ANKARA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

YÜKSEK LİSANS TEZİ

BİLGİ SİSTEMLERİ DENETİMİNDE VERİ TABANI SİSTEMLERİNİN İNCELENMESİ

Sercan CANBEY

BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI

ANKARA
2019

TEZ ONAYI

Sercan CANBEY tarafından hazırlanan **"Bilgi Sistemleri Denetiminde Veri Tabanı Sistemlerinin İncelenmesi"** adlı tez çalışması 09/09/2019 tarihinde aşağıdaki jüri tarafından oy birliği ile Ankara Üniversitesi Fen Bilimleri Enstitüsü Bilgisayar Mühendisliği Anabilim Dalı'nda **YÜKSEK LİSANS TEZİ** olarak kabul edilmiştir.

Danışman : Doç. Dr. Recep ERYİĞİT
Ankara Üniversitesi Bilgisayar Mühendisliği Anabilim Dalı

Jüri Üyeleri :

Başkan : Dr. Ögr. Üyesi Mehmet Dikmen
Başkent Üniversitesi, Bilgisayar Mühendisliği Anabilim Dalı

Üye : Doç. Dr. Semra GÜNDÜÇ
Ankara Üniversitesi Bilgisayar Mühendisliği Anabilim Dalı

Üye : Doç. Dr. Recep ERYİĞİT
Ankara Üniversitesi Bilgisayar Mühendisliği Anabilim Dalı

Yukarıdaki sonucu onaylarım.

Prof. Dr. Özlem YILDIRIM
Fen Bilimleri Enstitü Müdürü V.

ETİK

Ankara Üniversitesi Fen Bilimleri Enstitüsü tez yazım kurallarına uygun olarak hazırladığım bu tez içindeki bütün bilgilerin doğru ve tam olduğunu, bilgilerin üretilmesi aşamasında bilimsel etiğe uygun davrandığımı, yararlandığım bütün kaynakları atıf yaparak belirttiğimi beyan ederim.

09/09/2019

Sercan CANBEY

ÖZET

Yüksek Lisans Tezi

BİLGİ SİSTEMLERİ DENETİMİNDE VERİ TABANI SİSTEMLERİNİN İNCELENMESİ

Sercan CANBEY

Ankara Üniversitesi
Fen Bilimleri Enstitüsü
Bilgisayar Mühendisliği Anabilim Dalı

Danışman: Doç. Dr. Recep ERYİĞİT

Teknolojinin hızlı gelişmesi, organizasyonlarda işlenen veri ve bilgi miktarını artırmakta ve kontrol çevresini önemli ölçüde etkilemektedir. Bilgi teknolojileri artık organizasyonların iş stratejileri ve temel iş faaliyetlerinin önemli bir bileşeni haline gelmiştir. Buna göre BT' nin etkin ve verimli yönetimi çoğu organizasyonun başarısı için hayati öneme sahiptir.

Bilişim teknolojilerindeki bu hızlı gelişim ile birlikte veri tabanlarının önemi gün geçtikçe artmaktadır. Bilgiyi kullananlar (çalışanlar, müşteriler), gerçek zamanlı ve hatasız bir şekilde verilere anlık olarak erişebilmeyi talep etmektedir. Ayrıca, veri tabanlarının güvenliğini sağlama konusundaki farkındalık düzeyi, artan yasal ve mevzuata uyumluluk gereklilikleri nedeniyle istikrarlı bir şekilde artmaktadır. Organizasyonlar; mali kayıp, pazardaki olumsuz algı, paydaş değeri kaybı da dahil olmak üzere veri güvenliği ihlalinden kaynaklanabilecek ticari etkinin farkına varmaya devam etmektedir. Bu faktörler, veri tabanı yöneticilerinin, uygulama geliştiricilerinin ve riskin tanımlanmasında değerlendiricinin rolü için daha yüksek güvenlik beklentilerine yol açmıştır. Sonuç olarak, makul bir şekilde öngörülebilir tehditlerden korunmaya karşı uygun güvenlik önlemlerinin bulunduğundan emin olmak için veri tabanının kendisinin denetlenmesine yönelik artan bir odaklanma olmuştur.

Bu çalışmada, veri tabanlarına yapılan saldırılar sonucunda ortaya çıkan skandallardan bahsedilmiş ve bunların ortaya çıkmasına neden olabilecek en sık rastlanan veri tabanı tehditleri ele alınmıştır. Veri tabanlarında bulunan veriler, işletmeler için en değerli varlık olduğundan veri güvenliği konusunun temeli açıklanmıştır. BT genel kontrolleri içerisinde yer alan veri tabanı kontrolleri bu çalışmanın esas konusunu oluşturmuş, sınıflandırılmış ve bu kontrollerin nasıl yapılacağı her kontrol maddesinde anlatılmıştır. Bu çalışma, okuyucuya, veri tabanı güvenliğini denetlemek için pratik, gerçek dünyada karşılaşılan durumlara yönelik yaklaşım sağlamaktadır.

Eylül 2019, 85 sayfa
Anahtar Kelimeler: Bilgi Sistemleri, Bilgi Sistemleri Denetimi, Veri Tabanı Güvenliği, Veri Tabanı Denetimi, Veri Tabanı Kontrolleri, Veri Tabanı İhlalleri

ABSTRACT

Master Thesis

INVESTIGATION OF DATABASE SYSTEMS IN THE AUDITING OF INFORMATION SYSTEMS

Sercan CANBEY

Ankara University
Graduate School of Natural and Applied Sciences
Department of Computer Engineering

Supervisor: Associate Prof. Dr. Recep ERYİĞİT

The rapid development of technology increases the amount of data and information processed in organizations and significantly affects the control environment. Information technology has now become an important component of organizations' business strategies and core business activities. Accordingly, effective and efficient management of IT is vital to the success of most organizations.

With the rapid development of information technologies, the importance of databases is increasing day by day. Users of information (employees, customers) demand instant access to data in real-time and accurate manner. Additionally, the level of awareness of securing databases is steadily increasing due to increased regulatory and regulatory compliance requirements. Organizations continue to recognize the business impact that may result from a breach of data security, including financial loss, negative market perception, loss of stakeholder value. These factors have led to higher security expectations for the role of database administrators, application developers, and assessors in risk identification. Consequently, there has been an increasing focus on auditing the database itself to ensure that appropriate security measures are in place to protect against reasonably foreseeable threats.

In this study, the scandals resulting from the attacks on the databases are mentioned and the most common database threats that may cause them are discussed. And then, the basis of data security is explained. The database controls are the main subject of this study, classified and explained in each control item. This study provides the reader with an approach to practical, real-world situations to control database security.

September 2019, 85 pages
Key Words: Information Systems, Information Systems Auditing, Database Security, Database Auditing, Database Controls, Database Breaches

TEŞEKKÜR

Çalışmamın her aşamasında yardımlarını ve fikirlerini paylaşmayı esirgemeyen ve bu projenin sonuca kavuşturulmasında büyük katkısı olan tez danışmanım Doç. Dr. Recep ERYİĞİT' e (Ankara Üniversitesi Bilgisayar Mühendisliği Anabilim Dalı) en içten duygularımla teşekkür ederim.

Sercan CANBEY
Ankara, Eylül 2019

İÇİNDEKİLER

KISALTMALAR DİZİNİ

ABD	Amerika Birleşik Devletleri
AES	Advanced Encryption Standart
API	Application Programming Interface
BDDK	Bankacılık Düzenleme ve Denetleme Kurumu
BS	Bilgi Sistemleri
BT	Bilgi Teknolojileri
BTK	Bilgi Teknolojileri ve İletişimi Kurumu
CPU	Central Process Unit
DBA	Database Administrator
DCL	Data Control Language
DDL	Data Definition Language
DLL	Dynamic Link Library
DoS	Denial of Service
DML	Data Manipulation Language
FTP	File Transfer Protocol
G/Ç	Girdi/Çıktı
HIPAA	Health Insurance Portability and Accountability Act
IDS	Intrusion Detection System
I/O	Input/Output
IP	Internet Protocol
MSSQL	Microsoft SQL Server
MVC	Model-View-Controller
NoSQL	Not Only Structured Query Language
NTFS	New Technology File System
SQL	Structured Query Language
SSH	Secure Shell
SSL	Secure Sockets Layer
RACF	Resource Access Control Facility
RAM	Random Access Memory
RDBMS	Relational Database Management System

SOX	Sarbanes Oxley Act
TCL	Transaction Control Language
TCP	Transmission Control Protocol
VPN	Virtual Private Network
VTYS	Veri Tabanı Yönetim Sistemi

ŞEKİLLER DİZİNİ

ÇİZELGELER DİZİNİ

1. GİRİŞ

Günümüzde ‘veri’, organizasyonlar için en önemli varlıktır. Günlük yaşantımızda veriler her yerde bulunur. Doğumdan ölüme kadar veri üretir ve tüketiriz. Bankalar için müşteri bilgileri, üniversiteler için öğrenci bilgileri, sosyal medyada kullanılan paylaşımlar, hastaneler için hasta kayıt bilgileri ve yaşamımızda gerçekleşen neredeyse her şey veri olarak adlandırılabilir. Organizasyonlar, bu verilerin gizliliğinden, bütünlüğünden ve erişilebilirliğinden sorumludur. Verileri kullanarak gelir yaratan organizasyonlar, böylece bu verileri ticari varlık olarak vasıflandırabilirler. Düzenleyici kurumlara karşı da bu verilerin kullanımından organizasyonlar sorumludur.

Peki, bu veriler nasıl korunmalıdır? Ya da denetimlerde korunduğu nasıl ispatlanabilir?

Bilgi fiziksel ve mantıksal olarak korunabilir. Giriş-çıkış turnike kapıları, güvenlik görevlileri, biyometrik sistemler fiziksel korumaya örnek gösterilebilir. Mantıksal koruma ise, sunucularda bulunan verilerin son kullanıcıya kadar gelen süreçte geçen her sistem olarak örnek verilebilir.

