt werden kann.

Cloud-basierte Datenbanken werden sowohl im Oracle Cloud Datacenter, wie in Frankfurt, bereitgestellt als auch lokal in eigenen Rechenzentrum über die Oracle Cloud@Customer Appliance.

2 Vorgehen

Das folgende Kapitel widmet sich der Herangehensweise beim Aufbau des Schutzes einer Datenbank. Es werden zwei Fälle kurz betrachtet: das Aufsetzen einer neuen Datenbank und die Analyse eines bestehenden Systems bzw. einer Systemlandschaft. Im Abschluss des Kapitels finden Sie Überlegungen hinsichtlich der Verantwortlichkeit, die auch im Rahmen von gesetzlichen Vorgaben und der Cyber-Security Problematik stärker als bisher in den Vordergrund treten.

2.1 AUFSETZEN ODER ASSESSMENT

Datenbanken entstehen aus verschiedenen Anforderungen. Im Schaubild wird die geordnete und strukturierte Anlage angenommen. Das wenig überraschende Vorgehen passt dabei sowohl für On-Premises Installationen als auch für Datenbanken in der Cloud. Für schon bestehende Installationen oder ad hoc installierte Datenbanken dient ein Einstieg ab Schritt 4.

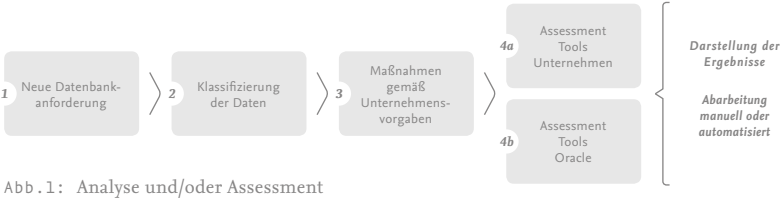

Abb.1: Analyse und/oder Assessment

Nach der Datenbankanforderung (Schritt 1) wird die Klassifizierung der Daten (Schritt 2) vorgenommen – durchgeführt entweder durch den Anforderer aus den Fachbereichen oder durch Analyse des gewünschten Datenmodells und der Rücksprache mit dem Datenschützer des Unternehmens. Seit der Einführung der DSGVO und vorher schon des BDSG gibt es die Rolle des Datenschützers oder die Verpflichtung dazu. Falls es keinen Datenschützer oder Verpflichtung zum Datenschutz im Unternehmen gibt, ist abzuwägen, ob die Entscheidung für die Klassifikation vom Auszuführenden, z.B. dem DBA, unter den gegebenen Haftungsaspekten übernommen werden soll.

Aus der Klassifikation, Unternehmensvorgaben und/oder aus den entsprechenden Regularien, z.B. DSGVO (personenbezogene Daten) oder PCI-DSS (Kreditkartenverarbeitung), leitet sich der Schutzbedarf ab, der die entsprechenden Maßnahmen spezifiziert (Schritt 3). Nicht alle Daten müssen gleich geschützt werden. Beispielsweise kann die Verschlüsselung und der Nachweis bezüglich der Zugriffe als Maßnahmen gesehen werden. International werden Daten mindestens nach den Klassen Vertraulichkeit, Integrität und Verfügbarkeit eingestuft. Bekannt ist hier das englische Kürzel CIA (für die Kategorien Confidentiality (C), Integrity (I), Availability(A)). Jede der Kategorien besitzt mindestens drei Abstufungen.

Folgendes Beispiel zeigt eine Einstufung von Daten:

- C3 I3 A1 für personenbezogene Daten
- C1 I2 A3 für Produktinformationen im Onlineshop
- C1 I3 A3 für Börsendaten (Live-Ticker)

Die Umsetzung der Maßnahmen erfolgt über die entsprechenden Werkzeuge, siehe Kapitel *Sicherheitstechnologien* (3).

Nach der initialen Umsetzung und auch im Betrieb ist es sinnvoll zu prüfen, ob nicht im Laufe der Zeit sicherheitsrelevante Einstellungen geändert oder personenbezogene Daten übersehen wurden. Im Idealfall hat das Unternehmen eigene Assessments, die automatisiert gegen die Datenbank laufen können (Schritt 4a). Falls keine eigenen automatisierten Assessments etabliert sind oder eine Ergänzung im Rahmen von Oracle's Best Practice oder umgesetzten Frameworks (z.B. STIG, PCI-DSS) gewünscht ist, können die entsprechenden Oracle Tools genutzt werden (Schritt 4b). Die Informationen hieraus helfen, einen Abgleich zwischen der Realität, dem aktuellen Sicherheitszustand und den gesetzlichen bzw. branchenspezifischen Anforderungen durchzuführen. Oft ist die Lücke zwischen der „gefühlten" Sicherheit und der geforderten Sicherheit signifikant größer als vermutet.

Die hieraus ermittelten Schwachstellen werden typischerweise anhand ihrer Kritikalität priorisiert. Scheinbar aufgedeckte Probleme

können im Umfeld des Unternehmens auch einfach nicht relevant sein. Die ggf. notwendige Abarbeitung bzw. Korrektur erfolgt entweder manuell oder automatisiert. In beiden Fällen ist es sinnvoll, dieses Vorgehen zu dokumentieren, da der Nachweis im Rahmen von Regulatorien oder spätestens im Problemfall gefordert wird. Es hat sich als vorteilhaft erwiesen, mit den technisch einfachen Maßnahmen zu beginnen und sich Zwischenziele zu setzen. Es könnte ebenfalls sinnvoll sein, Maßnahmen, die mit wenig Aufwand einen großen Sicherheitsmehrwert liefern, zuerst umzusetzen.

2.2 VERANTWORTLICHKEITEN

Setzen Sie eine Datenbank auf oder betreiben diese, so sind Sie, außer Sie handeln im Auftrag, auch für die Daten bzw. deren Schutz verantwortlich. Es ist ratsam, sich zu vergewissern, dass die Klassifikation und Ableitung von Maßnahmen stattgefunden hat. Vermutlich muss ein Teil der Maßnahmen, wie das Patching oder Überwachen der Auditlogs, auch im Betrieb von Ihnen übernommen werden. Verantwortung kann nicht vollständig delegiert werden, jedoch sollte einem impliziten Versuch entgegengewirkt werden, um den Sorgfaltspflichten nachzukommen. Die Anweisung des Anwendungsowners sollte dokumentiert werden. Im Zweifelsfall bieten auch die Landesdatenschutzstellen beim Umgang mit personenbezogenen Daten Hilfe an.

Auch im Falle der Nutzung von (Datenbank) Cloud-Services werden Aufgabenstellungen nicht automatisch gelöst. Erst die Klassifikation der Daten bestimmt, ob die Schutzmechanismen des Cloudproviders, z.B. Verschlüsselung, ausreichend ist. Der Kunde bleibt verantwortlich für die Daten. Das Modell in der Cloud heißt auch „shared responsibilities“. Alle Regularien beinhalten die Forderung, dass Sie sich von den Schutzmechanismen, auch Dritter wie Cloudanbieter, überzeugen müssen.

Regularien sind unterschiedlich detailliert. Einige bleiben bei Aussagen wie state-of-the-art Security oder der Forderung, dass Daten nur im Rahmen des freigegebenen Zwecks verwendet werden dürfen. Wie dies umgesetzt wird, z.B. über technische User, in Applikation oder mit Handlungsanweisungen, ist nicht ausgeführt. Andere Regularien wie PCI-DSS sind wesentlich detaillierter und fordern explizit eine Verschlüsselung. Besitzt der Cloud Provider, wie z.B. Oracle mit PCI-DSS, für IaaS eine Attestierung, betrifft das die Cloud Plattform an sich. Services, die der Kunde aufsetzt, sind damit nicht automatisch konform zu einem Regularium, sondern benötigen eine entsprechende Attestierung. Nur im Rahmen der SaaS Services ist die Anwendungsschicht inkludiert, der Kunde ist dann „lediglich“ für die Nutzungserlaubnis der Daten gemäß der erteilten der Zweckbindung verantwortlich.

2.3 SECURITY QUICK START FÜR DEN PRAKTIKER

Womit beginnen? Ein praktikabler Ansatz könnte die exemplarische Überprüfung einiger Datenbanken hinsichtlich der „gefühlten" Sicherheit sein. Typischerweise ist die Klassifikation der Daten vorhanden. So kann man sich aus jeder Klasse (z.B. vertraulich, personenbezogen, offen) ein System vornehmen, das kostenfreie non-invasive DBSAT (siehe Kapitel *DBSAT Tool* (3.1.1)) verwenden und die erwarteten Ergebnisse mit der Realität vergleichen. Ein Vorteil ist, dass keine Daten ausgelesen werden, so dass der Datenschutzaspekt bezüglich der gespeicherten Daten gewährleistet ist. Die Ergebnisse können dann bei Handlungsbedarf die entsprechenden Abhilfevorschläge geben.

Im Falle von neu aufzusetzenden Datenbanken sollten die Aspekte Basishärtung und Klassifikation im ersten Schritt bedacht werden. Wie die jüngsten Vorfälle belegen, gehen viele Angriffe auf andere Systeme von ungeschützten Systemen aus. Bei der Klassifikation von schützenswerten Daten kommen typischerweise folgende Technologien zum Einsatz: Verschlüsselung, verschlüsselter Zugriff, berechtigter Zugriff, geregelter Audit/Zugriffsnachweis und Aufrechterhalten der Security.

3 Sicherheitstechnologien

Es stehen eine Reihe von Technologien und Funktionen zur Absicherung der Oracle Datenbank bereit. Einige davon sind Mechanismen bzw. Optionen der Datenbank, andere sind zusätzliche Werkzeuge. Im Schaubild finden Sie beide Arten der Technologien, die in jeweils drei Bereiche unterteilt sind:

Überprüfen/Assess	*Schützen/Prevent*	*Monitoren/Detect*
Security Assessment	DBA & Operation Controls	Database Auditing
Sensitive Data Discovery	Data Encryption & Key Management	Database / SQL Firewall
Privilege Analysis	Data Masking & Subsetting	Centralized Sec. Monitoring

Abb. 2: Sicherheitstechnologien von und für Oracle Datenbanken

In der **ersten Säule** sind die Werkzeuge zur Überprüfung des Status Quo zu finden. Dies umfasst die sicherheitsrelevante Konfiguration der Datenbank, das Benutzermanagement und die Erkennung von sensitiven Daten. Die Privilege Analysis ermöglicht die Analyse tatsächlich genutzter Rechte, um gegebenenfalls das Rechte- und Rollenkonzept anzupassen. Ein Teil der Werkzeuge findet sich

auch in der dritten Säule wieder. Diese können periodisch gestartet werden, um ein kontinuierliches Monitoring zu ermöglichen.

Die **zweite Säule** beschreibt Mechanismen bzw. Werkzeuge zum Schutz der Datenbank. Gestartet wird bei Authentifizierung und dezentralem oder zentralem Usermanagement. Danach wird die Umsetzung einer Aufgabentrennung (SoD – Separation of Duty), gefolgt von der Verschlüsselung am Speicherort und beim Zugriff auf die Daten, beschrieben. Ein weiterer Abschnitt beschreibt Anonymisierung bzw. Pseudonymisierung von Daten, die beispielsweise notwendig wird, wenn Produktionsdaten in Test und Entwicklung verwendet werden sollen oder Anwendungen zu viele Daten anzeigen.

Auch das Thema Patching fällt unter präventiven Schutz.

Hochverfügbarkeit und Backup/Recovery sind ebenfalls Bestandteile von Sicherheitskonzepten. Diese sind jedoch hier in diesem Dokument nicht weiter ausgeführt.

Innerhalb der **dritten Säule** finden sich die Beschreibungen zu Monitoring, Audit, Threat Analyse und IT Compliance. Beginnend mit Überwachungsmöglichkeiten in der Datenbank und vor der Datenbank (DB Firewall) wird auch die Konfiguration überwacht (IT Compliance). Ein Securitymonitoring (auch SIEM/SOC) wertet die Aktivitäten hinsichtlich Policy Verletzungen, Anomalien und bekannte und unbekannte

Threats aus. Entsprechende Aktionen zur Korrektur oder zur Abwehr können hinterlegt werden (Remediation).

3.1 ÜBERPRÜFEN/ASSESS

Beim Überprüfen helfen verschiedene Werkzeuge von Oracle. Diese können einmalig eingesetzt werden oder auch periodisch um den Zustand zu monitoren. Die Werkzeuge enthalten je nach Typ „Best Practices“ oder Regelwerke basierend auf international gültigen Empfehlungen wie STIG. Eines der Werkzeuge, **DBSAT**, ist über gültige Oracle DB Supportverträge kostenfrei.

Das Ergebnis der Assessments ist dabei eine priorisierte Liste, bestehend aus Empfehlungen zur Sicherheitskonfiguration einschließlich Informationen über Benutzerkontokonfigurationen und möglicher sensitiver Daten. Bei sensitiven Daten würden dann die entsprechenden Schutzmaßnahmen, wie Schutz am Speicherort oder Auditierung des Zugriffs, geprüft werden.

Ein weiteres Werkzeug, das Bestandteil von **Database Vault** ist, prüft die verwendeten Privilegien, um das bestehende Berechtigungssystem feiner justieren zu können. Auch eine Simulation des neuen Modells ist möglich.

Auch können Patchstände und Vorgaben mit Hilfe des **Lifecycle Management** im Enterprise Manager oder dem Cloud **Configuration & Compliance Services** verglichen werden.

3.1.1 DBSAT TOOL: KONFIGURATION, USERMANAGEMENT, SENSITIVE DATEN (DICTIONARY)

Nicht jede Datenbank muss gleichermaßen geschützt werden. Bevor diverse Sicherheitsmaßnahmen diskutiert und umgesetzt werden, ist es ratsam, sich ein Überblick über den aktuellen Sicherheitszustand der Datenbank zu verschaffen. Dazu gehört die Überprüfung der bereits vorhandenen Sicherheitsmaßnahmen und Hinweise darauf, welche weiteren Sicherheitsmaßnahmen, entsprechend der Datenklassifikation, potentiell sinnvoll und notwendig wären. Das Oracle Database Security Assessment Tool (DBSAT) überprüft Datenbankkonfigurationen, Datenschutzfunktionalitäten und gibt Sicherheitsempfehlungen gemäß Oracle Datenbanksicherheit Best Practices. Zudem ermöglicht es, auf einfachste Weise, sensitive Daten in Oracle Datenbanken zu suchen. Die hiermit aufgezeigten potenziellen Angriffsziele und Sicherheitsrisiken werden dokumentiert und können bei Bedarf und Notwendigkeit vom Datenbankadministrator behoben werden.

DBSAT ist für Kunden mit gültigem Datenbanksupport kostenfrei und dient in erster Linie der kurzfristigen Identifizierung von Angriffszielen und hilft bei der Minimierung allgemeiner Risiken sowie bei der Umsetzung einer umfassenden Sicherheitsstrategie. Es ist sowohl für On-Premises als auch cloud-basierten Oracle Datenbanken einsetzbar. Für die Datenbank ist es nicht invasiv, d.h. es werden keine

Objekte oder Strukturen in der Datenbank angelegt. DBSAT führt Tests und Scans in folgenden Kategorien durch:

- Benutzerkonten, Privilegien und Rollen, Passwortrichtlinien
- Zugriffskontrolle und Berechtigungskontrolle
- Datenverschlüsselung
- Datenbank und Listener-Konfiguration
- Dateiberechtigungen auf dem Betriebssystem
- Sensitive Data Discovery (Data-Dictionary)

Im Wesentlichen besteht das Werkzeug aus drei Komponenten:

- DBSAT Discoverer (Aufspüren sensitiver Daten)
- DBSAT Collector (Sammeln von Konfigurations-Informationen)
- DBSAT Reporter (Erstellung des DBSAT Berichts)

Der DBSAT Discoverer lässt sich unabhängig vom DBSAT Collector und DBSAT Reporter verwenden.

Abb. 3: Database Security Assessment Tool

Folgende Schritte sind bei Verwendung des Oracle Database Security Assessment Tools notwendig:

- Herunterladen des Oracle Database Security Assessment Tools von Oracle Support (Document ID: 2138254.1)
- Schaffen der Ablaufumgebung und Zuweisen der Privilegien
- Sammeln der Informationen, indem Sie `dbsat collect` bzw. `dbsat discover` auf dem Ziel ausführen
- Ausführung des `dbsat report` Kommandos, um die Berichte zu erstellen (dies wird nur im Zusammenhang mit dem DBSAT Collector benötigt.)

Die aktuelle Vorgehensweise zur technischen Ausführung des Tools finden Sie in der Dokumentation: Security Assessment Tool User Guide Release 2.0.2.

Überprüfen Sie die „gefühlte" Sicherheit mit durch DBSAT erzeugte Reports. Das Tool ist kostenfrei, non-invasiv und liest keine gespeicherten Daten.