Veri tabanı teknolojileri birçok bilgi işlem sisteminin temel bileşenidir, en önemli ve en pahalı bilgilerin tutulduğu sistemlerdir. Verilerin elektronik ortamda tutulmasına ve paylaşılmasına izin verir ve bu sistemlerdeki veri miktarı gün geçtikçe artmaya devam etmektedir. Verilerin bütünlüğünü sağlama ve istenmeyen erişimlerden koruma ihtiyacı da buna paralel olarak artmaktadır.

Güvenlik, bilgi teknolojisinde, dijital bilgilerin ve BT varlıklarının iç ve dış, kötü niyetli ve yanlışlıkla olan tehditlere karşı korunmasıdır. Bir organizasyonun korunması gereken en değerli varlıkları arasında şüphesiz veri ve programları en önemli yeri alır. Pek çok organizasyon kritik verisini bilgisayarda saklamakta, işlemekte ve ağ üzerinden diğer bilgisayarlara aktarmaktadır. Gizli bilgilerin korunması bir iş zorunluluğu ve yasal bir gerekliliktir.

Bazı kurumların çıkardığı yasal zorunluluklar nedeniyle (BDDK, BTK, SOX gibi) verilere erişimlerin yetkilendirilmesi, bunların kayıt altına alınması gerekmektedir.

Bunun yanında, mobil sistemler, bulut bilişim, yapay zeka uygulamaları, sosyal medya yönetimi, iş sürekliliği gibi güncel kavramlar, organizasyonlar için yüksek risk taşımakta ve BT denetimlerini daha önemli hale getirmektedir. Buna göre, organizasyonların bilgi teknolojilerinden kaynaklanan riskleri belirleyebilmesi, iç kontrollerinin etkinliğini değerlendirebilmesi ve finansal denetimlerin daha anlamlı sonuçlar üretebilmesi amacıyla BT denetimlerinden yararlanılmaktadır.

Organizasyonlar, bağımsız denetim şirketleriyle sözleşme imzalayarak kendi yaptıkları işlemlerin doğruluğunu ve bu işlemlerin kontrolünü denetim kanıtları sunarak ispatlarlar. Başarılı bir denetimin en önemli başlangıç noktası da ilgili teknik işlemlere koyulan kontrol noktalarının denetim kanıtı olarak sunulmasıdır.

Organizasyonlar, denetimlerde sorun yaşamamak adına gerekli kontrolleri devreye alır ve kendi öz değerlendirmelerini yapar. Bilgi sistemleri kontrollerinin bir parçası olan veri tabanı kontrolleri bu çalışmanın kapsamını oluşturmaktadır.

Veri tabanı kavramının ve öneminin anlaşılması, veri tabanında bulunan verilerin öneminden dolayı doğabilecek riskleri makul düzeye indirgemek için uygulanması gereken kontrollerin belirlenmesi, veri tabanlarına ilişkin bir denetimde, denetçinin hangi konulara bakması gerektiğini bilmesi bu çalışmanın amacını oluşturmaktadır.

Diğer taraftan bilgi sistemlerine ilişkin denetimlerde veri tabanlarına yönelik hali hazırda bir yönetmelik ve benzeri bir çalışma yayımlanmamış olması nedeniyle bu tez bu alanda yapılacak çalışmalara bir çerçeve oluşturması açısından önem arz etmesi beklenmektedir.

İlk olarak, ikinci bölümde denetim eksikliğinden veya gerekli güvenlik önlemlerinin alınmamasından kaynaklanan, yaşanan dünya çapındaki veri ihlalleri olaylarından

bahsedilecektir. Üçüncü bölümde en sık rastlanan veri tabanı güvenlik tehditleri anlatılacak olup dördüncü bölümde veri güvenliği konusuna değinilecektir. Çalışmanın esas konusu olan veri tabanı denetimi kontrollerine geçilmeden önce beşinci bölümde BT denetimi hakkında bilgi verilecek olup, veri tabanlarının BT denetimindeki yeri anlatılacaktır. Altıncı bölümde denetçinin veri tabanı ile ilgili bilmesi gereken genel konular anlatılacak, veri tabanı ile ilgili kontroller sınıflandırılıp kontrol seti halinde verilecek ve her birinin nasıl yapıldığı açıklanacaktır.

2. YAŞANAN EN BÜYÜK VERİ İHLALLERİ

ABD'de kredi raporlama kurumu olan Equifax, yaşanan veri ihlali nedeniyle yaklaşık 143 milyon kişinin bilgilerinin sızdırıldığını açıklamıştır ki bu sayı neredeyse ABD nüfusunun yarısını oluşturmaktadır. Sızdırılan bilgiler arasında kredi kartı numaraları, sosyal güvenlik numaraları, doğum tarihleri gibi detaylı kişisel bilgiler bulunmaktadır. Yaşanan veri ihlalinin açıklanmasının ardından şirketin hisse değerleri büyük oranda düşmüş ve şirket büyük bir itibar kaybına uğramıştır (tom's guide, 2018).

İnternet servisi Yahoo da, son yıllarda yaşadığı siber saldırı olaylarının üzerine inanılmaz hızlı değer kayıpları yaşamıştır. Skandal yaratan siber saldırı olaylarının ardından bir dizi işten çıkarma ve yeniden yapılanmaya giden Yahoo, yine de büyük mali kaybına engel olamamıştır. İnternet tarihinde en büyük siber saldırılardan birine maruz kalan ve bir milyardan fazla kullanıcı bilgisini kaptıran Yahoo, neredeyse tüm veri tabanını kaptırmıştır (tom's guide, 2018).

Sosyal ağ siteleri olan Myspace ve Tumblr kullanıcılarına ait bilgiler çalınmış ve yaklaşık 360 milyon hesap ele geçirilmiştir. Kullanıcı adı ve parolalar 2013 yılında çalınmış ancak 2016 yılında çalındığı ortaya çıkmıştır. Bu olaylardan sonra bu siteler popülerliğini ve güvenilirliğini kaybetmiştir (tom's guide, 2018).

İş dünyasında bulunan kişilerin iletişim kurmasını sağlayan sosyal medya platformu olan Linkedin' de de daha önce benzer bir durum meydana gelmiş ve yaklaşık 160 milyon kullanıcının hesabı internet ortamında satışa sunulmuştur (tom's guide, 2018).

Bu zamana kadarki en büyük kredi kartı bilgilerinin çalınması olaylarından biri de, ödeme sistemleri alanında çalışan Heartland şirketinin başına gelmiştir. Yaklaşık 130 milyon kart numarası ve diğer hesap bilgileri ele geçirilmiş, Heartland şirketi bu saldırıdan etkilenen Visa, Mastercard, American Express ve başka şirketlere toplamda 110 milyon dolardan daha fazla bir para ödemek zorunda kalmıştır (tom's guide, 2018).

2013’ ün sonlarında ABD’ de bulunan mağaza zinciri Target, 110 milyon müşterisinin hesaplarının çalındığını açıklamıştır. Target, olabilecek en kötü zamanda, tatil alışverişi döneminde bu siber saldırısıyla sarsılmış, bilgisayar korsanları, o dönemde Target için ısıtma ve havalandırma hizmeti veren firmadan bilgisayar oturumlarını ele geçirerek Target’ in ödeme sistemini kontrol etmiştir. Bilgisayar korsanları saldırı sonucunda 40 milyon kredi kartı numarasını ve 70 milyon hesaptan kişisel bilgileri ele geçirmiştir (tom's guide, 2018).

Ülkemizde ise bankacılık sektöründe bir skandal yaşanmış, bunun sonucunda İmar Bankası’nın faaliyetleri, BDDK tarafından 2003 yılında durdurulmuştur. Bu olayda, yönetim tarafından tek hesap üzerinden çifte kayıt yapılması istenmiş, yani bankaya para yatıran müşterilerin hesapları ikinci bir veri tabanında tutulmuş, banka yönetimi tarafından bu hesaplar istenildiği gibi kullanılabilir hale getirilmiştir (YILDIRIM, 2017)

2001 yılında dünya genelinde büyük yankı uyandıran Enron skandalı ve 2003 yılında ülkemizde yaşanan İmar Bankası skandalı ile denetim kavramı daha önemli hale gelmiş, BT denetiminin de finansal denetimdeki yeri kaçınılmaz hale gelmiştir.

Yaşanan bu skandallar sonucunda işletmeler bilgi sistemleri güvenliğine ve buna bağlı olarak bilgi sistemleri denetimine önem vermesi gerektiğini anlamışlardır. Yukarıda bahsedilen organizasyonlar sadece yüksek miktarda değer kaybı yaşamamış aynı zamanda kendilerine olan güven de büyük ölçüde sarsılmıştır. Bilgi sistemlerinin bir parçası olan veri tabanları denetiminin önemi, yaşanan bu skandalların, veri tabanında bulunan verilerin açığa çıkmasıyla daha çok anlaşılmıştır. Bu denetimler için uyulması gereken kontrollerle ve alınması gereken güvenlik önlemleriyle yaşanacak veri kaybı en az seviyeye indirilecektir fakat tüm güvenlik önlemleri alınsa bile bilgisayar korsanları yeni yollar bulup veri hırsızlığına devam edecektir. Yeni teknolojilerle birlikte yeni güvenlik açıkları ortaya çıkmaktadır.

3. EN SIK RASTLANAN VERİ TABANI GÜVENLİK TEHDİTLERİ

Veri tabanları, gizliliği en çok ihlal edilen varlıkların başında gelmektedir. Veri tabanlarının çok sık hedef alınmasının sebebi oldukça basittir: müşteri kayıtlarını ve diğer gizli olan ticari verileri depolayan herhangi bir organizasyonun kalbidir. Fakat şu soru aklımıza gelebilir; veri tabanları ihlallere karşı neden çok savunmasızdır?

Bunun nedenlerinden biri, kuruluşların bu önemli varlıkları yeterince iyi koruyamamasıdır. ABD merkezli market araştırmaları şirketi IDC' ye göre, güvenlik ürünlerinde harcanan 27 milyar doların %5'inden azı doğrudan veri merkezlerine yöneliktir (IDC, 2011).

Bilgisayar korsanları ve kötü niyetli kullanıcılar hassas verilere eriştiklerinde, verileri hızla çıkarabilir, onlara zarar verebilir veya ticari işlemlere etki edebilir. Mali kayıp veya itibar kaybına ek olarak ihlaller; düzenleyici uyumsuzluklara, para cezalarına ve yasal yaptırımlara neden olabilir. Bununla birlikte, 2013 yılında OTA(Online Trust Alliance)'ya göre olayların %97'den fazlası basit adımlar uygulayarak, iç kontrol ve en iyi uygulamaları takip ederek engellenebilmektedir (Imperva, 2015).

Veri merkezi güvenliği alanında çözümler sunan Imperva firması tarafından tespit edilen en önemli 10 veri tabanı tehdidi aşağıdaki maddelerde verilmiştir (Imperva, 2015).

3.1 Aşırı ve Kullanılmayan Yetkiler

Veri tabanı kullanıcılarına gerektiğinden fazla erişim yetkisi verildiği zaman bu yetkiler kullanıcılar tarafından suistimal edilebilir. Örneğin, bir banka çalışanı sadece hesap sahibinin iletişim bilgilerini değiştirebilirken aşırı yetkiye sahip olduğunda arkadaşının hesap bakiyesini değiştirme olanağı bulur. Ayrıca, bir çalışanın organizasyon içerisinde rolü değiştiğinde veya işi bıraktığında, hassas verilere erişim hakkının alınması genelde unutulur. Çalışanın işten kötü şartlar altında ayrılması durumunda, eski haklarını

kullanarak verileri açığa çıkarabilir.

İş rolleri için yetki kontrol mekanizmaları iyi tanımlanmazsa veya yönetilmezse, kullanıcılar aşırı yetkiye sahip olabilir. Sonuç olarak, kullanıcılar belirli iş gereksinimlerini aşan genel veya varsayılan erişim ayrıcalıklarına sahip olabilir, bu durum da gereksiz risk oluşturur.

Ayrıcalıkların kötüye kullanımına karşı alınacak önlemler şu şekilde olabilir;

- Erişim kontrol politikası ile kullanıcıya gereksiz ayrıcalıklar verilmez
- Meşru ayrıcalıkların kötüye kullanımı, iyi bir denetim izi sağlayarak durdurulabilir.

3.2 Yetki Suistimali

Kullanıcılar veri tabanı yetkilerini izinsiz amaçlar için kullanabilirler. Bir web ara yüzü aracılığıyla bireysel hasta kayıtlarını görüntülemek için kullanılan bir sağlık uygulaması düşünelim. Böyle bir web uygulaması normalde kullanıcıları, hastaların sağlık geçmişini görüntülemesini sınırlar. Çoklu hasta kayıtları aynı anda görüntülenemez ve elektronik kopyalarının alınmasına izin verilmez. Fakat, veri tabanına Microsoft Excel gibi farklı istemciler kullanılarak bağlanılması mümkün olabilmektedir. Yeterli kısıtlamanın olmadığı bir ortamda böyle bir bağlantı olması durumunda kötü niyetli bir kullanıcı hasta kayıtlarının tümünü görüntüleyip istediği işlemi yapabilir.

3.3 SQL Enjeksiyonu

Veri tabanı sistemleri uygulamanın arka tarafındaki işleri yapmak için kullanılır. Kullanıcı tarafından sağlanan verilerin girişi genellikle doğrudan veri tabanına etki eden SQL ifadelerinin dinamik olarak oluşturulması için kullanılır. Giriş(input) enjeksiyonu, saldırgan tarafından uygulamanın gerçek amacını ortadan kaldırmaya yönelik bir saldırı türüdür.

İki temel veri tabanı enjeksiyon saldırısı türü vardır:

- Geleneksel veri tabanı sistemlerini hedefleyen SQL enjeksiyonu
- Büyük Veri platformlarını hedef alan NoSQL enjeksiyonu

SQL enjeksiyon saldırıları genellikle web uygulamalarının girdi alanlarına yetkisiz veya kötü amaçlı ifadeler eklemeyi (veya 'enjekte etme') içerir. Öte yandan, NoSQL enjeksiyon saldırıları, Büyük Veri bileşenlerine (örneğin, Hive veya MapReduce) zararlı ifadeler eklemeyi içerir.

Her iki türde, başarılı bir girdi enjeksiyonu saldırısı, saldırgana veri tabanı için sınırsız bir erişim hakkı verebilir. Büyük Veri çözümlerinin SQL enjeksiyon saldırılarından etkilenmemesi teknik olarak doğru olduğu halde – çünkü herhangi bir SQL tabanlı teknoloji kullanmaz – aslında, aynı temel saldırı sınıfından etkilenirler (Pawar, 2015).

Bu saldırı türüyle yapılabilecek zararlı girişimler şunlardır;

- Veri tabanında istenmeyen sorguların yazılabilmesi(silme, değiştirme, ekleme vb.)
- Uygulama üzerinde bulunan kimlik doğrulamanın atlatılabilmesi
- İşletim sistemi üzerinde bazı komutların çalıştırılabilmesi
- Veri tabanında yeni kullanıcılar oluşturulabilmesi.

SQL enjeksiyonuna karşı alınacak önlemler şu şekilde olabilir;

- Direkt olarak sorguların yazılması yerine saklı yordamlar kullanılır
- MVC gibi mimariler uygulanır

3.4 Kötü Niyetli Yazılımlar

Bilgisayar korsanları organizasyonlara sızmak ve hassas verileri çalmak için gelişmiş

taktikler uygular. (örneğin, eposta oltalama ve zararlı yazılım). Kötü niyetli yazılımlar (malware) kullanıcıların farkında olmadan bilgisayarlara bulaşıp, normal kullanıcılar aracılığıyla ağa ve hassas verilere erişim sağlarlar.

Kötü niyetli yazılımlara karşı alınacak önlemler şu şekilde olabilir;

- Güvenlik duvarı koruması aktif edilir ve antivirüs programı yüklenir.

3.5 Zayıf Denetim İzi

Hassas verilerin yer aldığı veri tabanı işlemlerinin otomatik olarak kaydedilmesi herhangi bir veri tabanı sisteminin parçası olmalıdır. Veri tabanında faaliyet izlemeye ilişkin ayrıntılı denetim kayıtları toplamakta zayıf kalmak, birçok düzeyde ciddi organizasyonel risklere neden olur.

Zayıf (veya bazen var olmayan) veri tabanı denetleme mekanizmalarına sahip organizasyonlar düzenleyici gerekliliklerle ters düşmeye başlayacaktır. Muhasebe hatalarına ve hileli uygulamalara karşı koruma sağlayan Sarbanes-Oxley(SOX) ve sağlık sektöründe Sağlık Bilgi Taşınabilirliği ve Hesap Verebilirlik Yasası(HIPAA), veri tabanı denetim gereksinimlerini açıkça içeren düzenlemelere örnek olarak verilebilir. Birçok işletme, veri tabanı sağlayıcıları tarafından sağlanan entegre denetim araçlarına yönelmekte veya geçici ve manuel çözümlere güvenmektedir. Bu yaklaşımlar, denetimi, saldırı tespitini ve adli olayları destekleyen gerekli ayrıntıları kaydetmezler. Dahası, bu entegre veri tabanı denetim mekanizmaları işlemci ve disk kaynaklarını tükettiğinden işletmeler genelde kullanmaya sıcak bakmamaktadır. Çoğu entegre denetim mekanizması veri tabanı platformuna özgüdür. Örneğin, Oracle logları MSSQL'den farklı, MSSQL logları DB2 veri tabanı loglarından farklıdır. Birden çok veri tabanı ortamlarına sahip kuruluşlar için bu durum, tekdüze ve ölçeklenebilir denetim süreçlerini uygulamada engel teşkil eder.

Kullanıcılar veri tabanına kurumsal web uygulamaları (SAP, Microsoft Dynamics,

PeopleSoft, Oracle E-Business Suit gibi) ile eriştiğinde, hangi veri tabanı erişim faaliyetinin belirli bir kullanıcıyla ilişkili olduğunu anlamak zor olabilir. Çoğu denetim mekanizması, son kullanıcının kimliğini tespit edemez çünkü tüm faaliyetler, web uygulaması hesabı adıyla ilişkilendirilir. Sorumlu olan kullanıcıyla bağlantı olmadığından raporlama, görünürlük ve adli analiz güçleşmektedir.

Sonuç olarak, yasal olarak veya kötü niyetle veri tabanına yönetici olarak erişimi olan kullanıcılar, hileli faaliyetleri gizlemek için entegre veri tabanı denetimini kapatabilir. Görevler ayrılığının sağlanması için denetim becerileri ve sorumlulukları ideal olarak hem veri tabanı yöneticileri hem de veri tabanı sunucusu platformundan ayrılmalıdır.

Zayıf denetim izine karşı alınacak önlemler şu şekilde olabilir;

- Ağ bazlı denetim araçları iyi bir çözümdür. Bu tür cihazlar veri tabanı performansını etkilememeli, tüm kullanıcılardan bağımsız çalışmalı ve ayrıntılı veri toplama sunmalıdır.

3.6 Depolama Ortamı İfşası

Yedekleme depolama ortamları genellikle saldırılardan korunmasız şekilde tutulmaktadır. Bunun sonucunda, veri tabanı yedekleme diskleri ve kasetlerinde çok sayıda güvenlik ihlali ortaya çıkmaktadır. Hassas bilgilere düşük seviyede erişime sahip yönetici aktivitelerinin izlenmesinde ve denetlenmesinde yetersiz kalmak da verileri riskli duruma düşürebilir. Hassas verilerin yedek kopyalarını korumak ve ayrıcalıklı kullanıcıları izlemek için uygun önlemleri almak sadece veri güvenliği için en iyi uygulama değil aynı zamanda birçok düzenleme tarafından zorunlu kılınmıştır.

Depolama ortamı ifşasına karşı alınacak önlemler şu şekilde olabilir;

Veri tabanı şifreleme: veriler şifrelenmiş şekilde saklanır. Bu durum, veri tabanlarının hem üretim hem de yedek kopyalarını güvenli hale getirmeyi sağlar. Daha sonra, işletim

sistemi ve depolama katmanlarında, veri tabanına kimin eriştiği konusunda aktiviteler denetlenir. Organizasyonlar, veri tabanı denetimini (audit) şifreleme ile birlikte kullanarak hem veri tabanı içindeki hem de dışındaki kullanıcıları izleyebilir ve kontrol edebilir.