3.1.2 ENTERPRISE MANAGER: SENSITIVE DATENANALYSE (ADM) UND IT COMPLIANCE

Mit dem Oracle Enterprise Manager steht ein Werkzeug zur Verfügung, dass auch dazu genutzt werden kann, sich einen Überblick zu verschaffen. Neben dem Management von Umgebungen können folgende Funktionen bzw. sogenannte „Packs“ genutzt werden:

1. Application Data Modelling aus dem Data Masking und Subsetting Pack, um sensitive Daten aufspüren zu können
2. Lifecycle Pack zur Auflistung verfügbarer nicht eingespielte Recommended Patches und der Prüfung der Compliance nach eigenen Vorgaben oder Industriestandards (z.B. STIG, PCI)

Der zweite Punkt ist im Kapitel *Monitoring* (3.3) beschrieben, da dies (Patchen, IT Compliance) typischerweise kontinuierliche Aufgaben sind. Der erste Punkt zum Aufspüren sensitiver Daten über Application Data Modelling (kurz ADM) wird hier kurz skizziert. DBSAT, das kostenfreie Werkzeug aus dem vorhergehenden Abschnitt operiert auf Data Dictionary Ebene und kann aktuell keine Daten wie ADM analysieren.

Das Aufspüren von sensitiven Daten in einem komplexen Datenmodell stellt in verschiedenen Situationen eine große Herausforderung dar. Dabei wird zunächst ein „Application Data Model“ mit dem Enterprise Manager erstellt, in dem Schemata und

Tabellen hinzugefügt werden. Als nächstes werden die Beziehungen der Spalten untereinander festgelegt. Auch hier hilft Enterprise Manager, indem bereits vorhandene Fremdschlüssel-Constraints automatisch berücksichtigt werden. Darüber hinaus können aber auch weitere Abhängigkeiten eingetragen werden, was zum Beispiel für alle Datenmodelle wichtig ist, welche nur sehr sparsam mit Fremdschlüsseln arbeiten. Ist das Datenmodell fertig, kann Enterprise Manager den Datenbestand nach sensitiven Daten durchsuchen. Dieses geschieht unter Berücksichtigung von Spaltennamen und Datenmustern. Dazu gibt es diverse Vorlagen zum Beispiel für IP-Adressen, Kreditkartennummern, ISBN-Nummern oder Email-Adressen, die Enterprise Manager vorgefertigt nutzt. Diese Vorlagen können beliebig erweitert werden.

ADM kann nur im Rahmen einer Lizenz genutzt werden, entweder via Data Masking und Subsetting Pack oder als „restricted" Lizenz im Rahmen von Audit Vault and Database Firewall, Database Vault, Advanced Security und Label Security. Diese Information stammt von Oktober 2018, siehe auch Database Licensing Guide und Audit Vault Licensing Guide.

Bevor Sie ADM „sensitive data search" nutzen, klären Sie bitte, ob Sie die Daten aus Datenschutzgründen scannen dürfen.

3.1.3 ORACLE MANAGEMENT CLOUD: IT COMPLIANCE

Oracle Management Cloud (OMC) ist eine cloudbasierte IT Management Lösung zum Monitoren heterogener Systeme in der Cloud als auch On-Premises. Dadurch können Probleme schneller identifiziert werden und das Einhalten von IT Compliance Richtlinien überwacht werden. Durch die Nutzung von Machine Learning Algorithmen (Clustering, Korrelationen, Anamoly Detection, Time Series Analysis, Seasonality Analysis) erhalten Sie tiefere Hintergrundinformationen und Analysen. Als Cloud-Service brauchen Sie keine weiteren separaten Management Lösungen zu installieren, der Service und die Wartung der Lösung im eigenen Rechenzentrum entfällt. Dies spart Zeit und Kosten.

Die OMC ist in der Lage, Oracle Datenbanken im eigenen Rechenzentrum, in der Oracle Cloud, bei Amazon Webservice oder bei Microsoft Azure zu überwachen und die Konfigurationen hinsichtlich vorgenommener Änderungen im Blick zu behalten.

Zur Prüfung von IT Compliance Vorgaben nutzt OMC Configuration & Compliance vorhandene Benchmarks (STIG, NIST, PCI-DSS, CIS und SCAP) und stellt diese als Regelwerke zur Verfügung, ähnlich dem Enterprise Manager. Das Hinterlegen von eigenen Regeln wird unterstützt. Eine weitere Erläuterung dieses OMC Services erfolgt im Kapitel *Oracle Management Cloud: IT Compliance* (3.1.3), da dieser neben der einmaligen Analysemöglichkeit auch für fortwährendes Monitoring eingesetzt werden kann.

Zur Auswertung und zum Ausführen der Regelwerke (Assessments) wird ein Agent oder Zugriff auf ein REST API benötigt. Assessments sind als Skripte implementiert.

In der nachfolgenden Abbildung ist die Architektur von OMC dargestellt:

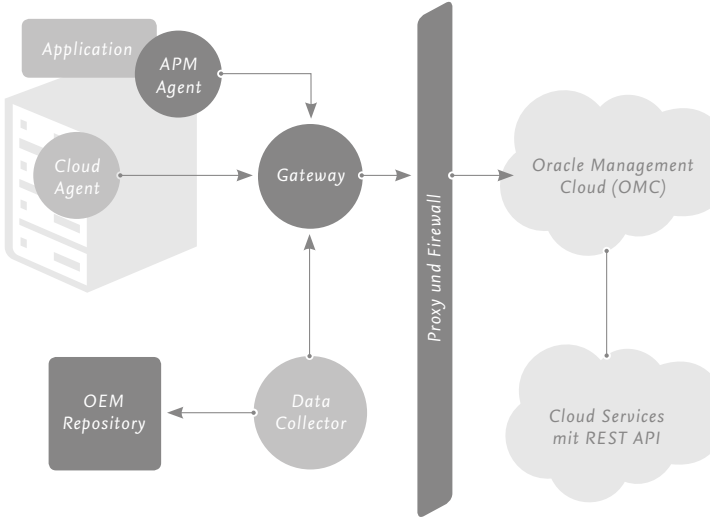

Abb. 4: OMC Architektur

Mit Hilfe der OMC Agenten (Cloud Agent, APM Agent und Data Collector) oder REST API werden Daten, Metriken, Log Dateien und Assessment Ergebnisse der zu überwachenden Systeme gesammelt und an die OMC für die weitere Analyse und Compliance Prüfung übertragen. Ein Gateway kümmert sich um die Konsolidierung der gesammelten Daten der OMC Agenten, speichert diese zwischen und leitet diese gebündelt an die OMC Plattform weiter.

In der OMC Plattform werden die gesammelten Daten in einer BigData Appliance weiter verarbeitet und in ein einheitliches Datenmodel (unified data model) für weitere Analysen abgelegt. Jegliche Auswertungen greifen auf dieses Datenmodell zu. Eine automatische Auswertung der Log Files erfolgt im Hintergrund und ist direkt im IT Compliance Dashboard verfügbar.

Compliance Ergebnisse (Compliance Score Wert, Violations, Alerts und Fehler) der Assessments werden in der OMC Oberfläche dargestellt und in Log Files protokolliert. Diese bilden die Basis für die Abschätzung der Einhaltung der Vorgaben und sind der Startpunkt für weitere Aktionen. OMC bietet im Vergleich zum Enterprise Manager die Möglichkeit Workflows zu nutzen, die Gegenmaßnahmen, sogenannte Auto-Remediations, automatisch einzuleiten und eine erneute Compliance Prüfung starten zu können.

In den Details der Compliance Prüfungsergebnisse findet der Nutzer/Auditor eine detaillierte Erklärung der Regel, Status der Regelprüfung inklusive Regelverletzungen und Hinweise zu Fehlermeldungen und Gegenmaßnahmen (remediation).

Beispielsweise wurde in einer Compliance Prüfung für eine Oracle Datenbank herausgefunden, dass das DB Auditing deaktiviert ist. Die Prüfung der Regel „Enable Database Auditing" weist den Auditor darauf hin, dass der `AUDIT_TRAIL` Parameter nicht gesetzt wurde und unterbreitet den Vorschlag (Finding) diesen zu setzen, damit das DB Auditing einwandfrei funktioniert.

Auf diese Weise erhält der Auditor einen verständlichen Bericht der Prüfung und einen besseren Überblick auf die IT Infrastruktur und Anwendungslandschaft ohne tiefere Technikkenntnis zu besitzen.

3.1.4 ORACLE DATABASE VAULT: GENUTZTE BERECHTIGUNGEN

Die durch das Database Assessment Tool aufgezeigten und den Benutzern zugeordneten Rollen und Privilegien sollten hinsichtlich ihrer Notwendigkeit überprüft werden. Oberstes Ziel ist, nur die Rollen und Privilegien zu vergeben, welche für die fehlerfreie Ausführung der entsprechenden Funktionalität notwendig sind. Die Tatsache, dass die meisten

Applikationen über wesentlich mehr Rollen und Privilegien verfügen als notwendig sind, stammt oft daher, dass Applikationsentwickler lieber ein wenig mehr oder höhere Privilegien besitzen. Sogar die Vergabe der DBA-Rolle ist hier, nur um keine Zugriffsprobleme bei der Entwicklung zu bekommen, keine Seltenheit. Nach Beendigung der Entwicklung wird diese Rolle aus verschiedensten Gründen oft nicht wieder entzogen. Die hieraus resultierenden Probleme entstehen erst beim Betrieb der Applikation beziehungsweise der Datenbank, wenn eine Sicherheitsüberprüfung der Applikation beziehungsweise des Verfahrens ansteht. Jetzt wird gefragt: *„Sind diese Rollen und Privilegien wirklich notwendig?“* und *„Welche dieser Rollen und Privilegien können dem Benutzer entzogen werden?“*. In den meisten Fällen können diese Fragen nicht beantwortet werden. Das Risiko ist einfach zu hoch, dass die Applikation, nach dem Entzug von vermeintlich nicht benötigten Rollen und Privilegien, nicht mehr fehlerfrei funktioniert.

Unterstützung bei dieser Herausforderung liefert eine Funktionalität von Oracle Database Vault ab der Datenbank-Version 12c. Die sogenannte „Privilege Analysis“ ermöglicht das Protokollieren der vom Benutzer wirklich verwendeten Privilegien beim Zugriff auf Datenbank-Objekte, unabhängig davon, welche Rollen und Privilegien ihm zugeordnet sind. Die Protokollierung erfolgt mittels eines sogenannten Capture-Prozesses über einen bestimmten Zeitabschnitt. Dieser Prozess wird durch das PL/

SQL-Package - DBMS_PRIVILEGE_CAPTURE – initialisiert. Die Aufzeichnung der verwendeten Privilegien durch den Privilege-Capture Prozess lässt sich auf Basis von Session Context Informationen einschränken. So ist es möglich, nur bestimmte Applikationen oder Benutzer zu überprüfen. Nach Beendigung des Capture Prozesses lassen sich die Ergebnisse mittels der Tabellen DBA_USED_PRIVS, DBA_USED_OBJPRIVS, DBA_UNUSED_PRIVS und DBA_UNUSED_OBJPRIVS auswerten. Die hieraus resultierenden Ergebnisse ermöglichen das Erstellen neuer Rollen mit ausschließlich benötigten Privilegien.

Überprüfungen dazu können auch mittels dem Database Security Assessment Tool unter „Authorization Control“ erfolgen.

3.2 SCHÜTZEN/PREVENT

Viele unerwünschte Zugriffe auf eine Datenbank erfolgen über die DB Benutzer. Daher ist die Konzeption und Konfiguration von Benutzern und ihren Berechtigungen wichtig. Die korrekte Verwendung von Berechtigungen und Rollen zur Einschränkung des Benutzerzugriffs in Verbindung mit einer starken Authentifizierung legt die Grundlage für eine sichere Datenbank.

Eine weitere ergänzende Schutzmaßnahme ist die Härtung des Systems. Dies betrifft sowohl die Datenbank selbst, bei der nicht benötigte Services nicht installiert werden, als auch das darunter-unterliegende Betriebssystem und Netzwerk.

Soll ein stärkerer Schutz der Daten umgesetzt oder auch sichergestellt werden, beispielsweise dass Administratoren nicht mehr auf Daten zugreifen können, kann eine stärkere Trennung in der Datenbank bis hin zu SoD (**Segregation of Duties/Gewaltentrennung**) umgesetzt werden.

Zum Schutz von Daten kommt typischerweise eine **Verschlüsselung** zum Ansatz. Dabei soll sowohl eine Transportverschlüsselung als auch die Verschlüsselung am Speicherort stattfinden. Mit einer Verschlüsselung kommt das Schlüsselmanagement mit Anforderungen wie Key Rotation und Zentralisierung zum Tragen. Lösungen für alle drei Anforderungen sind in weiteren Kapiteln beschrieben.

Werden Daten in Testsystemen oder für Auswertungen benötigt, ist oft ein **Anonymisieren** erforderlich. Hierbei werden entweder die Daten am Speicherort (in der Test DB) verändert oder dynamisch während des Zugriffs.

Natürlich ist das Einspielen von **Patches** für das Aufrechterhalten der Security erforderlich.

3.2.1 AUTHENTIFIZIERUNG

Authentifizierung bedeutet, dass die Identität eines Benutzers überprüft wird, bevor Daten, Ressourcen oder Anwendungen verwendet werden können. Die Oracle Datenbank bietet für die Benutzerauthentifizierung im Wesentlichen vier Kategorien an. Diese werden durch die „`IDENTIFIED BY`“ Klausel im `CREATE/ALTER USER` Befehl bestimmt.

```
CREATE USER psmith IDENTIFIED BY password;
```

Die klassische Art sich an einer Oracle Datenbank zu Authentifizieren ist die Verwendung eines **Passworts**. Ein Passwort gilt aktuell als sicher, wenn es gemäß des Security Technical Implementation Guides (STIG) folgende Eigenschaften aufweist:

- Das Passwort hat mindestens 15 Zeichen.
- Das Passwort hat mindestens 1 Kleinbuchstaben und mindestens 1 Großbuchstaben.
- Das Passwort hat mindestens 1 Ziffer.
- Das Passwort hat mindestens 1 Sonderzeichen.
- Das Passwort unterscheidet sich vom vorherigen Passwort um mindestens 8 Zeichen.

Unter zur Hilfenahme von Passwort-Profiles lässt sich die Einhaltung von Passwort-Richtlinen erzwingen. Seit der

Datenbankversion 12c stellt Oracle beispielsweise ein entsprechendes Profile - `ORA_STIG_PROFILE`- gemäß Security Technical Implementation Guides (STIG) zur Verfügung.

Das Passwort selbst wird bei Verwendung der 12c-Passwortversion innerhalb der Oracle Datenbank durch einen Password-Based Key Derivation Function 2 (PBKDF2) basierenden SHA512 Hashing Algorithmus geschützt. Darüber hinaus fügt die 12c-Passwortversion dem Passwort beim Hashing-Vorgang einen Salt hinzu, um den Schutz zu erhöhen. Aus diesem Grund ist es dringend empfohlen, die aktuellste Oracle Client Software einzusetzen, da nur diese über diese Sicherheitseigenschaften verfügt. Über die SQLNET Parameter `SQLNET.ALLOWED_LOGON_VERSION_SERVER` und `SQLNET.ALLOWED_LOGON_VERSION_CLIENT` lässt sich die Verwendung eines entsprechend aktuellen Oracle Clients erzwingen. So erzwingt der Parameter `SQLNET.ALLOWED_LOGON_VERSION_SERVER = 12a` die Verwendung der 12c-Passwortversion.

Die aktuelle Einstellung dieses Wertes wird auch durch das Oracle Database Assessment Tool ermittelt und eine Empfehlung ausgegeben.

Die Übertragung des Passworts beim Anmeldevorgang eines Datenbank-Clients erfolgt dabei immer verschlüsselt.

```
CREATE USER ops$psmith IDENTIFIED EXTERNALLY;
```

Die Klausel `IDENTIFIED EXTERNALLY` ermöglicht eine **Authentifizierung über das Betriebssystem** und damit die Möglichkeit, ein

Single Sign On (SSO) für Datenbank-Verbindungen umzusetzen. Hierunter fällt die nicht mehr empfohlene Möglichkeit, den Betriebssystem-Benutzer selber zu verwenden. Falls es doch notwendig sein sollte diese Art der Authentifizierung zu verwenden, sollte dies durch das Setzen des Datenbank-Initialisierungs-Parameters `OS_AUTHENT_PREFIX` erfolgen. Klassischerweise wurde hierfür das Präfix `OPS$` verwendet.

In Zeiten von Oracle Virtual Box und ähnlichen Virtualisierungsmöglichkeiten ist diese Variante der Benutzerauthentifizierung als hohes Risiko zu bewerten. Damit lässt sich leicht innerhalb des Unternehmens ein virtuelles Betriebssystem mit entsprechendem Benutzer, zum Beispiel „psmith“, erstellen, um sich damit dann passwortfrei an der Datenbank anzumelden.

Die wesentlich bessere und vor allem sichere Variante, sich extern authentifizieren zu lassen, ist die **Authentifizierung über Kerberos** beziehungsweise Zertifikate (PKI). Diese Varianten setzen allerdings ein vorhandenes Kerberos-System beziehungsweise eine PKI voraus.

Angelegt werden die Datenbank-Benutzer bei einer Kerberos Authentifizierung durch:

```
CREATE USER psmith IDENTIFIED EXTERNALLY AS 'PrincipleName@Realm';
```

Hierbei entspricht der ‚PrincipleName' der Identität des Kerberos-Ticket Benutzers.

Soll der Benutzer über ein **Zertifikat** authentifiziert werden, muss er entsprechend seines im Zertifikat verwendeten Common-Names angelegt werden.

```
CREATE USER psmith IDENTIFIED EXTERNALLY AS 'cn=psmith,ou=sales,…';
```

Diese Variante eignet sich gut zur Verwendung bei Applikations-Servern, um zum Beispiel passwortfrei (SSO) die JDBC Connection Pools zu authentifizieren.

```
CREATE USER global_user IDENTIFIED GLOBALLY
```

Die `IDENTIFIED GLOBALLY` Klausel wird verwendet, wenn der Datenbank-Benutzer über einen **zentralen LDAP** Server authentifiziert werden soll. Der im `CREATE USER` Befehl angegebene Benutzer ist im Prinzip ein Proxy Benutzer ohne weiterführende Datenbank Privilegien und ohne Passwort. Passwort und Passwort-Richtlinie befinden sich im angeschlossenen LDAP-Server. Direkt unterstützte LDAP Server sind das Oracle Internet Directory (OID), Oracle Unified Directory (OUD) inklusive Proxyfunktion zu anderen LDAP. Die eigentlichen Authentifizierungsmethoden (Passwort, Kerberos und Zertifikate) bleiben davon unberührt. Im Wesentlichen geht es darum, Datenbank-Benutzer und Rollenvergabe zu zentralisieren. Die Datenbank Funktionalität hierfür heißt Enterprise User Security.

Neu mit der Datenbank Version 18c ist die direkte Integration mit einem Microsoft Active Directory Server durch das sogenannte ‚Centrally Managed User'. Bei Verwendung der passwortbasierten Authentifizierung ist eine Schema-Erweiterung im Microsoft Active Directory Server notwendig ebenso wie die Installation der sogenannten Passwort-Filter DLL auf dem Active Directory Server. Der Passwort-Filter hat die Aufgabe, das Benutzerpasswort in einem von Oracle verwendbaren Hash-Algorithmus zu verschlüsseln. Der Passwort-Filter und die Schemaerweiterung ist bei einer Benutzer-Authentifikation mittels Kerberos oder Zertifikat nicht notwendig.

```
CREATE USER psmith NO AUTHENTICATION;
```

Ab der Datenbank Version 18c ist es möglich, Benutzer **ohne eine Authentifizierungsmethode** anzulegen. Also ohne eine `IDENTIFIED` Klausel. Dies ist bei Anwendungsfällen sinnvoll, bei denen es nur darauf ankommt, Datenbank-Objekte abzulegen. Auch die Verwendung solcher Benutzer als Proxy Benutzer hat sich als sehr praktikabel erwiesen. Oft war das zwanghafte Angeben eines Passworts lästig.

Diese Variante ist gut geeignet beim Gebrauch von Proxy Benutzern. Die Verwendung der `GRANT CONNECT THROUGH` Klausel des `ALTER USER` Befehls ermöglicht einem anderen Datenbank Benutzer die Rechte und Rollen des Proxy Benutzers zu verwenden.

```
SQL> CREATE USER mike0815 IDENTIFIED BY welcome1;

User created.

SQL> ALTER USER psmith GRANT CONNECT THROUGH mike0815;

User altered.

SQL> CONN mike0815[psmith]/welcome1@pdb1_cdb3
Connected.
SQL> SHOW USER
USER is "PSMITH"
```

Eine Überprüfung ist ebenfalls durch das Database Security Assessment Tool unter „User Accounts“ möglich.

3.2.2 AUTORISIERUNG

Die Autorisierung, also die Zuweisung von Rechten, ermöglicht berechtigten Benutzern den Zugriff, die Verarbeitung oder das Ändern von und auf Daten. Berechtigungen, auch Privilegien genannt, können auf Objekte wie Programme, Funktionen, Schemas, ganze Tabellen oder sogar auf Tabellenzeilen vergeben werden. Einzelne Objekt- und Systemprivilegien werden entsprechend den Anforderungen in Datenbank-Rollen organisiert und einem berechtigten Benutzer zugeordnet. Dabei sollte strikt nach dem Prinzip „Least Privilege“ vorgegangen werden. Eine Überprüfung, ob dieses Prinzip eingehalten wird, lässt sich mittels Oracle Privilege Analysis durchführen, siehe Kapitel *Oracle Database Vault: Genutzte Berechtigungen* (3.1.4).

Im Standard werden Rollen innerhalb der Datenbank dem Benutzer zugewiesen oder entzogen. Dieses Vorgehen ist bei einer kleinen Anzahl von Datenbanken (< 10) auch gut durchführbar. Bei einer größeren Anzahl von Datenbanken ist es ratsam, sowohl die Datenbank-Benutzer, als auch die Berechtigungsvergabe auszulagern beziehungsweise zu zentralisieren. In Kapitel *Authentifizierung* (3.2.1) ist die Möglichkeit beschrieben, Datenbank-Benutzer in ein LDAP-System auszulagern und dort den Benutzern entsprechende Datenbank-Rollen zuzuweisen oder zu entziehen. Das Datenbank Enterprise Edition Feature „Enterprise User Security" ermöglicht die Zentralisierung von Datenbank-Benutzern und Datenbank-Rollen.

Mit dem Einsatz der Enterprise User Security werden folgende Themen adressiert:

- Zentrale Provisionierung und De-Provisionierung von Datenbankbenutzern
- Zentrale Passwortverwaltung und Self Services, wie Passwort-Resets
- Zentrale Berechtigungsverwaltung durch den Einsatz globaler Datenbankrollen
- Stets aktuelle Berichtserstellung bei Verwendung eines Identity Management Systems

Die Voraussetzung hierfür ist die Verwendung eines von der Oracle Datenbank zertifizierten LDAP Servers. Unterstützte LDAP Server sind das Oracle Internet Directory (OID), Oracle Unified Directory (OUD) inklusive Proxyfunktion zu anderen LDAP und ab der Oracle Datenbank 18c auch das Microsoft Active Directory.

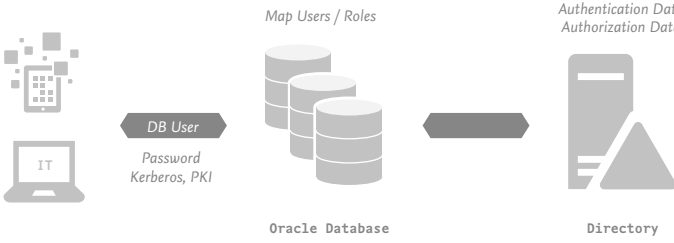

Abb. 5: Prinzipielle Architektur: Centrally Managed User und Enterprise User Security

Auch über die Steuerung des **SYSDBAs** sollte nachgedacht werden. Lange Zeit war die Verwendung der SYSDBA Rolle für viele privilegierte Datenbank-Operationen und Funktionalitäten zwingend notwendig. Eine strikte Trennung der Rollen gemäß „principle of least privilege“ wurde zwar auch für administrative Tätigkeiten gefordert, konnte aber bis zur Datenbank Version 12.1 für bestimmte Operationen und Funktionalitäten nicht umgesetzt werden.

Ab der Datenbank Version 12.1 ist es möglich, Funktionstrennungen für das Key-Management (SYSKM), für das Backup durch den RMAN (SYSBACKUP), für Data Guard Operationen (SYSDG) und für den Betrieb eines Real Application Cluster (SYSRAC) umzusetzen.

Eine ausführliche Beschreibung welche Privilegien hinter den SYS-Rollen stecken, finden Sie in der offiziellen Dokumentation im Security Guide 12c im Kapitel *Configuring Privilege and Role Authorization*.

Diese neuen Rollen stellen die Möglichkeit einer Funktionstrennung zur Verfügung. SYSDBA bleibt allerdings weiterhin der mächtigste Datenbankbenutzer. Es sei denn, Database Vault wird eingesetzt, siehe Kapitel *Zugriffsmanagement: Gewaltentrennung / SoD* (3.2.4).

Prüfen Sie ob diese feingranulareren Möglichkeiten in Ihrem Hause einen Mehrwert liefern können.

Normalerweise werden Rollen durch einen `GRANT` Befehl statisch einem Benutzer zugewiesen.

```
GRANT rolle_abc TO user_a;
```

Es besteht jedoch die Möglichkeit, Rollen dynamisch innerhalb einer Datenbanksession zu vergeben. Das

Datenbank-Feature, welches das ermöglicht, heißt **Secure Application Roles (SAR)** und ist Bestandteil der Enterprise Edition. Das Prinzip ist, bei der Erstellung einer Rolle durch den Datenbankbefehl `CREATE ROLE` eine `IDENTIFIED USING` Klausel gefolgt von einer PL/SQL Prozedur (`dba_sar_proc`) anzufügen.

```
CREATE ROLE dba_sar IDENTIFIED USING dba_sar_proc;
```

Die Prozedur `dba_sar_proc` muss zu diesem Zeitpunkt noch nicht existieren. Nun können Privilegien oder andere Rollen der SAR-Rolle `dba_sar`, entsprechend den Anforderungen, zugeordnet werden.

```
GRANT CREATE USER TO dba_sar;
```

Diese Rolle kann jetzt einem Benutzer zugeordnet werden. Zu beachten ist aber, dass diese Rolle nicht aktiv ist.

```
GRANT dba_sar TO SCOTT;
```

Die Aktivierung dieser SAR Rolle kann nur durch die Ausführung der Prozedur `dba_sar_proc` durchgeführt werden.

```
CREATE OR REPLACE PROCEDURE dba_sar_proc AUTHID CURRENT_USER AS
BEGIN  dbms_session.set_role('DBA_SAR'); END;
```

Als letzten Schritt wird nur noch dem Benutzer SCOTT die Ausführungsrechte für die Prozedur gegeben.

```
GRANT EXECUTE ON dba_sar_proc TO scott;
```

Jetzt ist die Secure Application Role einsatzbereit.

Zur Überprüfung melden Sie sich als Benutzer SCOTT an und führen danach die Prozedur aus.

```
SQL> SELECT * FROM SESSION_ROLES;
ROLE
------------------------------
CONNECT
SQL> EXECUTE SYS.DBA_SAR_PROC;
PL/SQL procedure successfully completed.
SQL> SELECT * FROM SESSION_ROLES;
ROLE
------------------------------
DBA_SAR
```

Diese Variante der Rollenvergabe ist extrem flexibel und hat viele Einsatzgebiete.

Eine Überprüfung ist durch das Database Security Assessment Tool unter „Privileges and Roles“ möglich.

Die existierenden Objektprivilegien, die Anwendern das Lesen, Einfügen, Ändern und Löschen von Daten erlauben, zielen immer auf alle Zeilen einer Tabelle. Soll der Zugriff auf Zeilenebene gesteuert werden, weicht man entweder auf die Steuerung des Zugriffs über Anwendungen aus oder verwendet Views. Beides ist uneffektiv: die Steuerung über Anwendungen muss in jeder Anwendung separat programmiert werden und Änderungen in jeder Anwendung nachgezogen werden. Bei der Steuerung über Views wird bei großen Benutzergruppen oder vielen Views das ganze Berechtigungssystem schnell unüberschaubar. Oracle bietet schon seit der Version Oracle8i eine Lösung für dieses Problem: Die Lösung ist unter den Namen **Fine Grained Access Control (FGAC)** oder auch **Virtual Private Database (VPD)** bekannt. Es handelt sich dabei um ein Feature der Enterprise Edition, dessen Nutzung in der Regel eine enge Zusammenarbeit von Datenbankadministratoren und Anwendungsentwicklern voraussetzt.

VPD implementiert die Kontrolle für den Zugriff auf einzelne Zeilen auf der Ebene der Tabelle: Mit dem Paket `DBMS_RLS` werden die Befehle `INSERT`, `UPDATE`, `DELETE` und `SELECT`, die auf eine mit VPD geschützte Tabelle zugreifen, um eine zusätzliche `WHERE`-Bedingung erweitert. Diese `WHERE`-Bedingung wird flexibel durch eine Funktion erstellt, die der Datenbankadministrator oder Anwendungsentwickler schreiben muß. Enthält das `SELECT`, `UPDATE` oder `DELETE` bereits eine `WHERE`-Bedingung, wird die von

der Funktion erzeugte WHERE-Bedingung einfach mit AND angehängt. Die Verbindung zwischen Funktion und Tabelle wird als Policy bezeichnet, und die Funktion deshalb auch als Policy-Funktion.

Es ist möglich, identische oder auch unterschiedliche Policy-Funktionen für die Aktionen SELECT, INSERT, UPDATE und DELETE anzulegen. Sogar mehrere Policy-Funktionen für ein und dieselbe Aktion sind erlaubt. Und man kann auch Policy-Funktionen in Gruppen zusammenlegen (Oracle-Terminologie PARTITIONED FGAC): die Ergebnisse der Funktionen werden einfach durch AND verbunden.

Die wahre Leistungsfähigkeit der VPD offenbart sich im Zusammenspiel mit Variablen, die als sogenannter Application Context im Arbeitsspeicher angelegt werden. Ein Application Context kann z. B. beim ersten Connect auf die Datenbank im Rahmen eines Logon-Triggers mit Werten aus der Betriebssystem- oder Anwendungsumgebung des Anwenders, aber auch mit Informationen aus der Datenbank, initialisiert werden. Durch den Zugriff auf den Application Context vermeiden Policy-Funktionen dann unnötige Mehrfachausführungen von Programmcode oder aufwendige SQL-Zugriffe. Ein Application Context kann im Bereich des Arbeitsspeichers eines Anwenders angelegt werden. Er kann aber auch in einem Bereich angelegt werden, der für

alle Benutzer verfügbar ist. Dann spricht man von sogenannten Global Application Contexts. Global Application Contexts unterstützen das Connection Pooling, Multitier-Anwendungen oder Anwendungen, die ihre eigene Authentifizierung vornehmen und selbst als Datenbankbenutzer agieren.

Die VPD-Infrastruktur ist für die Anwender nicht sichtbar. Die Policies werden auf alle Benutzer, außer auf den Benutzer SYS, angewendet: SYS hat als einziger das Privileg EXEMPT ACCESS POLICY. Allerdings kann sich ein ‚normaler' Datenbankadministrator das Privileg zuweisen. Soll auf jeden Fall verhindert werden, dass Datenbankadministratoren auf Benutzerdaten zugreifen, geschieht das durch den Einsatz der Datenbankoption Oracle Database Vault.

Datenbankentwickler können mit Oracle Database 18c XE VPD verwenden und testen.

3.2.3 HARDENING

Ist das Betriebssystem unsicher, sind alle darauf betriebenen Anwendungen und Verfahren gefährdet. Ebenso gefährdet sind dann auch Systeme, die sich im gleichen Netzwerk-Segment wie das unsichere Betriebssystem befinden. Jede IT-Komponente sollte im Rahmen des Möglichen gehärtet werden. Dabei ähneln sich die meisten Maßnahmen zur Härtung von IT-Komponenten.

Zum Beispiel sind viele Maßnahmen zur Härtung eines Betriebssystems und einer Datenbank identisch:

Restriktive Rechtevergabe, Deaktivierung und Deinstallation unsicherer Protokolle und Software, regelmäßige Softwareaktualisierung und ein zielgerichtetes Auditing.

Unter dem Härten von IT-Komponenten wird eine Konfiguration verstanden, die ausschließlich die Funktionen zulässt, die für die korrekte Ausführung eines Service, einer Anwendung oder eines Verfahrens benötigt wird. Nicht mehr und nicht weniger. Zusätzlich wird überprüft, ob Standardfunktionen beziehungsweise notwendige Zusatzfunktionen entsprechend vorgeschriebener Sicherheitsrichtlinien richtig eingestellt sind, wie zum Beispiel keine Nutzung von Standardkennwörtern, Zurücknahme von nicht notwendigen Privilegien, Zugriffskontrolle auf sensible Daten, Datenverschlüsselung etc.