3.7 Zayıf, Yanlış Yapılandırılmış Veri Tabanlarının Suistimali

Güvenlik açığı olan ve yamalanmamış veri tabanlarını bulmak veya varsayılan hesaplara ve yapılandırma parametrelerine sahip veri tabanlarını ortaya çıkarmak yaygın bir yöntemdir. Saldırganlar, kuruluşa yönelik saldırılar başlatmak için bu güvenlik açıklarını nasıl kullanacaklarını bilirler. Ne yazık ki kuruluşlar, yamalar mevcut olduğunda bile veri tabanı yapılandırmalarını sürdürmekte sıkıntı yaşarlar. Yüksek iş yükü ve ilgili veri tabanı yöneticileri için bekleme günlükleri (mounting backlogs), yamaları test etmek için karmaşık ve zaman alan gereksinimler gibi tipik sorunlar mevcuttur. Genelde veri tabanlarının yama işlemlerinin uygulanması aylarca sürdüğü için bu zamanlar veri tabanları savunmasız kalmaktadır.

2014 Bağımsız Oracle Kullanıcı Grubu (IOUG) Kurumsal Veri Güvenliği Anketi'ne göre, Oracle kullanıcılarının yüzde 36'sı kritik yama güncellemelerini uygulamış ve altı aydan fazla zaman almışken, yüzde 8'i hiç uygulamamıştır (McKendrick, 2014).

Yanlış yapılandırılmış veri tabanlarına karşı alınacak önlemler şu şekilde olabilir;

- Varsayılan hesaplar bulunmamalıdır. Hesaplar yeni kullanıcı adı ve şifre kullanılarak oluşturulmalıdır.
- Güncel yamalar zamanında ve prosedüre uygun olarak uygulanmalıdır.

3.8 Yönetilmeyen Hassas Veriler

Birçok şirket, veri tabanlarının hatasız bir envanterini ve bunlarda bulunan kritik veri nesnelerini yönetmek için mücadele etmektedir. Unutulan veri tabanları hassas bilgiler

içerebilir ve yeni veri tabanları güvenlik ekibine görünmeden ortaya çıkabilir – örneğin, uygulama test ortamlarında-. Bu veri tabanlarındaki hassas veriler; gerekli kontroller ve izinler uygulanmazsa tehditlere maruz kalmaktadır.

Yönetilmeyen hassas verilere karşı alınacak önlemler şu şekilde olabilir;

- Veri tabanında bulunan hassas veriler şifrelenir.
- Gerekli kontroller ve izinler veri tabanına uygulanmalıdır.

3.9 Hizmet Durdurma Saldırısı (DoS)

Hizmet dışı bırakma (DoS), ağ uygulamalarına veya verilere erişimin, amaçlanan kullanıcılara engellenmesi ile gerçekleştirilen genel bir saldırı türüdür. DoS koşulları birçok teknikle oluşturulabilir. Veri tabanı ortamlarında kullanılan en yaygın teknik, bellek ve CPU gibi sunucu kaynaklarını aşırı miktarda sorgu ile doldurarak veya orantısız miktarda sistem kaynağı tüketen iyi hazırlanmış sorgularla yapılabilir. Her iki durumda da sonuç aynıdır, kaynak yetersizliği olan sunucular yanıt vermez ve bazı durumlarda çökebilir. DoS saldırılarının ardındaki nedenler, genellikle dolandırmaya zorlamaktır ki uzaktaki saldırgan, talepleri yerine gelene kadar sunuculara tekrar tekrar saldırarak sunucuları çökertme işlemi yapar. DoS saldırıları, kullanıcılar, organizasyonlar ve internet altyapıları için ciddi bir tehdit oluşturmakta ve organizasyonların finansal istikrarına ve itibarına zarar verebilmektedir.

Hizmet durdurma saldırısına karşı alınacak önlemler şu şekilde olabilir;

- TCP/IP belleği artırılır. TCP bağlantı kuyruğu boyutu artırılarak, bağlantı kurulum aralıkları azaltılarak, dinamik bekleyen mekanizmaları kullanarak bağlantı sırası hiç bitmemesi sağlanır.
- Saldırı tespit sistemi (IDS) kullanılır çünkü bu sistemler saldırıları otomatik olarak algılar ve yanıt verir.

3.10 Sınırlı Güvenlik Uzmanı ve Eğitim

İç güvenlik kontrolleri veri büyümesine ayak uyduramamaktadır ve birçok kuruluş güvenlik ihlali ile uğraşacak yeterli donanıma sahip değildir. Genellikle bunlar, güvenlik kontrollerini, politikaları uygulayan uzmanlığın eksik olmasından kaynaklanmaktadır.

Ponemon Enstitüsü Veri İhlali Çalışması 2014 Maliyetleri verilerine göre, veri ihlali olaylarının yüzde 30'unda ana sebep 'insan faktörü' olarak sınıflandırılmıştır – diğer bir deyişle, ihmalkar bir çalışan ya da yüklenici- (2014 Cost of Data Breach Study, 2014).

Bu soruna karşı alınacak önlemler şu şekilde olabilir;

- Kullanıcı eğitimi ve farkındalığı
- Deneyimli güvenlik uzmanı yetiştirilmesi

4. VERİ GÜVENLİĞİ

Bilgi sistemi güvenliği, bir bilgi sisteminin ve onun ana varlığı olan verinin, gizliliğini, bütünlüğünü ve erişilebilirliğini temin eden faaliyet ve önlemleri ifade etmektedir. Verilerin güvenliği, organizasyon genelinde kapsamlı bir yaklaşım gerektirir. Diğer bir deyişle, donanım sistemleri, yazılım uygulamaları, ağ ve aygıtları, iç ve dış kullanıcılar, prosedürler ve verilerin kendisi de dahil olmak üzere tüm işlemler ve sistemler güvence altına alınmazsa, veriler güvenli olamaz. Veri güvenliğinin kapsamını anlamak için üç güvenlik hedefinin her birini daha ayrıntılı olarak düşünelim:

Gizlilik, verilerin yetkisiz erişime karşı korunmasını sağlar ve veriye yetkili bir kullanıcı tarafından erişilirse, yalnızca yetkili bir amaçla kullanılır. Başka bir deyişle, gizlilik, bir kişinin veya bir organizasyonun özel haklarını ihlal edecek herhangi bir bilginin ifşa edilmesine karşı verilerin korunmasını gerektirir. Veri gizlilik düzeyine göre değerlendirilmeli ve sınıflandırılmalıdır: çok kısıtlı (çok az kişi erişime sahip), gizli (yalnızca belli gruplara erişim var), sınırsız (tüm kullanıcılar tarafından erişilebilir). Veri güvenliği görevlisi, organizasyonun istenilen düzeyde gizlilikle uyumlu olmasını sağlamak için çok zaman harcar. Uyumluluk, veri mahremiyeti ve güvenlik raporlama yönergelerini karşılayan faaliyetleri ifade eder. Bu yönergeler, ya iç prosedürlerin bir parçasıdır ya da dış düzenleyici kurumlar tarafından uygulanmaktadır. Veri mahremiyetini ve gizliliğini sağlamak için ABD yasalarına örnek olarak Sağlık Sigortası Taşınabilirlik ve Sorumluluk Yasası (HIPAA), Gramm-Leach-Bliley Yasası (GLBA) ve Sarbanes-Oxley Yasası (SOX) verilebilir (Coronel & Morris, 2016, s. 629).

Bütünlük, veri güvenliği çerçevesinde, verilerin tutarlı ve hatadan veya anormallikten uzak tutulmasıyla ilgilidir. Veri tabanı yönetim sistemleri, veri tabanındaki verilerin bütünlüğünü sağlamada çok önemli bir rol oynamaktadır. Bununla birlikte, güvenlik açısından bakıldığında, organizasyonel süreçler, kullanıcılar ve kullanım düzenleri de bütünlüğü korumalıdır. Örneğin, evde çalışan biri interneti kullanarak ürünün maliyetine erişmesi kabul edilebilir bir kullanım olarak düşünülebilir fakat güvenlik standartları, evdeki verileri yönetmek için çalışanın güvenli bir bağlantı kullanmasını ve katı prosedürleri izlemesini (mesela, verilerin yerel sabit sürücüye kopyalanması için

şifreleme) gerektirebilir. Veri bütünlüğünün korunması, veri toplamayla başlayan ve veri depolama, işleme, kullanım ve arşivleme ile devam eden bir süreçtir. Bütünlük temelinin ardındaki mantık, veriyi organizasyondaki en değerli varlık olarak ele almak ve özenli veri doğrulamasının organizasyon içindeki tüm düzeylerde gerçekleştirilmesini sağlamaktır (Coronel & Morris, 2016, s. 629).

Kullanılabilirlik, yetkili kullanıcılar tarafından ve yetkili amaçlar için gerekli olduğunda erişilebilirliği ifade eder. Veri kullanılabilirliğini sağlamak için sistemin tümü harici ve dahili kaynakların neden olduğu hizmet kaybından ve kesintilerden korunmalıdır. Hizmet kesintileri, şirketler ve kullanıcılar için çok pahalı olabilir. Sistem kullanılabilirliği, güveliğin önemli bir hedefidir (Coronel & Morris, 2016, s. 629).

4.1 Güvenlik Politikaları

Normal olarak sistemi ve verileri güvence altına alma görevi, tutarlı bir veri güvenliği stratejisi oluşturmak üzere birlikte çalışan veri tabanı güvenlik sorumlusu ve veri tabanı yöneticisi tarafından gerçekleştirilir. Böyle bir strateji, güvenilir ve kapsamlı bir güvenlik politikası tanımlamayla başlar. Güvenlik politikası, bir sistemin güvenliğini garanti etmek ve denetimi, uyumluluğunu sağlamak için oluşturulan standartlar, politikalar ve prosedürlerin toplamıdır. Güvenlik denetim süreci, kuruluşun bilgi sistemi altyapısındaki güvenlik açıklarını belirleyerek ve bu güvenlik açıklarına karşı sistemi ve verileri koruyacak tedbirleri belirleyerek başlamaktadır.