Das Unternehmen sollte klare Richtlinien zur Härtung des Betriebssystems definieren und nachhaltig umsetzen und überwachen. Standardisierung von Betriebssystemen oder Datenbanken können wesentlich dazu beitragen, nachhaltig ein hohes Maß an Sicherheit zu gewährleisten.

Hilfestellung beim Härten von Oracle Datenbanken geben auch das Security Assessment Tool, das Oracle Enterprise Manager Compliance Framework und der Oracle Mana-

gement Cloud Configuration & Compliance Service, die neben den Oracle Best Practices auch den Security Technical Implementation Guide (STIG) als Basis zur Überprüfung verwenden.

Ein stets aktualisiertes Dokument zur Sicherung von Oracle Datenbanken finden Sie im Oracle Support unter „Security Checklist: 10 Basic Steps to Make Your Database Secure from Attacks (Doc ID 1545816.1)“ sowie in der aktuellen Datenbank-Dokumentation unter „Security → Database Security Guide“.

Informationen zum Härten von Oracle Linux finden Sie über die Suche nach „Tips for Securing an Oracle Linux Environment“.

Prüfen Sie periodisch, ob in Ihren Hardening Richtlinien die Empfehlungen der Checklisten enthalten sind. Eine Überprüfung ist durch das Database Security Assessment Tool möglich.

3.2.4 ZUGRIFFSMANAGEMENT: GEWALTENTRENNUNG/SOD (DB VAULT)

Neben dem klassischen Zugriffsmanagement der Oracle Datenbank über Privilegien und Rollen (Kapitel *Autorisierung* (3.2.2)), bietet die Oracle Datenbank auch ein spezielles Zugriffsmanagement mit der Datenbank-Option Oracle Database Vault für hochprivilegierte Benutzer an. Mit Database Vault können verschiedene Bereiche der Datenbankadministration auf unterschiedliche Personen beziehungsweise Rollen verteilt werden. Datenbankadministratoren können keine Benutzerdaten mehr lesen oder ändern

(Schutz vor dem sogenannten Innentäter), SQL-Befehle können beliebigen Regeln – z.B. dem 4-Augen-Prinzip – unterworfen und applikationsspezifische Administratoren eingerichtet werden. Database Vault verwendet sogenannte Realms (Schutz-Bereiche) um Objekte zu schützen. Ein Realm ist ein logisches Konstrukt vergleichbar mit einer „Käseglocke". Alle Objekte (wie Schemas, Tabellen, Programme, Views), die sich innerhalb dieses Realms befinden, also unterhalb der „Käseglocke", sind vor dem Zugriff eines Benutzers geschützt, auch wenn mittels System-Privileg, zum Beispiel `SELECT ANY TABLE`, auf das entsprechend geschützte Objekt zugegriffen wird. Ausschließlich Benutzer mit dedizierten Objekt-Privilegien – `SELECT ON TABLE XYZ`- können weiterhin wie gewohnt auf die Objekte zugreifen. Dies ist unabhängig davon, ob die Objekt Privilegien direkt oder indirekt über eine Rolle dem Benutzer zugewiesen wurden. Aus diesem Grund ist es wichtig und sinnvoll, sich vor dem Einsatz von Database Vault einen Überblick über das Rollen und Privilegien-Konzept der Applikation zu verschaffen. Dabei kann das Database Vault Feature Privilege Analysis unterstützen, siehe Kapitel *Oracle Database Vault: Genutzte Berechtigungen (*3.1.4).

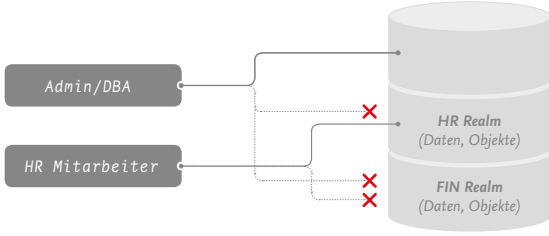

Abb.6: Schutz durch Database Vault

Neben den Realms gibt es noch eine weitere Funktion von Database Vault: die Command-Rules. Mit Command-Rules lässt sich die Ausführung einzelner Datenbank-Befehle steuern. Das bedeutet nicht, dass die Verwendung der Befehle entweder erlaubt oder untersagt ist, sondern die Steuerung erfolgt individuell über sogenannte Rules beziehungsweise Rule-Sets. Eine Rule ist im Prinzip eine Funktion, welche ein TRUE oder FALSE zurückgibt. Oracle liefert im Standard bereits mehrere Rules aus. Sollten diese nicht ausreichen, können weitere Rules auf Basis von Funktionen erstellt werden. Die einzelnen Rules werden dann in Rule-Sets logisch miteinander verbunden. So ist es möglich, sowohl einfache als auch komplexe Regelwerke gemäß organisatorischen Anforderungen zu erstellen. Diese Rule-Sets steuern dann die Ausführung entsprechend sensibler Datenbank-Befehle.

Damit die Gewaltentrennung zwischen technischer, fachlicher und Sicherheits-Administration funktioniert, bringt Database Vault zwei neue Rollen/Benutzer mit, den Database Vault Administrator und den Database Vault Account Manager. Es existieren nun drei privilegierte Rollen, die voneinander strikt getrennt sind, das heißt, sich keine hohen Privilegien teilen. Es gibt den klassischen Datenbankadministrator (DBA), der ein paar seiner Privilegien entzogen bekommen hat, beziehungsweise manche Aktivitäten ohne Hinzunahme eines Dritten, zum Beispiel des Sicherheitsadministrators (Database Vault Administrator), nicht mehr ausführen kann. Dazu gehört auch das Benutzermanagement, welches nun durch eine separate Rolle, dem Database Vault Account Manager, durchgeführt wird.

Damit wird deutlich, dass die Einführung von Database Vault neben den technischen Implikationen, auch Auswirkungen auf die Organisation hat.

Hilfreich ist die Erstellung eines Service-Katalogs, mindestens die drei hier erwähnten Rollen:

- *Datenbankadministrator*
  Verantwortlich für: Datenbank-Patching, Performance, Tuning, Ressource-Management

- *Account Manager*
  Verantwortlich für: Anlegen neuer Datenbank-Benutzer, Passwort-Management (außer von Database Vault Administrator), Passwort-Profiles
- *Sicherheitsadministrator*
  Verantwortlich für: Erstellung von Sicherheitsrichtlinien zum Schutz der Daten, Aktivierung beziehungsweise Deaktivierung von Database Vault (in Zusammenarbeit mit dem Datenbankadministrator)

Nachdem entsprechende Database Vault Sicherheitsrichtlinien erstellt worden sind, stellt sich immer wieder die Frage, ob es damit Probleme gibt und ob die Regeln wie beabsichtigt funktionieren. Bevor die Database Vault Sicherheitsrichtlinien „scharf" geschaltet werden, empfiehlt es sich, diese vorab ohne Risiko zu testen. Der Database Vault Simulations-Modus unterstützt bei der sicheren Implementierung der Database Vault Sicherheitsrichtlinien, damit man keine bösen Überraschungen erfährt.

Durch Database Vault werden Zugriffe auf Daten, Objekte und Kommandos restriktiv kontrolliert. Bevor nun neue Zugriffsregeln in einer Produktivumgebung aktiviert werden, wird der Simulations-Modus genutzt. In diesem Modus setzt Database Vault die Sicherheitsregeln nicht aktiv um, sondern protokolliert lediglich entsprechende Zugriffsverletzungen gemäß den erstellten Sicherheitsregeln. Dies hilft bei der Feinjustierung

der Sicherheitsregeln und vermeidet Probleme in der Produktion. Nachdem keine unbeabsichtigten Verstöße mehr protokolliert werden, kann der Simulations-Modus beendet werden. Danach wird Database Vault die entsprechenden Sicherheitsrichtlinien restriktiv umsetzen.

Prüfen Sie, ob die Regelungen zum Schutz sensitiver Daten noch zeitgemäß sind. Regularien fordern state-of-the-art technische Maßnahmen. Eine Überprüfung des Status Quo ist mit dem Database Security Assessment Tool im Bereich „Authorization Control" möglich.

3.2.5 VERSCHLÜSSELUNG

Die Oracle-Datenbank stellt Verschlüsselungsmöglichkeiten für unterschiedliche Sicherheitsmaßnahmen zur Verfügung: Zum einen ist es möglich, die Übertragung von Daten im Netzwerk nativ oder über SSL/TLS zu verschlüsseln. Zum anderen stehen prozedurale und deklarative Möglichkeiten zur Verfügung, gespeicherte Daten zu verschlüsseln.

3.2.5.1 VERSCHLÜSSELTE DATENÜBERTRAGUNG

Die Verschlüsselung der Kommunikation zwischen einem Oracle-Client und einer Oracle Datenbank, egal ob OCI, Thin-JDBC, Thick-JDBC oder ODBC, ist heutzutage obliga-

torisch. Die technische Umsetzung dieser Sicherheitsmaßnahme ist simpel und dank der Lizenzänderung, die Ende Juni 2013 in Kraft getreten ist, auch kostenfrei für alle noch im Support befindlichen Oracle Datenbank-Editionen. Wenn möglich, sollte die aktuellste Oracle Client Version verwendet werden. Gründe dafür sind die im Kapitel *Authentifizierung* (3.2.1) beschriebene Verwendung aktueller Passwortversionen, zum Beispiel 12a für die Nutzung eins SHA-2 Algorithmus zur Speicherung des Passworts und die Bereitstellung aktueller Verschlüsselungsalgorithmen wie AES256. Entsprechende Einstellungen können sowohl auf dem Datenbank-Server, für alle Oracle Clients verbindlich, als auch individuell je Oracle Client vorgenommen werden. Dies muss über die Parameter `SQLNET.ENCRYPTION_CLIENT` beziehungsweise `SQLNET.ENCRYPTION_SERVER` in der SQLNET.ORA erfolgen.

- REJECTED – Verschlüsselung wird grundsätzlich abgelehnt
- ACCEPTED (Default) – Verschlüsselung wird akzeptiert, wenn der Kommunikationspartner das möchte
- REQUESTED – Verschlüsselung wird gewünscht, aber nicht verlangt
- REQUIRED – Verschlüsselung wird verlangt

Werden diese Parameter nicht gesetzt, gelten die Standard-Werte. Das würde allerdings bedeuten, dass keine verschlüsselte Verbindung zustande kommt, da sowohl Client als auch Server den Wert ACCEPTED haben. Die einfachste Implementierung der Netzwerkverschlüsselung ist die zentrale Konfiguration über den Datenbank-Server. Hierbei entscheidet der Datenbankadministrator, ob eine Verschlüsselung erzwungen, erwünscht, akzeptiert oder sogar abgelehnt werden soll. Stellt der Datenbankadministrator in der SQLNET.ORA den Parameter SQLNET.ENCRYPTION_SERVER auf REQUESTED werden alle Oracle Clients dazu aufgefordert, eine verschlüsselte Verbindung aufzubauen. Da alle Oracle Clients den Standard-Wert ACCEPTED haben, wird also verschlüsselt, ohne den Client speziell konfigurieren zu müssen. Die Verwendung des Wertes REQUESTED auf dem Datenbank-Server hat den Vorteil, dass zum Beispiel spezielle oder älterer Oracle Clients über das Setzen des Parameters SQLNET.ENCRYPTION_CLIENT=REJECTED in der entsprechenden SQLNET.ORA auf dem Client, aus verschiedensten Gründen, keine Kommunikationsverschlüsselung aufbauen müssen.

Bei sensitiven Daten prüfen, ob SQLNET.ENCRYPTION_SERVER auf dem Datenbank-Server auf REQUIRED konfiguriert ist.

Auch der Verschlüsselungsalgorithmus kann gewählt werden. Je nach Version der Datenbank, beziehungsweise des Oracle-Clients, stehen verschiedene Algorithmen zur Verfügung. Der aktuellste Verschlüsselungsalgorithmus ist der Advanced Encryption Standard (AES). Soll dieser für die Kommunikationsverschlüsselung genutzt werden, kann dies über die Paramater `SQLNET.ENCRYPTION_TYPES_SERVER` beziehungsweise `SQLNET.ENCRYPTION_TYPES_CLIENT` eingestellt werden. Hier besteht die Möglichkeit, eine Liste von akzeptierten Verschlüsselungsalgorithmen anzugeben, zum Beispiel `SQLNET.ENCRYPTION_TYPES_SERVER =(AES256, AES128, 3DES168)`. Die Verschlüsselungsalgorithmen in dieser Liste werden beim Verbindungsaufbau von vorne nach hinten durchgegangen, bis sich Datenbank-Server und Client auf einen der Algorithmen geeinigt haben. Sollten sich Server und Client nicht einig werden, wird auch keine Verbindung aufgebaut. Dies ist sinnvoll, da zum Beispiel eingesetzte ältere Oracle Clients modernere Algorithmen nicht verstehen. Wird der Parameter nicht gesetzt, werden alle aktuell zur Verfügung stehenden Verschlüsselungsalgorithmen durchgegangen, bis sich Server und Client einig werden.

In der Dokumentation zur Datenbank findet sich eine Liste aller zur Verfügung stehenden Verschlüsselungsalgorithmen für die Verbindung unter „`SQLNET.ENCRYPTION_TYPES_CLIENT`“.

Prüfen Sie, ob die Verschlüsselungseinstellungen noch zeitgemäß sind. Regularien fordern state-of-the-art technische Maßnahmen. Eine Überprüfung durch das Database Security Assessment Tool ist unter „Network Configuration“ möglich.

3.2.5.2 VERSCHLÜSSELUNG AM SPEICHERORT

Neben der Kommunikationsverschlüsselung sollte auch das Verschlüsseln bei der Speicherung der Daten durch die Datenbank selber in Betracht gezogen werden. Hier existieren im Wesentlichen drei Ansätze:

- Verschlüsselung der Daten durch die Applikation selbst
- Verschlüsselung der Daten auf dem Filesystem durch eine Filesystemverschlüsselung
- Verschlüsselung der Daten durch die Datenbank beim Schreiben der Daten in ein Filesystem

Die Herausforderung hierbei ist, dass eingesetzte Applikationen beziehungsweise Verfahren uneingeschränkt und ohne Änderungen weiter betrieben werden können. Zudem kommen noch Aspekte bezüglich der Performance und der Datenmengen hinzu.

Diese Anforderungen sprechen gegen die Verschlüsselung durch die Applikation, da weder andere Applikationen (z.B.

ein Berichtswesen) mit verschlüsselten Daten arbeiten können, noch die Performance in Ermangelung von Indizes optimal ist. Hinzu kommt noch, dass alle Daten in einem TEXT-Format abgespeichert werden müssen, also nicht Datentyp-Echt sind, was wiederum die Datenmenge und Anwendungskomplexität erhöht.

Besser ist die Verschlüsselung über ein Filesystem. Dies ist transparent für die Datenbank und somit auch für die Applikation. Zu beachten ist jedoch, dass diese Art von Verschlüsselung ausschließlich die auf dem verschlüsselten Datenträger befindlichen Daten bei Verlust beziehungsweise bei einem Austausch des Datenträgers schützt. Werden die Daten auf ein anderes Medium kopiert, zum Beispiel durch ein Backup, muss darauf geachtet werden, dass das Zielmedium ebenfalls entsprechend verschlüsselt ist. Hinzu kommt, dass privilegierte Benutzer weiterhin transparenten Zugriff auf die dort gespeicherten Daten haben. Zudem gibt es eine Supporteinschränkung bei Verwendung von sogenannten Encryption-Agents, welche die Oracle Datenbank-Dateien verschlüsseln (siehe Support Note 2098275.1).

Die mit Abstand flexibelste und sicherste Variante, Daten verschlüsselt abzuspeichern, ist die Verschlüsselung durch die Datenbank selbst. Transparenz für die Applikation, Performance und speichermedienübergreifender Zugriffsschutz sind hier gewährleistet. Oracle bietet mit der Transparent Data Encryption (TDE) genau diese Eigenschaften. TDE ist Bestandteil der Advanced Security Option und muss für On-Premises Datenbanken

separat lizensiert werden. Anders sieht das bei einem Oracle Database Cloud Service aus: dort ist die Verschlüsselung in der Service-Subskription bereits enthalten.

Bei der Umsetzung von TDE sollten ein paar Fragen beantwortet werden:

Sind bereits Daten in der Datenbank enthalten, die nachträglich verschlüsselt werden sollen?

Welcher Verschlüsselungsalgorithmus soll verwendet werden?

Werden Datenbank-Technologien wie RAC, DG, OGG eingesetzt?

Wie werden verschlüsselte Datenbanken geklont, gesichert und wiederhergestellt?

Wie viele Datenbanken sollen verschlüsselt werden?

Wie soll der Verschlüsselungs-Schlüssel gespeichert beziehungsweise verwaltet werden?

Wer übernimmt das Schlüsselmanagement?