4.2 Güvenlik Açıkları

Güvenlik açığı, yetkisiz erişime izin vermek veya hizmet kesintilerine neden olmak için kullanılabilecek bir sistem bileşeni zayıflığıdır. Bu güvenlik açıkları aşağıdaki kategorilerden birine girebilir:

Teknik: Örnek olarak işletim sisteminde veya web tarayıcıda bir hata

Yönetimsel: Örneğin, bir organizasyonun, kullanıcıları kritik güvenlik konuları hakkında eğitmemesi

Kültürel: Kullanıcılar klavyeleri altında şifreleri gizleyebilir veya gizli raporları imha etmeyi unutabilir.

Prosedürel: Şirket prosedürleri, karmaşık şifreler veya kullanıcı kimlik kontrolü gerektirmeyebilir.

Bir güvenlik açığı kontrol edilmeden bırakıldığında güvenlik tehdidi haline gelebilir. Bir güvenlik tehdidi olası bir güvenlik ihlalidir.

Bir güvenlik ihlali, sistemin gizliliğini, bütünlüğünü veya kullanılabilirliğini tehlikeye atacak bir güvenlik tehdidi olduğunda ortaya çıkar. Güvenlik ihlalleri, bütünlüğü korunmuş veya bozulmuş bir veri tabanına sebep olabilir:

Korunmuş: Bu gibi durumlarda, benzer güvenlik sorunlarının tekrarından kaçınmak için harekete geçilmesi gerekir ancak veri kurtarma gerekli olmayabilir. Nitekim, çoğu güvenlik ihlali, bilgi amaçlı yetkisiz ve fark edilmeyen erişim yoluyla ortaya çıkar ancak bu durum veri tabanını bozmaz.

Bozulmuş: Benzer güvenlik sorunlarının tekrarından kaçınmak için işlem yapılması gerekli olur ve veri tabanı tutarlı bir duruma getirilmelidir. Güvenlik ihlallerinin bozulması şunları içerir; bilgisayar virüsleri ve verileri yok etmek veya değiştirmek isteyen bilgisayar korsanları tarafından veri tabanına erişim (Coronel & Morris, 2016, s. 630).

4.3 Veri Tabanı Güvenliği ve Yöneticisi

Veri tabanı güvenliği, kuruluşun güvenlik gereksinimlerine uyan VTYS özelliklerine ve diğer ilgili önlemlere atıfta bulunur. Veri tabanı yöneticisinin bakış açısından, VTYS' yi

hizmet kesintisine karşı korumak ve veri tabanını kaybetme, bozulma veya hatalı kullanıma karşı korumak için güvenlik tedbirleri uygulanmasıdır. Kısacası, veri tabanı yöneticisi, veri tabanını kurulum noktasından işleyiş ve bakım işlemlerine kadar sağlamlaştırmalıdır.

James Martin'e göre güvenlik stratejisi, veri tabanı güvenliğinin yedi temel unsurunu temel almaktadır; verilerin korunmuş, yeniden yapılandırılabilir, denetlenebilir, dayanıklı ve kullanıcıların tanımlanabilir, yetkili ve izlenir olmasıdır (Martin, 1977).

VTYS'yi hizmet bozulmalarına karşı korumak için bazı güvenlik önlemleri önerilir (Coronel & Morris, 2016, s. 631-632). Bunlar;

- Varsayılan sistem parolaların değiştirilmesi
- Varsayılan yükleme yollarının değiştirilmesi
- En son yamaların uygulanması
- Yükleme klasörlerini uygun erişim haklarıyla güvenli hale getirme
- Sadece gerekli servislerin çalıştığından emin olma
- Denetim loglarının kurulması
- Oturum loglarının kurulması
- Oturum şifrelemesi gerekliliği

Dahası, veri tabanı yöneticisi, VTYS' yi ve ağda çalışan tüm hizmetleri koruyan ağ güvenliğini uygulamak için ağ yöneticisi ile yakından çalışmalıdır. Modern organizasyonlarda, bilgi mimarisinde en kritik bileşenlerden biri ağdır.

Veri tabanındaki verilerin korunması, yetkilendirme yönetiminin bir fonksiyonudur. Yetkilendirme yönetimi, veri tabanı güvenliği ve bütünlüğünü koruma ve garanti etme prosedürlerini tanımlar. Bu prosedürler şunları içerir:

Kullanıcı erişim yönetimi: Bu işlev, veri tabanına erişimi sınırlamak için tasarlanmıştır; en azından aşağıdaki prosedürleri içerir:

Her kullanıcıyı veri tabanına tanımlama: Veri tabanı yöneticisi, bu işlevi işletim sistemi düzeyinde ve VTYS düzeyinde gerçekleştirir. İşletim sistemi düzeyinde, veri tabanı yöneticisi, bilgisayar sisteminde oturum açan her son kullanıcı için benzersiz bir kullanıcı kimliği oluşturulmasını isteyebilir. VTYS düzeyinde, veri tabanı yöneticisi, ya farklı bir kullanıcı kimliği oluşturabilir ya da VTYS' ye erişmek için aynı kimliği kullanabilir.

Her kullanıcıya şifre atanması: Veri tabanı yöneticisi, bu işlevi de hem işletim sistemi hem de VTYS düzeylerinde gerçekleştirir. Veri tabanı şifrelerinin kullanım sürelerinin önceden belirlenip son kullanıcılara periyodik olarak hatırlatılması yetkisiz girişleri daha az seviyeye indirmeyi sağlamaktadır.

Kullanıcı gruplarının tanımlanması: Kullanıcıları ortak erişim ihtiyaçlarına göre gruplar halinde sınıflandırmak, veri tabanı yöneticisine bireysel kullanıcıların erişim ayrıcalıklarını kontrol etme ve yönetmesine yardımcı olabilir. Ayrıca, veri tabanı yöneticisi, sahte kullanıcıların sistemdeki etkisini en aza indirgemek için veri tabanı rollerini ve kaynak sınırlarını kullanabilir.

Erişim ayrıcalıklarının atanması: Veri tabanı yöneticisi, veri tabanlarına erişmek için belirli kullanıcılara erişim ayrıcalıkları atar. Erişim hakları, salt okunur olabilir veya okuma, yazma ve silme ayrıcalıklarını içerebilir. İlişkisel veri tabanlarındaki erişim ayrıcalıkları, SQL GRANT ve REVOKE komutlarıyla atanır.

Fiziksel erişimi kontrol etme: Fiziksel güvenlik, yetkisiz kullanıcıların VTYS kurulumuna ve araç gereçlerine doğrudan erişmesini önleyebilir. Büyük veri tabanları kurulumları için ortak fiziksel güvenlik; güvenli girişler, parola korumalı iş istasyonları, elektronik personel kimlik kartları, ses tanıma ve biyometrik teknolojilerini içerir. (Coronel & Morris, 2016, s. 632)

Görünüm (view) tanımı: Veri tabanı yöneticisi, yetkili bir kullanıcı tarafından erişilebilen verilerin kapsamını korumak ve kontrol etmek için veri görünümlerini

tanımlamalıdır. VTYS, bir veya daha fazla tablodan oluşan görünümlerin tanımlanmasına izin veren ve kullanıcılara erişim hakları atanması gereken araçları sağlamalıdır. SQL CREATE VIEW komutu ilişkisel veri tabanlarında görünümleri tanımlamak için kullanılır. Oracle, veri tabanı yöneticisinin farklı kullanıcılara yönelik, verileri kişiselleştirilmiş görünümler oluşturmasına olanak tanıyan Sanal Özel Veri Tabanı (VPD)' nı sunar. Bu özellikle, veri tabanı yöneticisi, standart bir kullanıcının sadece gerekli satır ve sütunları görmesini sağlayabilir.

VTYS erişim kontrolü: Veri tabanı erişimi, VTYS sorgu ve raporlama araçlarının kullanımına sınır koyarak kontrol edilebilir. Veri tabanı yöneticisi, araçların doğru ve yalnızca yetkili personel tarafından kullanılmasını sağlamalıdır.

VTYS görüntüleme: Veri tabanı yöneticisi veri tabanındaki verilerin kullanımını da denetlemelidir. Çeşitli VTYS paketleri, tüm kullanıcılar tarafından gerçekleştirilen veri tabanı işlemlerinin kısa bir açıklamasını otomatik olarak kaydeden bir denetim logunun oluşturulmasına izin veren özellikler içerir. Bu denetim izleri, veri tabanı yöneticisinin erişim ihlallerini tespit etmesine yardımcı olur. Denetim izleri, tüm veri tabanı erişimlerini veya sadece başarısız olanları kaydetmek için ayarlanabilir.

Bir veri tabanının bütünlüğü, veri tabanı yöneticisi kontrolü dışındaki dış etkenlerden dolayı bozulabilir. Örneğin, veri tabanı bir yangın, patlama veya deprem nedeniyle zarar görebilir. Bu bozulma veya afetler yüzünden, yedekleme ve geri alma prosedürleri veri tabanı yöneticisi için çok önemli hale getirmektedir.

Veri tabanı güvenliğinin sağlanmasında en yetkili kişi veri tabanı yöneticisi olduğundan şimdi veri tabanı yöneticisi kısa olarak tanımlanacaktır.

Veri Tabanı Yöneticisi

Veri tabanı uygulamaları sayısı arttıkça, veri yönetimi giderek daha karmaşık hale gelmeye başlamış ve böylece veri tabanı yönetiminin gelişmesine yol açmıştır. Merkezi

ve paylaşılan veri tabanının kontrolünden sorumlu olan kişi veri tabanı yöneticisi olarak bilinmektedir.

Veri tabanı yöneticisinin rol ve sorumlulukları organizasyonun yapısına göre şirketten şirkete değişmektedir fakat genelde veri tabanı yöneticisi, veri tabanı sistemlerinin kullanılabilirlik, erişilebilirlik, kapasite ve performans yönetimini sağlar, veri tabanı yedekten dönme test aktivitelerini yapar, sistem işletmeni tarafından yapılacak aktiviteleri belirler ve altyapı gereksinimlerine destek olur. Bu görevleriyle, veri tabanı yöneticileri bir organizasyonun başarısında önemli rol oynar.

5. VERİ TABANLARININ BT DENETİMİNDEKİ YERİ

BT denetimi, bir bilgisayar sisteminin veri bütünlüğünü korumak için tasarlanıp tasarlanmadığını, varlıkları koruma altına aldığını, kurumsal hedeflere etkin bir şekilde ulaşılmasını sağladığını ve kaynakları verimli kullanıp kullanmadığını belirlemek için kanıt toplama ve değerlendirme sürecidir. Etkili bir bilgi sistemi, kuruluşun hedeflerine ulaşmasını sağlar ve gerekli hedeflere ulaşmada asgari kaynakları kullanır.

Bununla birlikte BT denetimi, sistem üzerindeki riskleri belirlemeyi ve belirlenen risklerin giderilmesi için kontroller koyulmasına odaklanmaktadır. Diğer yandan, bilgisayarlı verilerin ve bu verileri işleyen, bakımını yapan ve raporlayan sistemlerin gizliliği, bütünlüğü, kullanılabilirliği ve güvenilirliği BT denetiminin diğer amaçlarından biridir. BT denetçileri, amaçlandığı şekilde çalıştıklarından emin olmak için BT kontrollerinin bilgi sistemlerindeki ve ilgili operasyonlardaki etkililiğini ve verimliliğini değerlendirir.

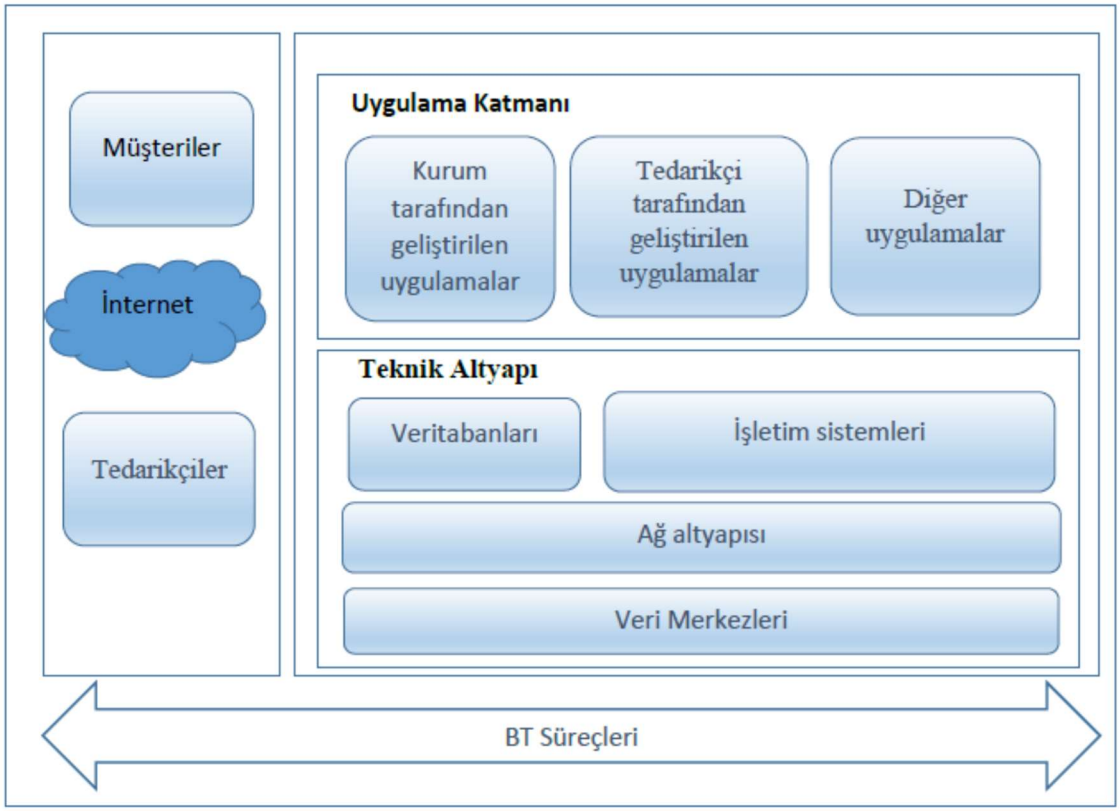
Teknolojinin hızlı gelişmesiyle birlikte, işletmelerde kullanılan sistemler de daha karmaşık bir hal almaktadır. Bu gelişmelerle birlikte mobil ve bulut sistemler, yapay zeka ile çalışan sistemler, uzman ve kurumsal kaynak yönetimi sistemleri, bilgi güvenliği gibi yeni kavramlar ortaya çıkmaktadır. Sektörün bu kadar hızlı gelişmesi nedeniyle bilgi sistemleri denetçilerinin işi zorlaşmaktadır. Bir organizasyonda BT denetimi yapılacaksa önce bilgi sistemleri anlaşılmalı, iş süreçleri belirlenmeli ve tanımlanmalı, bu bilgilere nasıl ulaşılabileceği bilinmelidir. Bu yüzden, denetçi, temel bilgisayar kavramlarına hakim olmalı ve teknolojinin getirdiği yenilikleri sürekli olarak takip etmek zorundadır.

Bilgi sistemleri; planlama, kontrol ve karar verme süreçlerinde gerekli olan bilginin toplanması, işlenmesi, erişilmesi, saklanması ve aktarılmasında kolaylık ve etkililik sağlanması amacıyla bilgi teknolojileri kullanılarak oluşturulan ve kullanıcılar ile etkileşim içerisinde bulunan sistemler bütünüdür. Bilgi sistemleri yönetimi, bilgi sistemlerinin etkin bir biçimde kullanılması ve organizasyonun amaçlarına uygun şekilde yürütülmesidir ve temel amacı, organizasyonun değerini ve kazancını

artırmaktır.

Standartlar, tüm denetimler ve denetçiler için bir çerçeve sunar ve denetimin zorunlu gerekliliklerini tanımlar. BT denetimi, bu çerçevede planlama, yürütme ve raporlama süreçlerinden oluşur ve finansal denetimlerle birlikte yürütülebileceği gibi güvenlik ve sistem denetimleri gibi bağımsız bir şekilde de yürütülebilir.

BT denetiminin hedefi, BT altyapı ve süreçlerinin, beklenen faydaları verip vermeyeceğine dair güvence vermektir. Bu faydalar etkililik, etkinlik, güvenlik, güvenilirlik ve yasalara uyumdur. Organizasyonların BT altyapıları birbirinden farklı olsa da genel olarak aşağıdaki şekildeki gibidir.

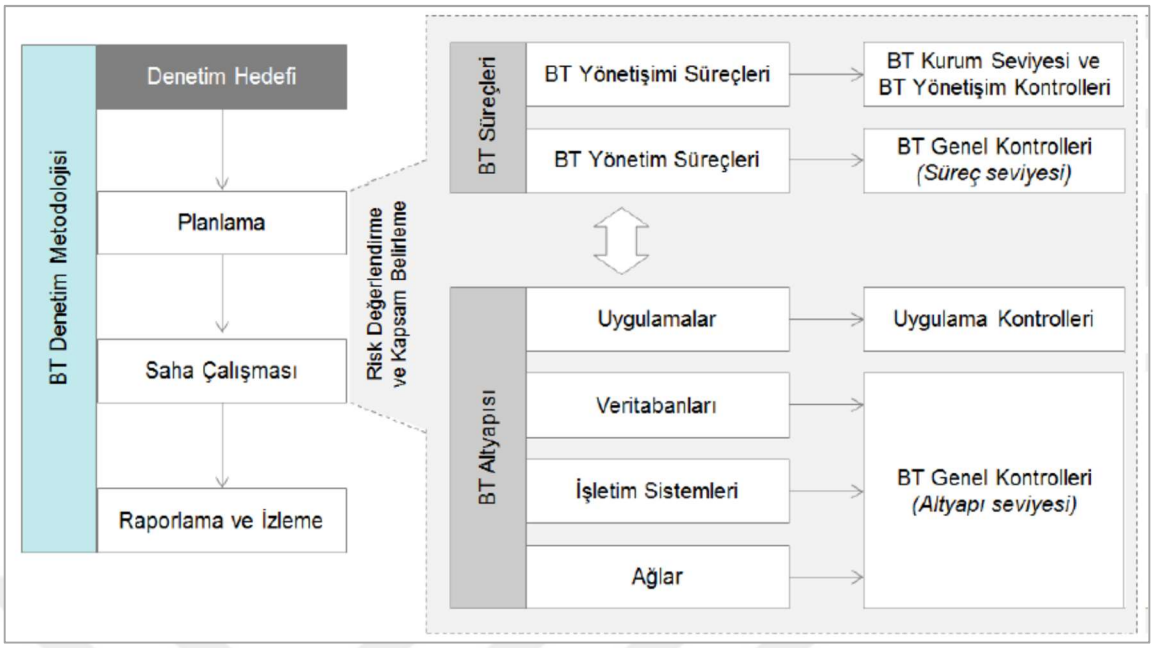


Şekil 5.1 BT' nin Genel Yapısı (Kamu Bilgi Teknolojileri Denetim Rehberi, 2014)

Denetimin etkili olabilmesi için risk analizinin yapılıp kapsam belirlenirken hangi katmanların öncelikli olduğu kararı verilmeli ve riskli olanlara uygun denetim kaynakları ayrılmalıdır. Denetim kapsamı BT' nin tüm katmanlarını içerecek şekilde