Manche dieser Fragen lassen sich technisch beantworten ein Teil davon nur organisatorisch. Die technischen Hilfestellungen sind in der Reihenfolge der Fragestellungen im Folgenden kurz aufgeführt.

In den meisten Fällen befinden sich bereits Daten in der Datenbank. Anhand der Daten-Klassifizierung werden entsprechende Schutzmaßnahmen erforderlich. Die gängigste Schutzmaßnahme, für jegliche Art von sensitiven Daten, ist die Verschlüsselung.

Bei der Suche nach sensitiven Daten in der Datenbank helfen Technologien wie die im Kapitel *DBSAT Tool* (3.1.1) erläuterte Möglichkeit, sensitive Daten anhand ihrer Metadaten zu ermitteln. Sollen sensitive Daten anhand von Daten-Mustern gesucht werden, steht die Oracle Enterprise Manager Funktionalität „Sensitive Data Discovery" in ADM zur Verfügung, siehe auch Kapitel *Enterprise Manager: Sensitive Datenanalyse (ADM) und IT Compliance* (3.1.2.).

Sind die Datenbanken mit sensiblen Daten identifiziert, stellt sich die Frage nach den Möglichkeiten zur nachträglichen Verschlüsselung. Hier existieren unterschiedliche Methoden, je nach Datenvolumen, Verfügbarkeit und Verschlüsselungsalgorithmus.

Die aktuellste Methode – mit TDE Live Conversion – wird im „Oracle Tipp 5941" online unter oracle.com erläutert.

Eine weitere Methode zur nachträglichen Verschlüsselung bei maximaler Verfügbarkeit wird im Whitepaper unter oracle.com „Converting to Transparent Data Encryption with Oracle Data Guard using Fast Offline Conversion" beschrieben.

Natürlich funktioniert auch die Methode mittels Daten-Export und Import via Oracle Datapump oder über die Online-Redefinition Funktionalität der Datenbank (über `DBMS_REDEFINITION`).

Bei Verwendung des Oracle Database Cloud Services gibt es bezüglich der Verschlüsselung ein paar spezielle Hinweise. Diese sind unter der Support Note 2359020.1 ausführlich beschrieben.

Die Sicherheit der verschlüsselten Information basiert im Wesentlichen auf der Stärke, Geheimhaltung und Verfügbarkeit des Schlüssels. Wobei die Stärke des Schlüssels durch seine Länge (128bit, 196bit, 256bit) und Erzeugung bestimmt wird. Die zur Verfügung stehenden Verschlüsselungsalgorithmen entsprechen dem heutigen Stand der Technik. Der wohl gängigste Algorithmus ist der Advanced Encryption Standard (AES). Mit einer Schlüssellänge von bis zu 256bit findet er Verwendung bei Daten mit der Klassifizierung TOP SECRET.

TDE verwendet jeweils ein Schlüsselpaar bestehend aus einem externen gespeicherten Master Encryption Key (MEK) und mehreren in der Datenbank gespeicherten Data Encryption Keys (DEK) pro Tabelle beziehungsweise Tablespace. Die Data Encryption Keys werden durch den MEK verschlüsselt. Die Verwaltung des DEKs übernimmt die

Datenbank eigenständig. Die Verwaltung des Master Encryption Keys hingegen muss mittels einer technisch-organisatorischen Maßnahme umgesetzt werden. Im Standard wird der MEK in einem Keystore gespeichert. Der von der Oracle Datenbank-Verschlüsselung verwendete Keystore zur Speicherung des TDE Master Keys – das sogenannte Oracle Wallet – ist eine mit AES256-bit und Kennwort verschlüsselte PKCS#12 Datei ewallet.p12 .

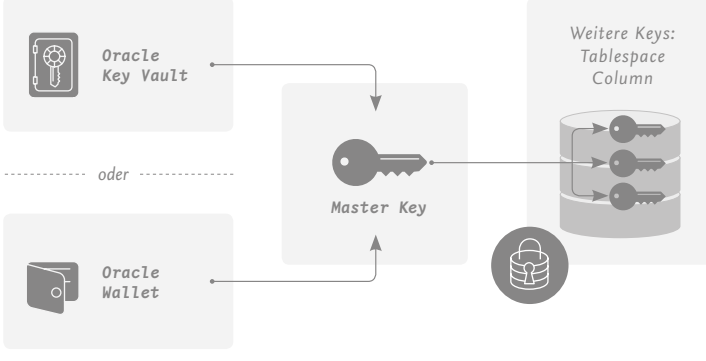

Abb.7: Schutz durch Database Vault

Zu beachten ist, dass jede Datenbank ihren eigenen Keystore besitzt. Beim Einsatz einer Real Application Cluster Datenbank müssen zum Beispiel alle Instanzen Zugriff auf denselben Keystore haben. Das Gleiche gilt beim Einsatz des Oracle Data Guards, bei Wiederherstellung von Datenbanken, Import beziehungsweise Export und beim

Klonen von Datenbanken, wenn TDE verwendet wird. Zudem beinhaltet der Keystore alle historischen Schlüssel der entsprechenden Datenbank. Wird bei einem Re-Keying ein neuer Schlüssel erstellt, muss der alte Schlüssel historisiert werden. Nur so ist das Wiederherstellen von Datenbanken aus älteren Datenbank-Backups, die gegebenenfalls einen älteren TDE-Master-Key verwenden, möglich. Hier wird deutlich, dass das Thema Schlüsselmanagement eine zentrale Rolle bei der Datenbankverschlüsselung spielt. Die Verantwortung zum sorgfältigen Umgang mit dem externen Schlüssel ist enorm. Steht der externe Schlüssel – TDE Master-Key – nicht zur Verfügung, sind die Daten verloren. Bei einer überschaubaren Anzahl von verschlüsselten Oracle Datenbanken, in einer überwiegend statischen Datenbanklandschaft, ist gegen das Verwenden des Oracle Wallets als Keystore des Master-Keys nichts einzuwenden. Das ändert sich schnell bei einer steigenden Anzahl von verschlüsselten Datenbanken in einer dynamischen Datenbanklandschaft, wie es zum Beispiel in einer Cloud-Architektur üblich ist. Gerade bei der Nutzung von Database as a Service ist es umso wichtiger, alle Datenbanken zu verschlüsseln und dabei die Schlüssel im eigenen Haus zu behalten. Nur so behält man als Besitzer der Daten weiterhin die Kontrolle.

Prüfen Sie die Stärke des Schutzes sensitiver Daten regelmäßig.

3.2.5.3 SCHLÜSSELMANAGEMENT MIT KEY VAULT

Oracle Key Vault (OKV) bietet eine moderne Umgebung zur sicheren, zentralisierten und zuverlässigen Speicherung, Verwaltung und Bereitstellung von Schlüsseln. Oracle Key Vault wird als Software Appliance ausgeliefert. Der Kunde stellt einen X86-64bit-Rechner zur ausschließlichen Nutzung für OKV zur Verfügung. Auf dem Rechner wird die OKV-Software mittels ISO-Images installiert. Der gesamte installierte Software-Stack besteht aus einem gehärteten Oracle Enterprise Linux (OEL) als Betriebssystem, einer Oracle-Datenbank und einer grafischen Benutzeroberfläche, der Oracle Key Vault Management Console. Die direkte Integration über PKCS#11 / KMIP ermöglicht dem Datenbank-Administrator ein vollkommen transparentes Arbeiten mit einer verschlüsselten Oracle Datenbank. Das Backup, die Bereitstellung und die Sicherung des TDE Master-Keys sind hiermit nicht mehr Aufgabe des Datenbank-Administrators.

Neben den Oracle TDE Master-Encryption-Keys können auch Schlüssel für mySQL, KERBEROS Key Tabellen, SSH Keys, Wallets und Java Keystores in Oracle Key Vault verwaltet werden.

Oracle Key Vault erleichtert den Umgang mit verschlüsselten Oracle Datenbanken beim Klonen, Import und Export, Backup und Recovery sowie bei verteilten Datenbanken

(Golden Gate), Real Application Cluster Umgebungen, Standby Umgebungen (Data Guard) und bei Datenbanken in der Cloud.

Oracle Key Vault ermöglicht zudem die On-Premises Verwaltung und Speicherung der Schlüssel bei Verwendung von Cloud Services, wie zum Beispiel bei Database as a Service. Hierzu existiert ein spezieller Oracle Key Vault Endpoint, der eine sichere Verbindung vom Cloud Service aus zum Oracle Key Vault (On-Premises) aufbaut. So kann der Datenbesitzer jederzeit den Zugriff auf seine in der Cloud betriebenen Daten unterbrechen.

Verifizieren Sie Ihr Schlüsselmanagement auch im Hinblick auf Recovery. Eine Überprüfung ist mit Hilfe des Database Security Assessment Tool unter „Data Encryption" möglich.

3.2.6 ANONYMISIEREN, PSEUDONYMISIEREN, AUSBLENDEN (MASKING, REDACTION)

Anonymisierung und Pseudonymisierung sind Anforderungen, welche sowohl im Zusammenhang mit der EU-DSGVO Erwähnung finden als auch bei der Generierung von Testdaten für Entwicklung und Funktionsüberprüfungen. Zur Umsetzung dieser Anforderungen bietet Oracle zwei unterschiedliche Technologien an. Data Masking wird eingesetzt

um Daten dauerhaft am Speicherort zu ändern, beispielsweise zu anonymisieren. Eine dynamische Änderung, die nur beim Abruf der Daten zum Ansatz kommt und damit die Daten am Speicherort unverändert lässt, ist Data Redaction.

3.2.6.1 ANONYMISIEREN UND PSEUDONYMISIEREN AM SPEICHERORT

Die Anonymisierung von Daten durch das Data Masking Pack maskiert sensible und vertrauliche Daten in der Entwicklungs- und Testumgebung.

Das Aufspüren von sensitiven Daten in einem komplexen Datenmodell stellt in verschiedenen Situationen eine große Herausforderung dar. So muss zum Beispiel im Rahmen einer Testdatengenerierung und der dazu gehörigen Anonymisierung festgelegt werden, welche Spalten denn überhaupt anonymisiert werden müssen und wenn ja, mit welcher Methode. Eine automatisierte Suche nach sensitiven Daten in Oracle Enterprise Manager ist Bestandteil des Data Masking and Subsetting Packs. Eine Beschreibung finden Sie in Kapitel *Enterprise Manager: Sensitive Datenanalyse* (3.1.2).

Mithilfe eines irreversiblen Prozesses, bei dem sensible Daten durch realistisch aussehende anonymisierte Daten ersetzt werden, stellt das Data Masking Pack sicher, dass die Originaldaten weder abgerufen noch wiederhergestellt

werden können. Die Integrität der Anwendung bleibt auch mit den maskierten Daten gewährleistet. Das Data Masking Pack bietet direkt einsatzbereite Datenmaskierungsvorlagen für verschiedene Datentypen, darunter Zufallszahlen, Zufallsziffern, Zufallsdaten und Konstanten. Darüber hinaus stehen weitere integrierte Maskierungsroutinen wie das Shuffling zur Verfügung. Shuffling mischt die Werte einer Spalte und verteilt sie neu auf die einzelnen Zeilen. Sollen die maskierten Werte realistisch sein, aber nicht auf den Originaldaten basieren, kann das Data Masking Pack die Originaldaten, etwa Namen und Adressen, durch fiktive Namen und Anschriften aus externen Datenquellen ersetzen. Bei speziellen Maskierungsanforderungen können zusätzlich eigene, benutzerdefinierte Maskierungsformate aufgenommen werden. Diese benutzerdefinierten Formate, auf Basis von PL/SQL, garantieren eine unbeschränkte Flexibilität.

Die Erstellung und Verwaltung der Datenmaskierungsvorlagen kann über das Data Masking Pack im Enterprise Manager erfolgen. Über eine graphische Oberfläche werden Sie durch alle notwendigen Schritte bis hin zu einer fertigen Datenmaskierungsvorlage unterstützt. Ist die Datenmaskierungsvorlage erstellt, generiert das Data Masking Pack aus den Maskierungsinformationen ein PL/SQL Package, welches die eigentliche Maskierung der Daten vornimmt. Danach existieren zwei mögliche Vorgehensweisen. Zum einen

können Sie das generierte PL/SQL Package auf einer Kopie beziehungsweise Klone der Produktionsdatenbank starten und damit die Daten maskieren. ACHTUNG: diese Variante führt zum unwiderruflichen Verlust der Originaldaten. Die zweite Variante bietet die Möglichkeit, die Daten während eines Datapump-Exports zu maskieren. Der Vorteil dieser Variante ist die direkte Maskierung auf der Produktionsdatenbank und erspart somit das Kopieren beziehungsweise Klonen der Datenbank. Sollten sich Änderungen im Datenmodell ergeben, können diese einfach über die graphische Oberfläche eingebracht werden. Natürlich muss nach jeder Änderung der Datenmaskierungsvorlage das PL/SQL-Package neu generiert werden.

3.2.6.2 ANONYMISIEREN UND PSEUDONYMISIEREN BEIM ABRUF DER DATEN

Im Gegensatz zur Anonymisierung von Daten durch das Data Masking Pack, bei dem die Daten unwiderruflich verändert werden, bietet die Pseudonymisierung mit Data Redaction die Möglichkeit, basierend auf den Produktionsdaten die Daten nur beim Lesen zu verändern. Die Daten selber bleiben unverändert. Diese Funktionalität ist ausschließlich für reine „Read-Only“ Anzeige in Applikationen gedacht, wie zum Beispiel für ein Berichtswesen, für das sensible Informationen unkenntlich gemacht werden müssen. Für Werkzeuge

wie SQL*Plus, die beliebige SQL-Statements erlauben, ist diese Technologie nur bedingt sinnvoll beziehungsweise nicht gedacht.

Die Umsetzung von Data Radaction setzt voraus, dass bekannt ist, in welchen Tabellen sich die Daten befinden und wie die Spalten heißen. Ist dies nicht bekannt, helfen Technologien aus dem Kapitel *DBSAT Tool* (3.1.1), um sensitive Daten aufzuspüren. Für die Pseudonymisierung stehen je nach Datentyp unterschiedliche Funktionen und Möglichkeiten zur Verfügung:

- Komplette und teilweise Ersetzung von Daten
- Ersetzung durch Zufallswerte
- Ersetzung mittels regulärer Ausdrücke

Beispiele:

Readaction	*Gespeicherter Wert*	*„Redacted“*
Vollständige Ersetzung	10.9.1992	1.1.2001
Partielle Ersetzung	052-52-2147	XXX-XX-2147
Mit regulärem Ausdruck	Timm.lee@acme.com	[redacted]@acme.com
Mit Zufallswerten	11334567678	87395521190

Welche Funktionalitäten für entsprechende Datentypen verwendet werden können, ist im Advanced Security Guide der Datenbank, Kapitel 12 *Data Redaction*, aufgeführt.

Data Redaction Policies lassen sich mittels SQL*Developer, Enterprise Manager oder nativ über das PL/SQL-Package `DBMS_REDACT` erstellen und verwalten. Auch für Data Redaction stehen vorgefertigte Daten-Formate wie Telefonnummern, Kreditkartennummern, Postleitzahlen zur Verfügung. Die Entscheidung, ob Daten ersetzt werden müssen, werden durch Bedingungen in der Redaction Policy gesteuert. Eine Bedingung (Expression) ist im Wesentlichen mit einer einfachen `WHERE` Klausel eines SQL Statements zu vergleichen. Die Auswertung der Bedingung muss ein boolescher Wert sein. In der Bedingung dürfen ausschließlich Operatoren wie `AND`, `OR`, `IN`, `NOT IN`, `=`, `!=`, `<>`, `<`, `>`, `>=` oder `<=` verwendet werden. Es können keine benutzerdefinierten Funktionen eingesetzt werden. Die Funktionen, die möglich sind, sind Namespace-Funktionen wie `SYS_CONTEXT` und `XS_SYS_CONTEXT`, STRING-Funktionen wie `SUBSTR` und `LENGTH`, APEX-Funktionen wie `V` und `NV` sowie Funktionen aus dem Bereich Label-Security.

Beispiel Bedingung mit der Namespace Funktion `SYS_CONTEXT`:

```
expression => 'SYS_CONTEXT(''USERENV'',''SESSION_USER'') =
''PSMITH'''
```

Hier werden nur Daten entsprechend der Redaction Policy ersetzt, wenn der Datenbankbenutzer PSMITH sie abfragt.

Die wohl einfachste Bedingung, die Verwendung finden könnte, lautet:

```
expression => '1=1'
```

Da diese Bedingung immer TRUE ist, werden die Daten entsprechend der Redaction Policy immer ersetzt.

Aktivierte Data Redaction Policies gelten für alle Datenbankbenutzer mit Ausnahme des SYSDBAs. Der SYSDBA besitzt das System-Privileg `EXEMPT REDACTION POLICY`, welche die Data Redaction Policies ignoriert. Die Datenbank-Rolle `DATAPUMP_EXP_FULL_DATABASE` enthält das System-Privileg `EXEMPT REDACTION POLICY` und verhält sich damit wie die DBA Rolle.