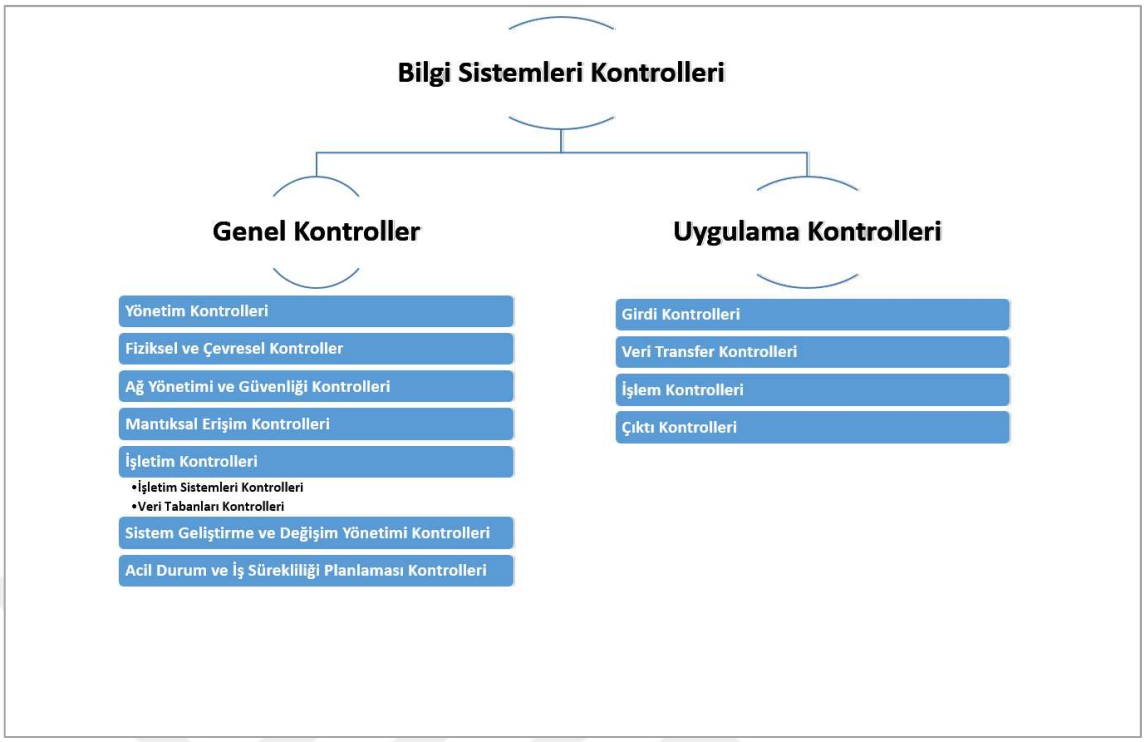
planlanmadıysa, BT' den kaynaklanan riskler yeterli şekilde ele alınmayabilir. Diğer yandan her katman tek tek denetlenirse denetim süreci uzun zaman alabilir ve denetim maliyetini arttırabilir. Bu nedenle denetim kapsamının iyi belirlenmesi gerekmektedir.

İşletmelerin en önemli verilerinden olan finansal bilgileri ve tabloları günümüzde bilgisayar sistemlerinde tutulduğundan finansal denetimlerin bilgi sistemlerinden istifade etmesi kaçınılmazdır. Bu nedenle finansal verilere makul güvence verilebilmesi için kurum bilgi sistemlerine makul güvence verilmelidir. Bilgi sistemlerine makul güvence verilmesi için de kurum bilgi sistemlerinin BT denetimine tabi tutulması gerekmektedir. BT denetimi aşamaları genel olarak planlama, denetimi yürütme, bulguların raporlanması ve izlenmesi şeklindedir. Planlama içerisinde; ön değerlendirme ve bilgi toplama işlemleri yapılarak denetimi yapılacak organizasyon anlaşılır, bu doğrultuda bilgi sistemleri ortamı anlaşılır ve buna göre denetim hedefi ve kapsamını tanımlamak için risk değerlendirmesi yapılır, denetimin nasıl yapılacağını belirten bir çalışma programı ve stratejisi oluşturulur. Yürütme sürecinde; kontrollerin ve bulguların değerlendirme işlemleri yapılır. Kontrollerin değerlendirme aşamasında, anahtar kontroller tespit edilir, kontrollerin tasarım ve işletim etkinliği değerlendirilir ve standart denetim prosedürleri kullanılır. Standart denetim prosedürlerinde BT genel kontrolleri, uygulama kontrolleri ve yönetim seviyesi kontrolleri (kurum seviyesi kontroller) bulunur. Raporlama ve izleme aşamasında ise, yürütme adımında bulunan bulgular raporlanır ve ilgililere sunulur, daha sonra raporlanan bulgulara alınacak aksiyonlar belirli aralıklarla izlenir. BT denetimi metodolojileri sektöre, ihtiyaca, denetim yetkisine göre farklılıklar gösterse de genel anlamda özet şekli aşağıdaki gibidir.



Şekil 5.2 BT Denetim Metodolojisi (Kamu Bilgi Teknolojileri Denetim Rehberi, 2014)

Şekilde görüldüğü üzere BT genel kontrolleri; veri tabanları, işletim ve ağ sistemleri kontrolleri olarak üç gruba ayrılmıştır. Sayıştay Bilişim Sistemleri Denetim Rehberine göre bu kontroller aşağıdaki şekildeki gibi gruplandırılmıştır:



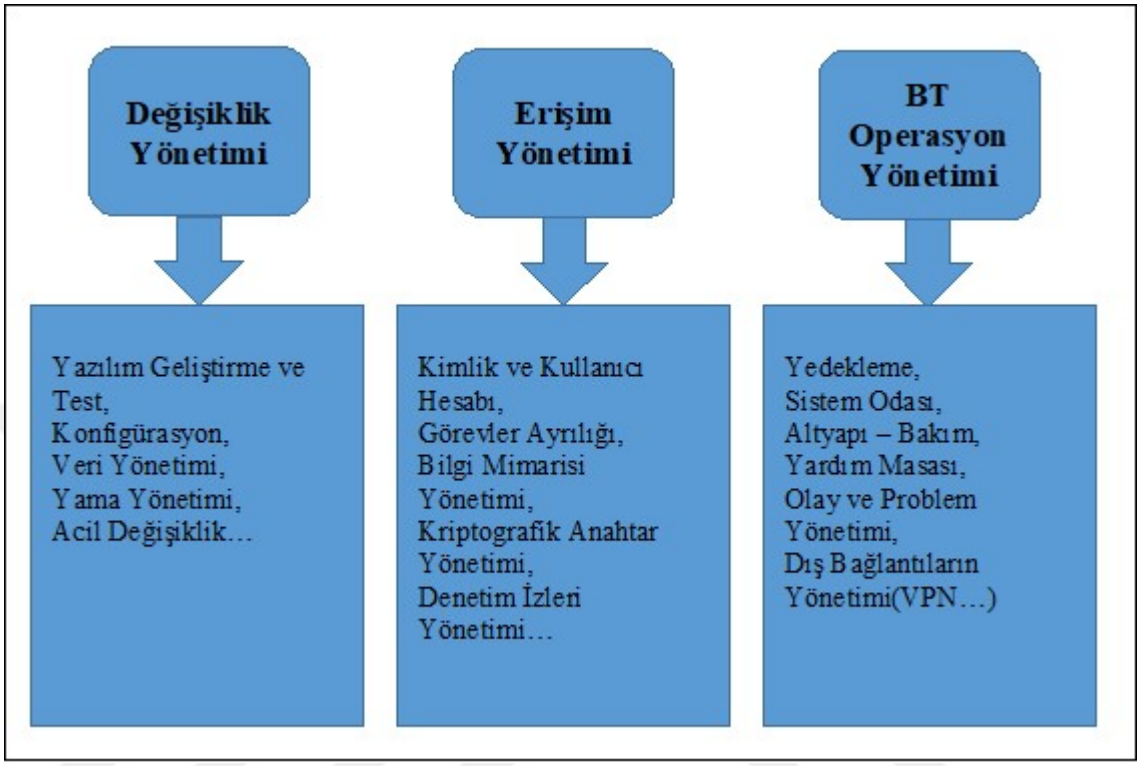
Şekil 5.3 BS Kontrolleri (Sayıştay, 2013)

BS genel kontrolleri; bilgi sistemlerinin kullanımı sonucunda ortaya çıkan riskleri azaltmaya (BS' ye bağımlı riskler) ve BS üzerinde kayıt edilmiş olan verilerin tamlığına, doğruluğuna ve bütünlüğüne yönelik güvence almayı hedefleyen kontrollerdir. Diğer yandan, BS genel kontrolleri, BT uygulamaları ve iş süreçleri için yeterince makul güvence ve destek sağlayabildiklerinden emin olmak için BT altyapı hizmetlerinde ve işletim sistemi, veri tabanı ve ağ gibi uygulamalarda yerleşik olan temel kontrollerdir.

BS genel kontrollerinin test edilmesinin amacı, süreç üzerindeki manuel ve otomatik uygulama kontrollerinin etkinliğine yönelik testler yapılmadan önce güvence elde edilmesidir. BS genel kontrollerinin etkin olmadığı durumlarda uygulama kontrollerinin tasarım ve işleyiş etkinliğine dair güvence elde edilmesi anlamlı değildir.

Denetim şirketleri, BS genel kontrollerini test ederken, kontrolleri genel olarak üç temel başlık altında sınıflandırmaktadır. Bunlar "Erişim Yönetimi", "Değişiklik Yönetimi" ve "BS Operasyonları" kontrolleridir. Bazı denetimlerde bu gruplandırmaya mevzuat olarak bazı ek maddeler de girebilmektedir. (Banka denetimlerinde İlkeler Tebliği

Mevzuatı gibi…) Kontrol gruplarının özelinde ise aşağıdaki şekilde yer alan temel kontrol hedeflerinin sağlanıp sağlanmadığına yönelik testler gerçekleştirilmektedir.



Şekil 5.4 BT Genel Kontrolleri

Şekilde yer alan, değişiklik yönetimi kontrol grubu; mevcut sistemlerin / uygulamaların yetkilendirilmiş, test edilmiş, onaylanmış, gerektiği şekilde uygulamaya alınmış ve dokümante edilmiş olmasını hedeflemektedir. Erişim yönetimi kontrol grubu; programlara erişim ve bilgi sistemleri kaynaklarına yetkisiz erişim riskini azaltmak amacıyla uygun kontrollerin tasarlanmasını ve işletilmesini hedeflemektedir. BT operasyonları kontrol grubu ise; sistem / uygulama işlemlerinin uygun olarak yetkilendirilmiş ve programlanmış işlemlerdeki sapmaların tanımlanmış ve çözümlenmiş olmasını hedeflemektedir.

Görüldüğü üzere, BT genel kontrolleri altında veri tabanları, tek başına ele alınabileceği gibi bu ortamlarda gerçekleştirilen değişiklik, erişim veya operasyonel gibi işlemler altında da değerlendirilebilir.