Prüfen Sie die Verwendung von Test- und Entwicklungsdatenbanken hinsichtlich sensitiver Daten. Verifizieren Sie Ihre Anwendungen hinsichtlich der Zweckbindung bei der Datenanzeige.
Eine Überprüfung ist durch das Database Security Assessment Tool unter „Fine-Grained Access Control“ möglich.

3.2.7 PATCHEN

Ohne Frage, Patchen ist die wichtigste Sicherheitsmaßnahme überhaupt. Keine IT-Komponente ist fehlerfrei, weder Hardware noch Software. Die Integrität, Vertraulichkeit und Verfügbarkeit von Informationen sind stetig durch Schwachstellen in Hard- und Software bedroht. Ist eine IT-Komponente ausreichend fehlerfrei und stabil, wird sie meist gegen eine neuere, mit neuen Schwachstellen, wieder ausgetauscht. Somit gehört das Patchen zu den nicht endenden Pflichten. Im Kontext der Oracle Datenbank heißt das, dass mindestens alle drei Monate – immer an einem Dienstag nach dem 17. Tag des Monats Januar, April, Juli und Oktober – wenn die aktuellen Release Updates für die Oracle Datenbank erscheinen – sich immer und immer wieder die gleiche Frage stellt:

Müssen wir dieses Release Update einspielen?

Einfach zu beantworten ist diese Frage, wenn es eine gesetzliche beziehungsweise regulatorische Anweisung dazu gibt, alle sicherheitsrelevanten Patches zeitnah nach dem Erscheinen zu installieren. Gibt es diese Anweisungen nicht, kann eine Entscheidung, ob der Patch eingespielt werden muss oder nicht, unter Zuhilfenahme der Common Vulnerability Scoring System (CVSS) Matrix getroffen werden.

Oracle veröffentlicht zu jedem Sicherheitspatch eine entsprechende CVSS Matrix. Oracle stellt eine spezielle Website für

Security Alerts mit relevanten Informationen rund um das Thema ‚Critical Patches' und Schwachstellen bereit. Von hier aus gelangt man auch zur aktuellen Oracle CVSS Matrix.

https://www.oracle.com/technetwork/topics/security/alerts-086861.html

Einfach den Link des aktuellsten Critical Patch Updates anklicken, entsprechende Oracle Produkte aus der dargestellten Tabelle auswählen, zum Beispiel „Oracle Database Server, Versions 11.2.0.4, 12.1.0.2, 12.2.0.1, 18.1, 18.2" und man bekommt die für das Produkt zur Verfügung gestellte CVSS Matrix. Unter folgendem Link findet sich zum Beispiel die Oracle Database Server Risk Matrix vom Juli 2018.

https://www.oracle.com/technetwork/security-advisory/cpujul2018-4258247.html#AppendixDB

Die in dieser Matrix gezeigten Informationen können helfen, eine Entscheidung darüber zu treffen, ob der Patch eingespielt werden sollte oder nicht. Die Matrix selber und wie sie zu bewerten ist, kann unter folgendem Link eingesehen werden.

https://www.first.org/cvss/v2/guide

Im aktuellen Beispiel (Juli 2018) werden drei Schwachstellen beschrieben. Interessant ist hier jeweils der „Base Score“ Wert. Zwei dieser Schwachstellen haben einen sehr hohen Wert – größer 8 – und scheinen leicht ausnutzbar zu sein, zu sehen am „Attack Complex“ Wert „low“. Sollte jetzt eine der hier aufgeführten Komponenten im Einsatz sein, wie hier zum Beispiel die Schwachstelle CVE-2018-2939 „Core RDBMS“, sollte die Entscheidung klar sein. Der Patch muss zeitnah eingespielt werden!

Monitoren Sie die Oracle Seite „Critical Patch Updates, Security Alerts and Bulletins“, um zu prüfen ob einer der entdeckten Fixes für Sie relevant sein könnte.

Unter Oracle Linux Bulletins auf der Oracle Alert Internet Seite, gibt es das Gleiche auch für Oracle Linux.

Weiter Informationen zum Thema Oracle Linux und Sicherheit finden sie unter: https://linux.oracle.com/security/

Eine Überprüfung durch das Database Security Assessment Tool unter „Basic Information“ ist ebenfalls möglich. Auch Oracle Enterprise Manager zeigt im Rahmen des Lifecycle Management Packs und Oracle Management Cloud Compliance & Configuration die Verfügbarkeit von Sicherheitspatches für die konkret verwendeten Datenbanksysteme an. Diese Anzeige beinhaltet jeweils die zum Download notwendigen Patchnummern.

3.3 MONITORING/DETECT INKL. AUDITING, THREAT DETECTION, IT COMPLIANCE

Nachhaltigkeit und die Einhaltung regulatorischer Anforderungen setzen eine stetige Überprüfung und Protokollierung jeglicher Veränderungen der IT-Landschaft voraus. Keine Sicherheit ohne Nachweise. Eine effektive Überprüfung erfordert, dass die Überwachungsrichtlinien die wichtigen Details über wichtige bzw. kritische Ereignisse erfassen.

Die Oracle Datenbank bietet bereits umfassende Audit-Funktionen. Mit dem Conditional Auditing besteht die Möglichkeit, eine genaue kontextabhängige Überwachung zu konfigurieren. Die vordefinierten Unified-Audit-Richtlinien vereinfachen den Konfigurations- und Verwaltungsprozess.

Oracle- und Nicht-Oracle-Datenbanken können eine große Menge an Audit-Daten erzeugen, die sinnvollerweise konsolidiert und für die Alarmierung und Berichterstattung gesichert werden. Oracle **Audit Vault** konsolidiert und sichert Auditdaten aus Datenbanken, Betriebssystemen und Verzeichnissen. Oracle Audit Vault bietet umfangreiche Reporting- und Warnfunktionen um Auditoren und Sicherheitspersonal detaillierte Informationen und Frühwarnungen bereitzustellen. Die große Vielfalt an Berichten wie z.B. zu Aktivitäten, Korrelationen, Anomalien und Trends ermöglichen eine schnelle und effiziente Möglichkeit Datenverletzungen zu erkennen und zu untersuchen.

Die zunehmende Anzahl von Angriffen auf Datenbanken über SQL-Injections oder missbrauchte Credentials von Insidern machte die Überwachung der Datenbankaktivität über eine vorgeschaltete SQL Firewall sinnvoll. Die **Datenbankfirewall** bietet eine anspruchsvolle SQL-Grammatikanalyse-Engine und unterstützt White List, Black List und Ausnahmeliste-basierte Richtlinien. Die Ereignisse werden im Audit Vault Server protokolliert, so dass Berichte auch diese Informationen enthalten.

Sogenannte Konfigurationsdrifts, dem Abweichen vom erwarteten Stand entweder durch aktives Zutun des Berechtigten oder einem unabsichtlichen Vergessen beim Installieren, sind ein Thema für eine spezielle Kategorie von Werkzeugen, der sogenannten **IT Compliance**. Dafür stehen On-Premises Komponenten über den Enterprise Manager zur Verfügung als auch ein cloud-basiertes Tool wie Oracle Management Cloud Configuration & Compliance. Mit Letzterem können auch ähnliche Umgebungen geclustert werden, um gemeinsame Typen von Umgebungen zu entdecken.

Alle aufgeführten Monitoring Ergebnisse werden typischerweise in ein **Securitymonitoring (SIEM inkl. UEBA)** aggregiert, entweder als Rohdaten oder bereits aufbereitet und reduziert. Oracle stellt hierzu ein Framework zur Verfügung, dass eine Vorintegration der vorher aufgeführten

Komponenten aufweist. Dieses Framework geht weit über eine Logfile Analyse hinaus. Mit Hilfe von Machine Learning Algorithmen werden Daten und Zugriffsmuster analysiert, Bewertungen abgeleitet und den Security Analysten zur weiteren Behandlung vorgelegt oder bereits automatisiert behandelt. Mit Hilfe von künstlicher Intelligenz ist es möglich, auf die stetig ändernden Bedrohungen reagieren zu können. Bekannte Threat Vektoren fließen ein bzw. werden aktualisiert. Die höchstmögliche Automation ermöglicht es, Ihrem Sicherheitsteam eine größere Anzahl von Systemen, bzw. der steigenden Anzahl von potentiellen Problemfällen, erfolgreich zu begegnen. Dieses Framework ist je nach Zusammenstellung auch bekannt als Oracle Identity SOC oder Trust Fabric.

3.3.1 AUDITING DER ORACLE DB

Im Datenbankumfeld versteht man unter Auditieren das Aufzeichnen von Benutzeraktivitäten. Seit Oracle Version 6 konnte protokolliert werden, wer mit welchen Befehlen auf welche Objekte zugreift oder wer welche Systemprivilegien nutzt. Obwohl das Datenbank-Auditing in der Vergangenheit eher ungern aktiviert wurde, ist es heutzutage eine wesentliche und obligatorische Sicherheitsmaßnahme geworden. Sicherheitsaspekte und Compliance-Anforderungen erzwingen ein immer umfangreicheres Monitoring

der Benutzeraktivitäten. Sichtbar wird diese veränderte Einstellung dem Auditing gegenüber auch in den neuen Möglichkeiten, welche mit der Datenbankversion 12c eingeführt wurden. Mit der Funktionalität Unified Auditing stellt Oracle eine komplett neue und extrem flexible Möglichkeit zur Verfügung, das Auditing so effizient wie möglich zu gestalten. Dazu gehört auch das Vermeiden des Speicherns unnötiger Daten, wie es unter anderem auch das Bundesdatenschutzgesetz fordert.

Die dazu nötige vollständige Überarbeitung der Architektur des Auditing Verfahrens bietet gleichzeitig die Gelegenheit, weitere Verbesserungen zu implementieren. Das betrifft sowohl die Performance als auch die Öffnung des gesamten Auditierens zur Nutzung durch diverse weitere Oracle Werkzeuge wie SQL*Loader und RMAN.

Das neue Auditing ist nicht abhängig von Initialisierungsparametern. Um zu überprüfen, ob das Unified Auditing für eine Datenbank der Version Oracle Database 12c aktiviert ist, kann man die View `V$OPTION` danach abfragen:

```
SELECT VALUE FROM V$OPTION WHERE PARAMETER = 'Unified Auditing';
```

Erscheint hier als Rückgabe der Wert `FALSE`, so bedeutet das nicht, dass das Unified Auditing nicht aktiviert ist. Es bedeutet vielmehr, dass das Unified Auditing nicht

als Standard-Auditingverfahren eingerichtet ist und dass beide Verfahren, das klassische und auch das neue Audit, nebeneinander aktiv sind. Dies wird auch als Mixed Mode Auditing bezeichnet. In diesem gemischten Modus gelten die neuen Standard Audit Einstellungen durch das Unified Auditing zusätzlich zu den gegebenenfalls bestehenden klassischen Audit-Einstellungen.

Das Unified Auditing verwendet sogenannte Audit Policies. Im Standard werden diverse Audit Policies mit ausgeliefert. Die wichtigste heißt `ORA_SECURECONFIG`. Sie ist mit dem Unified Auditing grundsätzlich aktiviert und bildet die Audit Optionen ab, die auch in Oracle Database 11g standardmäßig mit der Einstellung `AUDIT_TRAIL=DB` eingeschaltet sind. Sinnvollerweise sollte man sich eigene Policies anlegen, denn `ORA_SECURECONFIG` enthält zum Beispiel keinerlei Regeln für das Auditieren von Benutzerobjekten. Bei der Erstellung eigener Audit Policies wird deutlich, wie flexibel und exakt Auditing-Richtlinien umgesetzt werden können. Das Anlegen einer Policy geschieht mit dem Befehl `CREATE AUDIT POLICY`.

```
CREATE AUDIT POLICY HR_AUDITING
PRIVILEGES      SELECT ANY TABLE
ACTIONS         CREATE USER, ALTER USER, SELECT ON SCOTT.EMP
ROLES           RESOURCE
WHEN            'SYS_CONTEXT(''USERENV'', ''MODULE'') <> (''HRADMIN'')'
EVALUATE        PER STATEMENT
CONTAINER       = CURRENT;
```

Die Struktur einer Audit Policy sieht folgendermaßen aus:

- Eine Policy ist ein Datenbankobjekt und benötigt also einen Namen, hier `HR_AUDITING`.
- Es folgt eine Auflistung der Systemprivilegien und Aktionen, die auditiert werden sollen.
- Der Eintrag zum Parameter `ROLES`, hier `RESOURCE`, führt dazu, dass alle Privilegien, die im Rahmen der aktivierten Rolle `RESOURCE` genutzt werden, auditiert werden.
- Der Parameter `WHEN` bietet die neue und ausgesprochen hilfreiche Möglichkeit festzulegen, unter welchen Bedingungen ein Audit Eintrag geschrieben wird. Die Bedingung darf maximal 4000 Zeichen lang sein. Außerdem unterliegt sie einigen Restriktionen, die im Handbuch SQL Language Reference beschrieben sind.

- Im Beispiel wird davon ausgegangen, dass die Anwendung `HRADMIN` einen eigenen applikationsspezifischen Audit Trail schreibt. Deshalb wird für diese Anwendung das Auditing ausgeschlossen. Datenbankseitig werden also nur dann Zugriffe erfasst, wenn die Anwendung umgangen wird.
- Die `EVALUATE` Klausel legt fest, wann ein Audit Eintrag entsteht. Neben dem angegebenen `PER STATEMENT` stehen auch noch `PER SESSION` und `PER INSTANCE` zur Verfügung.
- Schließlich kann beim Einsatz in einer Container Datenbank (CDB) angegeben werden, ob die zu auditierenden Aktionen nur in der aktuellen Pluggable Database (PDB) erfasst werden sollen (hier so angegeben) oder in allen PDBs einer `CDB (CONTAINER = ALL)`. Für den Fall `CONTAINER = ALL` gelten einige besondere Regeln, auf die hier nicht näher eingegangen werden soll.

Nachdem mit dem Anlegen einer Policy festgelegt ist, was und unter welchen Bedingungen auditiert wird, muss das Auditing noch gestartet werden. Wie im ‚alten' Auditing geschieht das mit dem Befehl `AUDIT`.

```
AUDIT POLICY HR_AUDITING;
```

Für das Auswerten des Unified Auditing müssen bestimmte Voraussetzungen erfüllt sein. Es muss entweder die Rolle AUDIT_ADMIN oder die Rolle AUDIT_VIEWER verliehen worden sein. Letztere erlaubt ausschließlich das Lesen und wird typischerweise einem Auditor zur Verfügung gestellt.

Die Daten selber können über die View UNIFIED_AUDIT_TRAIL ausgelesen werden.

Prüfen Sie, ob das Unified Auditing nicht die Gelegenheit ist, das Audit zu verfeinern.

Sinnvollerweise werden die Auditdaten zeitnah in ein zentrales Audit-Warehouse überführt und von der eigentlichen Quell-Datenbank entfernt. Hierzu bietet Oracle eine passende Lösung: die Oracle Audit Vault & Database Firewall, siehe Kapitel *Zentrales DB Audit: Audit Vault & DB Firewall* (3.3.2).

Das Entfernen der Datenbank-Auditdaten kann über das PL/SQL-Package DBMS_AUDIT_MGMT gesteuert werden. DBMS_AUDIT_MGMT stellt dafür eine Prozedur, nämlich CLEAN_AUDIT_TRAIL, zur Verfügung. Dabei kann Bezug genommen werden auf einen Zeitstempel, der ebenfalls über das Package zu setzen ist. Das automatisierte Löschen der Auditdaten kann über einen Job erreicht werden, der über DBMS_AUDIT_MGMT aufgesetzt wird. Wichtig hierbei ist das korrekte Setzen des Zeitstempels, bis zu

welchem Zeitpunkt die Auditdaten gelöscht werden dürfen.

Bei Verwendung von Oracle Audit Vault & Database Firewall als zentralen Audit-Server wird der Zeitstempel der letzten übertragenen Auditdaten automatisch gesetzt.

Prüfen Sie, ob die Menge Ihrer Auditdaten nicht eine Zentralisierung in eine Audit Vault Datenbank nahelegt. Zudem könnten hier auch für ein SIEM Daten voraggregiert werden.
Eine Überprüfung ist durch das Database Security Assessment Tool unter „Auditing" möglich.

3.3.2 ZENTRALES DB AUDIT: AUDIT VAULT & DB FIREWALL

Oracle stellt mit dem Produkt Audit Vault & Database Firewall (AVDF) eine Lösung für ein zentrales Auditing-System zur Verfügung. Neben dem Database Activity Monitoring bietet diese Lösung auch die Möglichkeit, aktiv ins Geschehen einzugreifen, wie zum Beispiel das Blocken unbekannter potenziell gefährlicher SQL Kommandos. Ein Alerting bzw. Benachrichtigungen bei besonderen Vorkommnissen mittels E-Mail oder Remote SYSLOG Integration ist möglich.

Die Lösung besteht aus drei Komponenten:

Audit Vault Server

Der Audit Vault Server ist ein dedizierter Server und ist für die sichere Speicherung und Auswertung der Audit Daten verantwortlich, welche von den Kollektoren erfasst werden. Als Auditdaten-Repository wird eine Oracle Enterprise Edition Datenbank eingesetzt. Sie entspricht dem modernsten Stand heutiger Datawarehouses, die nahezu unendliche Skalierungsfähigkeit bei steigendem Datenvolumen ermöglicht.

Der Audit Vault Server wird entweder über eine eigene Web-Konsole oder über ein Command Line Interface verwaltet. Die Web-Konsole wird sowohl zur Administration durch den AVDF Administrator als auch zur Analyse durch den Auditor genutzt.

Eine separierte Oberfläche für den Auditor stellt ein Dashboard zur schnellen Übersicht bereit, hier werden Berichte angestoßen und Alerts definiert. Weiterhin werden hier zentral die Database Firewall Policies aller unterstützten Datenbanken und die Audit Richtlinien der Oracle Datenbanken erstellt und verwaltet.

Der AVDF Administrator bedient sich hingegen einer weiteren separierten Oberfläche, die zur Konfiguration und Verwaltung des Audit Vault Server dient. Der AVDF Administrator hat keinen Zugriff auf die gesammelten Protokolldaten

und eingestellten Policies. Der Auditor hat keinen Zugriff auf die administrative Oberfläche. Somit wird eine strikte Trennung der Aufgaben (SoD) durchgesetzt.

Audit Vault Agent (Hostbasiertes Kollektieren)
Der Audit Vault Agent befindet sich auf den Servern der Quellsysteme. Der Audit Vault Agent steuert das Übertragen der lokalen Audit Daten aus den Quelldatenbanken bzw. den nicht datenbankbezogenen Audit Trails zum Audit Vault Server.

Database Firewall (Netzbasiertes Kollektieren)
Die Firewall ist ein dedizierter Server, welcher SQL Statements direkt in Echtzeit analysiert die über das Netzwerk zur Datenbank gesendet werden. Die Analyse erfolgt auf Basis der Database Firewall Policies. Die Policies unterteilen sich in White Lists und Black Lists. White Lists enthalten alle erlaubten SQL Statements. Die Black Lists hingegen enthalten eine Liste der nicht erlaubten SQL Statements. Gewöhnlicherweise werden White Lists verwendet. Diese lassen sich durch einen sogenannten Trainingslauf automatisch aufbauen. Black Lists werden genommen, um bekannte Bedrohungsszenarien abzuwehren (ähnlich dem Virtual-Patching). Diese Policies lassen sich weiter verfeinern. Es können Ausnahmen auf Basis von Kontextinformationen wie OS Benutzer, DB Benutzer, Client Program, IP Adressen u.ä. definiert werden.

Diese analysierten Daten werden dann ebenfalls zum Audit Vault Server übertragen. Dieser Kollektor hat je nach Betriebsart neben der Möglichkeit des reinen Database Activity Monitoring auch die Fähigkeit, aktiv in die Kommunikation einzugreifen. Er basiert nicht auf nativer Datenbank-Audittechnologie und hat somit keinen Einfluss auf die Ressourcen des Datenbankservers.

Die Database Firewall bietet drei Betriebsarten:

- *DPE Mode: Database Policy Enforcement*
  In diesem Modus ist es möglich, neben dem Monitoren der SQL Statements (Detective Controls) auch aktiv in die Kommunikation einzugreifen (Corrective Controls). Dies ermöglicht das Blocken bzw. das Substituieren unbekannter oder schädlicher SQL Kommandos.
- *DAM Mode: Database Activity Monitoring*
  Dieser Modus ist auf das Monitoren der SQL Statements ausgerichtet und kann nicht aktiv eingreifen. Üblicherweise wird die Database Firewall in dieser Betriebsart an ein TAP bzw. SPAN (Mirror) Port im Netzwerk plaziert und hat dann keinen Einfluss auf die Performance.
- *Remote Monitoring*
  Diese Betriebsart ist eine Kombination aus dem Audit Vault Agenten und der Database Firewall. Hier werden die SQL Statements direkt an den Netzwerkkarten des Daten-

bankservers mitprotokolliert und dann zur „Nachbearbeitung“ an eine Database Firewall gesendet. Dieser Modus benötigt hoch privilegierte Rechte auf den Datenbankservern und verursacht je nach Anzahl der Datenbanken eine hohe Netzlast.

Audit Vault ermöglicht Audit Daten aus Oracle DBs, mySQL, Microsoft SQL Server, Sybase ASE und IBM DB2 (LUW) Datenbanken sowie anderen nicht datenbankbezogenen Audit Trails wie Betriebssystemen, MS Active Directory, XML-Dateien und Log-Tabellen zentral und geschützt vor Manipulation zu speichern und systemübergreifend auswertbar zu machen.

3.3.3 ÜBERWACHUNG UND SICHERSTELLUNG DES KONFIGURATIONSMANAGEMENTS (IT COMPLIANCE)

Aus Gründen des Datenschutzes und der Verpflichtung zum Nachweis des Einhaltens von Regularien ist eine übergreifende Sicht auf die komplette Infrastruktur und Anwendungslandschaft notwendig. Dies ist unter dem Begriff IT Compliance zusammengefasst. Für die Sicherstellung oder das Monitoring von IT Compliance stehen zwei Werkzeuge von Oracle zur Verfügung: Enterprise Manager Lifecycle Management Pack, eine On-Premises Komponente, und ein cloud-basiertes Werkzeug, der Oracle Management

Cloud Configuration und Compliance Service. Beide können auf vorgefertigte IT Compliance Prüfungen zugreifen, z.B. DB STIG oder CIS for DB. Mit Benachrichtigungen abhängig von den Ergebnissen wird eine Integration mit Drittanbietersystemen für Incident Management Systeme (z.B. Services Now, PagerDuty) oder anderen Kommunikationsplattformen wie Slack ermöglicht. Workflows können definiert werden, die beispielsweise eine sichere DB Konfiguration vornehmen und dann eine erneute Compliance Prüfung durchführen.

Haben Sie cloudbasierte Systeme, die nur per REST API ansprechbar sind, ist Oracle Management Cloud die bessere Wahl, da mit dem Oracle Enterprise Manager keine REST Endpoints ansprechbar sind.

3.3.3.1 ENTERPRISE MANAGER

Das Framework von Oracle Enterprise Manager (EM) beinhaltet einen effizienten Mechanismus, Metriken auf den Servern durch einen Agenten zu erfassen, diese gegen Schwellenwerte zu vergleichen, ggf. Warnungen zu erzeugen und eine historische Speicherung im zentralen EM Repository vorzunehmen. Dieser Mechanismus ist die Basis für ein automatisiertes Compliance Management, denn unter den vom Agenten gesammelten Metriken sind auch viele aus dem Bereich Konfiguration.

Die Nutzung dieser Daten, auch wie es im Folgenden beschrieben wird, unterliegt der Lizenz des Lifecycle Management Packs.

Die automatisch gesammelten Konfigurationsmetriken, die auch durch benutzerdefinierte Metriken ergänzt werden können, können zum Beispiel dazu genutzt werden, um eine vorhandene Konfiguration eines Systems gegen Regelwerke zu prüfen. Dieses Regelwerk wird auch Compliance Framework genannt. Jedes Compliance Framework besteht aus verschiedenen ‚Standards', die ihrerseits aus den einzelnen Regeln bestehen, auf die die einzelnen Metriken getestet werden.

Oracle Enterprise Manager liefert verschiedene vorgefertigte Compliance Frameworks, wie zum Beispiel die folgenden aus dem **Database Security Technical Implementation Guide** (STIG).

Mit diesem Framework werden Regeln zu einem sicheren Betrieb von Oracle Datenbanken überprüft. Diese Regeln wurden von der amerikanischen Defense Information Systems Agency (DISA) in Zusammenarbeit mit Oracle erstellt und können auf der Webseite *https://iase.disa.mil* eingesehen werden. Unter all den damit überprüften Regeln gehören zum Beispiel:

- Überprüfungen hinsichtlich der Defaultpasswörter von Datenbankbenutzern, die natürlich geändert werden müssen, sowie, ob gut bekannte Demo-Benutzer (zum Beispiel SCOTT) in der Datenbank existieren.
- Überprüfungen hinsichtlich der Regeln für die Verwendung von Passwörtern, also deren Komplexität und Lebensdauer.
- Überprüfungen, ob Systemprivilegien an PUBLIC vergeben wurden.
- Überprüfungen, ob Instanzparameter, wie zum Beispiel `REMOTE_OS_AUTHENT` und `REMOTE_OS_ROLES` auf `FALSE` gesetzt sind, bzw. `REMOTE_LOGIN_PASSWORDFILE` auf `EXCLUSIVE` oder `NONE` gesetzt ist.
- Überprüfungen hinsichtlich der Dateiberechtigungen, die für die Oracle Software Dateien gesetzt sind.
- Überprüfung hinsichtlich der Netzwerkeinstellungen bzgl. Verschlüsselung.

Das **Support Framework** überprüft, ob sich Ihre Systeme noch im Rahmen des Oracle Supports bewegen und damit noch Sicherheitspatches zur Verfügung stehen.

Daneben wird automatisch auch das Vorhandensein von Sicherheitspatches für die von EM verwalteten Systeme

geprüft, wenn EM bzgl. der Patchsuche im Online Modus betrieben wird. In diesem Fall wird unter anderem auf der Compliance Dashboard Seite angezeigt, welche Ihrer Systeme mit einem Sicherheitspatch aktualisiert werden sollten.

Sobald ein von Oracle Enterprise Manager überwachtes System einem Framework zugeordnet ist, erfolgt automatisch alle 24 Stunden eine Überprüfung aller Compliance Regeln. Außerhalb dieses Intervalls können auch jederzeit manuell initiierte Überprüfungen aufgeführt werden. Im Falle einer Verletzung einer Regel wird entsprechend der Regel eine Anpassung des Compliance Scores vorgenommen. Anhand dieses Compliance Scores kann im Ergebnis für jedes System eine schnelle Einschätzung vorgenommen werden, inwieweit die aktuell bestehende Konfiguration dieses Systems den Compliance Vorgaben entspricht. Ein Dashboard zeigt dieses auch zusammenfassend für alle Systeme übersichtlich an.

Neben der Nutzung der vorgefertigten Compliance Frameworks können auch eigene Frameworks erstellt werden, sowohl unter Verwendung der vorgefertigten Compliance Regeln, als auch durch Erstellung neuer Regeln. In der Praxis zeigt sich oft, dass die Anforderungen von Kunden meistens durch die Standardregeln abgedeckt sind.

So gibt es zum Beispiel einen Compliance Standard mit dem Namen „**Basic Security Configuration For Oracle Database**“ (weitere Standards sind für Cluster Datenbanken – RAC – verfügbar), der unabhängig von den STIG-Regeln ein Mindestmaß an sicherem Betrieb beschreibt. Hier finden sich alle oben genannten Regeln auch wieder. Wer nicht gleich mit dem offiziellen Regelwerk von STIG arbeiten möchte, kann somit sehr schnell alle Oracle Datenbanken, die in Oracle Enterprise Manager verwaltet werden, mit diesem Regelwerk überprüfen lassen.

Alle Compliance Daten sind im EM Repository gespeichert und können mit dem integrierten BI Publisher anhand von Reports dargestellt werden.

Um sicherzustellen, dass wirklich alle Datenbanksysteme in einem Unternehmen mit diesem Compliance Management überprüft werden, kann in Oracle Enterprise Manager auch eine Suche nach Oracle Datenbanken im gesamten Netzwerk über eine Bandbreite von IP-Adressen vorgenommen werden. Wird eine bislang unbekannte Datenbank entdeckt, kann diese dann in EM eingebunden und mit dem Compliance Management verknüpft werden.

Mit der Nutzung dieser automatischen Auswertung der Compliance Frameworks stellen Sie nicht nur einen Betrieb sicher, der den Vorgaben entspricht, sondern bereiten sich auch optimal auf eventuell stattfindende externe Audits vor.

3.3.3.2 ORACLE MANAGEMENT CLOUD CONFIGURATION & COMPLIANCE

Oracle Management Cloud Configuration & Compliance Service (OMC C&C) ist eine cloud-basierte Compliance Lösung, die Prüfungen der IT Assets sowohl im eigenen Datacenter als auch bei einem externen Anbieter bzw. in einer Cloud durchführt. Die Definition, Verwaltung und Kontrolle erfolgt in einer browserbasierten Oberfläche. Zum Zugriff auf die zu prüfenden Systeme sind unterschiedliche Mechanismen wie direkte Nutzung von REST Schnittstellen oder die Verwendung von Agents integriert.

IT Compliance Prüfungen werden als Assessments definiert. Ein Assessment beinhaltet mindestens eine Regel („Rule“) oder ein Satz an Regeln („Ruleset“). Beispielsweise nutzt die DB CIS Prüfung mehrere Regelsätze. Eine Regel ist ein vorgefertigtes oder selbst erstelltes Skript. Das Skript verarbeitet die Eingabeparameter und liefert das Ergebnis der Prüfung zurück. Das Ergebnis führt den Status der Prüfung (Pass, Failed oder Error) auf und liefert Zusatzinformationen. Zusätzlich können weitere Metadaten, wie z.B. Beschreibung, Tags, Benachrichtigungen und Gegenmaßnahmen (Remediate) einer Regel hinterlegt werden. Neben den hostskriptbasierenden Regeln gibt es die cloud-basierten Regeln (cloud rules), die eine REST API nutzen. Für Amazon und Oracle Cloud sind cloud basierte Regeln bereitgestellt.

Ähnlich funktioniert es auch im Oracle Enterprise Manager, der im vorangegangen Abschnitt beschrieben ist. Oracle Management Cloud ist jedoch nicht eine Oracle Enterprise Manager Installation, die über einen Cloud-Service zur Verfügung gestellt wird sondern eine eigenständige Implementierung.

Die definierten Assessments werden regelmäßig bzw. beim Auftreten von Anomalien ausgeführt. Die Konfiguration der Assessments, wann und wie oft sie ausgeführt werden sollen, erfolgt in Assessments Templates. Eine adhoc Ausführung ist ebenfalls möglich.

Für die Oracle Datenbank werden Compliance Assessments inklusive Regeln und Benachrichtigungen für NIST, STIG, CIS und PCI-DSS bereitgestellt.

Durch die Nutzung der Security Content Automation Protocol (SCAP) Engine des Betriebssystems wird die Betriebssystem Compliance Prüfung durchgeführt. Oracle bietet für Oracle Linux, Red Hat und Solaris fertige Betriebssystem Compliance Prüfungen an.

Das Ergebnis der Analyse wird in der Oracle Management Cloud Oberfläche unter dem Menüpunkt „Configuration & Compliance“ im IT Compliance Dashboard dargestellt. Ein Drill Down in Detailinformationen ist möglich. Im

Administrationsbereich von OMC C&C werden die Compliance Regeln, Regelsätze, Benachrichtigungen, Assessments (Templates) und Compliance Engines verwaltet und konfiguriert.

Auf Basis des Ergebnis einer Regelprüfung können Maßnahmen, sogenannte Remediations definiert werden, die als Workflow in OMC C&C definiert sind und vom Oracle Management Cloud Orchestration Service ausgeführt werden.

Im Orchestration Service definiert der Nutzer Workflows und kann damit auf bestimmte Events, wie z.B. Anomalie oder Compliance Verstoß, reagieren und Aktionen oder Gegenmaßnahmen ausführen. Ein Workflow besteht aus einem oder mehreren Schritten und wird auf einem bzw. mehreren Systemen ausgeführt. Dieser wird im JSON Format beschrieben. Die einzelnen Aktivitäten, wie z.B. „DB beenden“, werden im Workflow Schritt gekapselt. Das Ergebnis des Workflow Schritts beinhaltet einen Status und eine entsprechende Statusmeldung.

Eine Instanz des Workflows ist eine Workflow Submission und im internen Datenmodell als Entity abgebildet. Diese beinhaltet Ausführungsplan, Workflow Schritte, Eingabewerte, Workers, Credential Referenz und Benachrichtigungen bzw. Statusmeldungen.

Workflows werden durch die Oracle Management Cloud entweder direkt über die REST API oder indirekt über einen Oracle Management Cloud Agenten ausgeführt. Beispielsweise beendet der Workflow DB Instanzen, die nicht konform sind, ändert entsprechend der Policy die Konfiguration und startet danach die DB Instanz wieder.

Compliance Verstöße können damit behandelt und Sicherheitslücken durch automatische Korrekturen geschlossen werden.

Bei OMC gibt es kostenfreie Trials. Damit können Sie eine Umgebung (On-Premises oder in der Cloud) testen.

3.3.4 SECURITY MONITORING, THREAT DETECTION (SIEM, UEBA)

Mit den bisher dargestellten Werkzeugen und Funktionalitäten im Kapitel *Monitoring* wurden die Aktivitäten in der Datenbank durch Datenbanknutzer und die sicheren Konfiguration der Datenbank betrachtet. Im folgenden Abschnitt geht das Monitoring einen Schritt weiter und und bietet ein Security Monitoring, dass die Aktivitäten hinsichtlich Policy Verletzungen, Anomalien, bekannten und unbekannten Threats auswertet:

- *Audit/Audit Vault:* Policies zum Protokollieren und Auswerten der DB Benutzer; Reporterzeugung typischerweise pro Datenbank

- *IT Compliance:* Policies zur Überwachung der Konfiguration und dem Nachweis für Regularien
- *Security Monitoring:* systemübergreifendes Monitoring der Aktivitäten und Systeme. Hier erfolgt eine Threat- und Schwachstellenanlayse mit Funktionalitäten aus dem Bereich SIEM, UEBA, CASB und IT Compliance.

Hinter den IT Begriffen stecken verschiedene Oracle Komponenten mit unterschiedlichen Überschneidungen:

Mit dem Oracle Cloud Access Security Broker (CASB) kann die Nutzung von Cloud Services gemonitort werden um einerseits eine Schatten-IT zu entdecken und andererseits die Cloud Nutzung inklusive Konfiguration an sich zu kontrollieren. Gängige Services wie Salesforce, Office365 oder AWS sind vorintegriert. Neben Policies, die die Systeme überwachen, wird Machine Learning genutzt, um abweichende Verhalten zu erkennen (UEBA) und verdächtige Aktionen aufzuspüren (Threat Analyse). Eine Automatisierung kann hinterlegt werden. Hier im Falle der Datenbanken können diese in der Oracle Cloud, bei AWS oder Azure gemonitort werden.

Oracle Management Cloud Security Monitoring Analytics (OMC SMA) basiert auf den Logfile-Integrationen von Systemen. Diese können On-Premises oder in der Cloud sein. Es erfolgt die Korrelation der Accounts eines Benutzers und

die Untersuchung nach verdächtigen oder abweichenden Aktionen. Eine Automatisierung von Tätigkeiten auf Basis von Problemfällen kann durch den Oracle Management Cloud Orchestration Service erfolgen. Hier im Falle der Datenbanken können Systeme On-Premises, in der Oracle Cloud, bei AWS oder Azure gemonitort werden.

Oracle Identity Cloud Service, Oracle Cloud Access Security Broker, Oracle Management Cloud Configuration & Compliance Service und weiteren Feeds können in OMC SMA integriert werden, zusammen ergeben sie das auf der Oracle Open World 2017 vorgestellte Identity SOC bzw. die in 2018 vorgestellte Trust Fabric.

Im Folgenden beschränkt sich die Darstellung auf OMC SMA. Der allgemeine Ablauf lässt sich wie folgt darstellen:

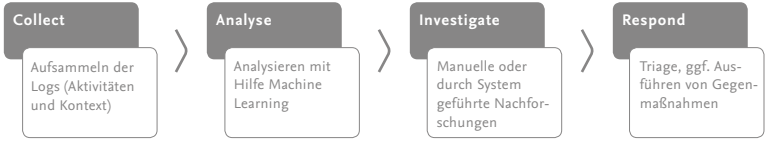

Abb. 8: Allgemeiner Ablauf OMC SMA

Der OMC SMA liest sicherheitsrelevante Informationen aus den gemonitorten Systemen direkt über REST API oder Agents aus. OMC SMA bündelt die Ergebnisse der Analyse von sicherheitsrelevanten Log Dateien, CASB Risk Events, Compliance Assessments, Threat Feeds, Assets und Users/Identities und stellt diese in domänenspezifischen Dashboards dar.

Vor der Analyse der genannten unterschiedlichen Datenquellen werden die Daten geparst und um weitere Metadaten angereichert. Somit entsteht ein einheitliches Datenformat, das Security Event Event Format (SEF). Auf dessen Basis werden Algorithmen des maschinellen Lernens (Anomalien erkennen) und Korrelationsregeln (Pattern Detection) angewendet. Das Resultat ist ein abgeleitetes SEF Schema, dass eine Anomalie oder Korrelationsevent (Pattern Match) beinhalten kann.

Das Security Event bzw. das Security Event Format enthält sieben Informationsbereiche:

Die **Kategorisierung** der Aktionen (z.B. Autorisierung), **Resultate** der Aktionen, **Ressourcen** die gelesen bzw. modifiziert wurden, **Assets** (umfassen die Ressourcen), **Nutzerdaten** (User/Identity), **Risiken** und **Metadaten** (Beschreibende Informationen zum Event).

Beispielsweise sieht ein SEF Eintrag des auf Basis des gelesenen DB Audit Log wie folgt aus:

sefEventStartTime: 2017-02-02T13:21:01

sefCategory: authentication.login // security enrichment

sefResult: success // security enrichment

sefDestinationResourceName: mbaker

sefDestinationResourceType: application.database.account

sefDestinationEPName: FINDB

sefDestinationEPType: omc_oracle_db_instance

sefDestinationEPAccountName: mbaker2

sefDestinationUserName: mary.baker@acmeloric.com // user context

sefDestinationUserPrimaryOrg: Finance // user context

Das ursprünglich DB Audit Log wurde durch weitere Daten angereichert und in SEF überführt, z.B. der Datenbank Name, User/Identity Daten. In der Analyse (Analyze Phase) werden nun die Pattern Matching und Anomaly Detection Verfahren angewendet.

Damit finden Sie Korrelationen und Anomalien in SQL Abfragen. In der Dokumentation „Using Oracle Security Monitoring and Analytics“ in Kapitel 3.a sind die vorhandenen Korrelationsregeln und in Kapitel 3.b die genutzten Machine Learning Algorithmen genauer beschrieben.

Anschließend werden die Ergebnisse weiter in der Investigationsphase betrachtet.

Am Ende erhält der Anwender eine Risikobetrachtung mit möglichen Vorschlägen, um die Bedrohung abzuwehren. Mögliche Gegenmaßnahmen können auf Basis von vorgefertigten Remediation Workflows auch automatisch ausgeführt werden. Das Ausführen der Gegenmaßnahmen kann der Oracle Management Cloud Orchestration Service übernehmen.

Für einen DBA gibt es ein Oracle Management Cloud Dashboard „Oracle Database Security“, das unter anderem auf einen Blick besondere DB Aktivitäten, SQL Anomalien, Threat Trends, Startups/ Shutdowns der DB zeigt.

3.4 CLOUD DATA SECURITY SERVICE

Einige der Funktionen, die in den einzelnen Abschnitten beschrieben wurden, sind auch gebündelt über einen einzigen Cloud Service verfügbar, den Cloud Data Security Service. Der Service ist noch nicht verfügbar (Stand Oktober 2018) und wird in der ersten Version auf Oracle Datenbanken beschränkt sein. Folgende Funktionen sind geplant:

Database Assessment (Security-relevante Überprüfungen der Konfiguration ähnlich DBSAT)

- Security Konfiguration
- User und Berechtigungen
- Vergleiche mit Best Practices und mit Compliance Vorgaben (EU-GDPR, CIS)
- Handlungsempfehlungen

User Assessment (Security-relevante Überprüfungen der Benutzer und Berechtigungen ähnlich DBSAT)

- Benutzer mit höchstem potentiellen Risiko
- Benutzerdefinition: type of user, pwd policies, etc.
- Benutzeraktivitäten: last login, audit data, etc.

Data Discovery (Security-relevante Überprüfungen hinsichtlich sensitiver Daten – ähnlich DBSAT)

- Über 80+ vorgefertigte Templates für sensitive Daten wie
  - PII: Name, address, phone, national ID (SSN), passport, email, Age, date of birth, sex, race, citizenship
  - IT Daten: IP address, pwd, userID
  - Finanzdaten: Credit card (CCN), bank account
  - Gesundheitsdaten: height, weight, health care provider
  - Beschäftigungsverhältnisdaten: income, stock
- Unterstützung von benutzerdefinierten sensitiven Daten

Data Masking (Masking der Daten für Nutzung durch Dev/Test/Partner usw.)

- Beibehaltung der Formate wie National Identifiers, Credit card usw.
- Zufallsdatengenerator für Datum, Zahlen und Zeichenketten
- Masking Transformationen: Shuffle, Conditional, Compound, Reversible, Deterministic, SQL expression, User defined
- Masking Report: # of values masked

Activity Auditing *(Security-relevante Auswertung der DB Audit Logs)*

- Aufsammeln der Auditdaten
- Provisionierung von default audit policies, compliance policies, custom policies, alert policies

Die Ergebnisse werden webbasiert in einem Dashboard dargestellt und stehen zur Weiterverarbeitung zur Verfügung.

4 Oracle Datenbanken XE

Mit der Oracle Datenbank 18c Express Edition (XE) gibt es erstmalig die Möglichkeit, Enterprise Features in einer Express Edition zu testen. Das sind Features der Oracle Database, Oracle Database Optionen und Oracle Management Packs. Folgende Security-relevante Funktionalitäten stehen nun damit zur Verfügung:

- Oracle Advanced Security inklusive Redaction, Column-Level Encryption, Tablespace Encryption (mit Einschränkung)
- Oracle Database Vault inklusive Privilege Analysis
- Oracle Label Security
- Centrally Managed Users, nicht jedoch Enterprise User Security
- Transparent Sensitive Data Protection
- Virtual Private Database

Mehr Informationen dazu unter Oracle Database Express Edition: Licensing Information User Manual.

5 Oracle Datenbanken als Cloud Service

Oracle Datenbanken werden von Oracle auch als Cloud-Services bereitgestellt. Es gibt dabei verschiedene Typen von Datenbank Cloud Services:

- Database as a Service mit Instanzen auf einer dedizierten Maschine oder innerhalb einer VM
- Exadata Services
- „Autonomous“ Datenbanken: ATP und ADW

Diese Datenbankservices können dabei sowohl aus einem der Cloud Rechenzentren von Oracle bereitgestellt werden als auch in Ihrem Rechenzentrum auf einer von Oracle bereitgestellten Appliance z.B. Cloud@Customer genutzt werden. ATP und ADW sind aktuell nur in der Oracle Cloud verfügbar (Stand Oktober 2018). Bei der On-Premises Appliance ist wählbar, inwieweit der Betrieb durch Oracle zentral erfolgen soll oder im sogenannten ‚disconnected mode‘ mit dediziertem Personal vor Ort.

5.1 ORACLE DATENBANK CLOUD SERVICE

Eine Oracle Cloud Datenbank wird entweder in einem Cloud Rechenzentrum von Oracle bereitgestellt oder mit Hilfe der Cloud Appliance „Cloud@Customer“ in Ihrem Rechenzentrum. Die Datenbanksoftware (z.B. 18c) ist gleich, lediglich die Lizenzbundles sind verschieden. Zusätzlich gibt es verschiedene Ausprägungen hinsichtlich Kapazität, Speicher und Durchsatz. Auch Exadata Systeme sind in der Cloud verfügbar.

Die Liste der verfügbaren Varianten ist unter https://cloud.oracle.com sowohl im Bereich IaaS als auch PaaS zu finden.

Cloud Datenbanken werden für Sie von Dritten, hier Oracle Mitarbeiter, bereitgestellt. Daher gibt es für Cloud Systeme Sicherheitszertifizierungen im Sinne des Betriebs einer Umgebung. Aufzuführen sind hier für Oracle Datenbanken in Frankfurt beispielsweise ISO27001 oder PCI-DSS Attestation.

Um dem Paradigma „Security-by-Default“ Rechnung zu tragen, ist die Grundinstallation der Oracle Datenbank wie folgt abgesichert:

- Bereitstellung mit aktuellsten Sicherheitspatches im Image
- Grundhärtung der Datenbank, indem nicht benötigte Schemas gelöscht sind und die Datenbank AES verschlüsselt ist (ohne benötigte Zusatzlizenz)

- Grundhärtung des Betriebssystems, indem nicht benötigte Services inaktiv sind, ein Login nur mit SSH Keys möglich ist und die lokale Firewall nur ssh als offenen Port zur Verfügung stellt
- Abschottung im umgebenden Cloud Netzwerk (VCN) wie von Ihnen vorab konfiguriert, z.B. System nur über VPN erreichbar
- Bereitstellung einer Konsole für Patching, Backup, Restore, Starten und Stoppen von Services
- Zugriff auf die Konsole nur für von Ihnen eingerichtete Benutzer über das von Ihnen festgelegte Verfahren zur (starken) Authentifizierung

Die unter Kapitel 3 dargestellten Technologien können alle, außer bei reinen Schema Services, zusätzlich wie folgt eingesetzt werden:

3.1 ***Überprüfen/Assess***

Sicherheitstechnologie	*Hinweise*
3.1.1 DBSAT Tool: Konfiguration, Usermanagement, Sensitive Daten (Dictionary)	DBSAT Zugriff entweder lokal oder per SQL (DBSAT > 2.0.2)
3.1.2 Enterprise Manager: Sensitive Datenanalyse (ADM) und IT Compliance	Entsprechender Agent auf OS installierbar
3.1.3 Oracle Management Cloud: Clustern von Umgebungen und IT Compliance	OMC integrierbar bzw. EM Agent installierbar
3.1.4 Oracle Database Vault: Genutzte Berechtigungen	Aktivierung DBVault lizenzabhängig möglich

3.2 *Schützen/Prevent*

Sicherheitstechnologie	*Hinweise*
3.2.2 Authorisierung	Direkte und delegierte Authorisierung
3.2.3 Hardening	Basishärtung gemäß voranstehendem Text
3.2.4 Zugriffsmanagement: Gewaltentrennung / SoD (DB Vault)	Aktivierung DBVault lizenzabhängig möglich
3.2.5 Verschlüsselung	Verschlüsselung (über TDE) ohne Zusatzlizenz
3.2.6 Anonymisieren, Pseudonymisieren, Ausblenden (Masking, Redaction)	Einsatz abh. von der Lizenz
3.2.7 Patchen	Ja, CLI oder GUI; passende Patches sind bereitgestellt und auswählbar

3.3 *Monitoring, Auditing, Threat Detection, IT Compliance*

Sicherheitstechnologie	*Hinweise*
3.3.1 Auditing der Oracle DB	Möglich
3.3.2 Zentrales DB Audit: Audit Vault & DB Firewall	Einsatz abhängig von der Lizenz
3.3.3 Überwachung und Sicherstellung des Konfigurationsmanagements (IT Compliance)	EM oder OMC zusätzlich notwendig
3.3.4 Security Monitoring, Threat Detection (SIEM, UEBA)	OMC zusätzlich notwendig

Stand: Oktober 2018

Weitere Informationen finden sich im Administration Guide des jeweiligen Cloud Datenbankservices unter https://cloud.oracle.com

5.2 ORACLE AUTONOMOUS CLOUD DATENBANKEN

Im Vergleich zu den cloud-basierten Oracle Datenbanken weisen die sogenannten Autonomous Systeme einen höheren Automatisierungsgrad auf. Dieser ist in den Kategorien „self-securing“, „self-driving“ und „self-repairing“ zusammengefasst. Entsprechende Service Level Agreements bis hoch zu 99,995% inkl. geplanten Wartungsarbeiten geben dem Benutzer dabei die nötige Sicherheit.

Security relevante Merkmale dieser Systeme sind dabei:

- Basishärtung Betriebssystem und Datenbank
- Kein Betriebssystemzugriff
- Automatische Verschlüsselung
- Automatisches Einspielen von (Security) Patches
- Überwachung des Betriebs hinsichtlich interner und externer Threats und deren Abwehr
- Überwachung sicherheitsrelevanter Konfigurationen
- Hochverfügbarkeit und Disaster Recovery

Hierbei wird in großem Stil Machine Learning eingesetzt, um auch außerhalb eines starren Regelwerks flexibel und automatisiert reagieren zu können.

Die unter Kapitel *Sicherheitstechnologien* (3) dargestellten Technologien können bedingt durch die höhere Automatisierung nur teilweise eingesetzt werden. Eine Beschreibung dazu finden Sie Online und in der Dokumentation der Autonomous Services.

Zugriff auf die komplette
Oracle Dojo-Bibliothek unter
https://tinyurl.com/dojos-online

ORACLE®

Herausgeber: Roland Aussermeier, Oracle Deutschland B.V.

Design: volkerstegmaier.de // Druck: Stober GmbH, Eggenstein

ORACLE®