6. VERİ TABANLARININ DENETİMİ

6.1 Veri Tabanı Hakkında Genel Bilgiler

Veri tabanı denince genellikle ilişkisel veri tabanı yönetim sistemi aklımıza gelir. Veri tabanı yönetim sistemleri tablolarda veri kayıtlarını ve ilişkilerini veya indeksleri tutar. İlişkiler; tablolar ve veriler arasında yaratılabilir ve yönetilebilir.

Daha genel kavram olarak veri tabanı, herhangi yapıdaki herhangi bir veri kümesine uygulanabilir. Örneğin, müşteri kayıtlarını içeren düz bir dosya, bir uygulama için veri tabanı görevi görebilir. Bununla birlikte, bu bölümde kapsamlı bir ilişkisel veri tabanı yönetimi sistemi denetimine odaklanılacaktır.

Genel olarak bir denetim; fiziksel çevre, işletim sistemi, politikalar vb. gibi çeşitli alanların derinlemesine incelenmesini içerir. Veri tabanları, denetimin doğru bir şekilde gerçekleştirilmesi için sabır ve teknik bilgi gerektiren karmaşık yazılımlardır. Veri tabanları, bilgi çağının sanal kasalarıdır. Kuruluşlar en değerli varlıklarını; çevre aygıtlarında, bir eposta sisteminde veya dosya sistemlerinde değil, veri tabanlarında saklar. Bir güvenlik ihlali ve hassas verilerin çalındığı duyulduğunda, 'Bu veriler saldırıya uğradığında nerede canlı olarak tutuluyordu?' sorusuna; 'veri tabanında' cevabı alınır.

Veri tabanları, güvenlik duvarının gerisinde kalmalı ve derinlere gömülmelidir, yani sistem yapısında arka taraflarda olmalıdır. Çoğu kuruluş, en değerli verilerini güvenli olmayan dışa açık ağa (public network) koymamayı bilecek kadar akıllıdır. Elbette, SQL enjeksiyonu gibi bazı saldırılar bir güvenlik duvarını geçebilir ve veri tabanına erişebilir.

Bununla birlikte, güvenlik duvarının arka taraflarında yer almasından dolayı, veri tabanlarını koruma ve denetleme genellikle sonradan düşünülür, ekstra zaman kalırsa bu işler yapılır. Bu, veri tabanı güvenliği için kötü bir durumdur. Normal bir veri tabanı

yöneticisi, veri tabanının güvenlik duvarı arkasında olduğundan temel güvenlik önlemlerinin bile gereksiz olduğunu düşünebilir.

Güvenli oluşturulmuş bir yapı, veri tabanı için yeterli düzeyde koruma sağlayabilir. Ne yazık ki, güvenlik duvarı ve buna benzer yapılar artık yeterli değildir ve 'son savunma hattı' değildir. Artık, odak noktası, verinin veri tabanında bulunduğu yerde korumaya doğru kaymaktadır. Denetçi olarak, veri tabanının güvenlik zincirinin zayıf halkası olduğu muhtemelen bulunur ve tavsiye edilen gerekli önlemlerle veri tabanı güvenliği sağlanabilir.

6.2 Veri Tabanı Denetimi Esasları

Veri tabanını etkin bir şekilde denetlemek için, veri tabanının işleyişi hakkında temel bir bilgiye sahip olmak gerekir. Denetimin düzgün bir şekilde olması için de bazı bileşenleri anlamak gereklidir. Öncelikle biraz tarih bilgisiyle başlamak gerekirse;

1990'ların başında uygulamalar, bir ağ üzerinden doğrudan veri tabanına bağlanan bir masaüstü programından oluşan istemci-sunucu modeli kullanılarak yazılmaktaydı. Buna iki katmanlı uygulama denmekteydi. 1990'ların sonunda üç katmanlı uygulamalar standart haline gelmeye başladı. Bu yeni model, orta katmanlı bir web uygulamasına bağlanan web tarayıcısından oluşuyordu. Bu orta katman daha sonra veri tabanı sunucusuyla bağlantı kuruyordu. Üç katmanlı uygulamalar ileriye doğru atılmış büyük bir adımdı. Bu, yazılımların her istemci iş istasyonuna kurulmasının gerekmeyeceği ve yazılım güncellemelerinin merkezi bir sunucuya uygulanabileceği anlamına geliyordu. İstemciler (clients), basit bir tarayıcıyı destekleyen herhangi bir işletim sistemini çalıştırabiliyordu ve üç katmanlı modelde veri tabanını güvence altına almak çok daha basitti (Oluwatosin, 2014). Fakat şimdiki tehlike, bir saldırganın veri tabanı sunucusuna saldırmak için web uygulamasını tuzağa düşürebilmesidir.

6.2.1 Önemli veri tabanı sağlayıcıları

Genel olarak denetim, Oracle veya DB2 gibi bir veya iki veri tabanı sağlayıcısı üzerinde yoğunlaşmaktadır. Fakat, orta ölçekli veya büyük organizasyonlar birçok farklı veri tabanı platformunu kullanabilmektedir. En yaygın kullanılan veri tabanları; Oracle, Microsoft SQL Server, IBM DB2, Sysbase, PostgreSQL ve MySQL olarak sayılabilir.

6.2.2 Veri tabanı bileşenleri

Her bir veri tabanı sağlayıcısı çeşitli veri tabanı bileşenlerine sahiptir. Uygulamada biraz farklılıklar gösterse de teori ve ilkeler genel olarak ortaktır. Denetçi olarak bilinmesi gereken temel veri tabanı bileşenleri bu bölümde anlatılacaktır.

6.2.2.1 Program dosyaları

Bir veri tabanı, yazılım sistemi olarak uygulanır ve çekirdek işletim sistemi dosyaları içerir. Bu dosyalar, veri tabanı yönetim sisteminin çalışmasını sağlayan yürütülebilir dosyaları (executable files) içerir. Ayrıca, yürütülemez olan, yardım, kaynak, örnek ve yükleme dosyaları gibi diğer program dosyalarını da içerebilir.

Bu dosyalar korunmalıdır çünkü veri tabanı, bütünlüğüne güvenmektedir. Bunlar, her türlü değişikliklerden korunmalıdır. Erişim kontrolleri, bu dosyaları tutan dizinde mümkün olduğunca kısıtlı olmalıdır. Sadece veri tabanı yöneticilerinin bu dizine erişimi olmalıdır.

6.2.2.2 Yapılandırma dosyaları

Veri tabanları, sistemin nasıl çalıştığını belirleyen yapılandırma dosyalarına büyük ölçüde güvenir. Bunları korumak önemlidir, çünkü bu dosyalar değiştirilirse güvenlik ihlali yapılmış olur.

Yapılandırma değerleri, aşağıdakiler de dahil olmak üzere çeşitli yerlerde bulunur;

- İşletim sistemi metin dosyalarında
- Veri dosyalarında
- Windows'ta, kayıt defterinde
- Ortam değişkenlerinde

Yapılandırma değerleri, aşağıdaki verilenler gibi geniş bir ayar yelpazesinde kullanılır;

- Güvenlik veya kimlik doğrulama model türü ayarları
- Veri tabanı yönetici grupları ayarları
- Parola yönetimi özelliklerini belirleme
- Veri tabanı tarafından kullanılan şifreleme mekanizmasının belirlenmesi

Yapılandırma değerlerinin bütünlüğünü doğrulamak, denetimin kritik bir bileşenidir.

6.2.2.3 Veri dosyaları

Veri tabanları, genellikle bir dizi dosyadan oluşan fiziksel işletim sistemi dosyalarında bulunan verileri tutmaya ihtiyaç duyar. Dosyaların biçimi genelde özeldir ve veri dosyaları aşağıdakiler gibi bilgiler içerir;

- Saklanan veriler
- İşaretçiler (Pointers); bir alandan diğer alana veya bir satırdan diğer satıra
- İndeks verileri

İndeksler, işaret ettikleri verilerin bir alt kümesini içerir. Yani, bir saldırgan eğer indekse erişebilirse, fiziksel veriye erişmesine gerek kalmaz. Bu yüzden, herhangi bir indekse erişimin, verinin kendisine erişilmesi ile aynı derecede yetkilendirildiğinden emin olunmalıdır.

Genellikle, veri tabanı sözlüğü bu veri dosyalarında saklanır, dolayısıyla, bu dosyalara herhangi bir erişim, veri tabanında bulunan kontrolleri aşmak için kullanılabilir.

6.2.2.4 İstemci/Ağ kütüphaneleri(Client/Network libraries)

Bir veri tabanı sisteminin önemli bileşenlerinden biri istemcidir. Genellikle, istemci veri tabanından uzakta bir sistemde bulunur. İstemci, yerel (local) sistemden de bağlanabilir ki bu durum genellikle toplu işlemlerde (batch processes) gerçekleşir.

Bir istemcinin veri tabanına bağlanması için cihazında istemci kütüphanesi veya sürücüsü olması gereklidir. Bu, genellikle DLL' ler ve paylaşılan nesneler gibi bir dizi yürütülebilir dosyanın yanı sıra istemcinin veri tabanına bağlanmak için kullanabileceği bir API' den oluşur. İstemci kütüphanelerini korumak zordur çünkü erişim kontrollerinin sürdürülmesi daha zor olan yerdeki uzak sistemlerde bulunurlar. Bununla birlikte, yöneticilerin veya hatta standart kullanıcıların bağlanacağı yerlerde istemci sürücülerinin bütünlüğünü korumak oldukça önemlidir.

İstemci kütüphanelerinin bütünlüğünün bozulması zayıf bir noktadır. İstemci sürücüleri değiştirilirse, kimlik bilgileri kolayca çalınabilir. İstemci sürücüleri truva atı veya klavye dinleme sistemi (keylogger) gibi zararlı programlara maruz kalabilir, bu da veri tabanının tehlike altında olması anlamına gelir.

Ağ üzerinden iletişim için veri tabanında ağ sürücüsünün bulunması gerekmektedir. Bu sürücüler denetçinin odaklanması gereken başka bir konudur, çünkü bir saldırganın veri tabanına erişmek için kullanacağı yollardan birisidir.

6.2.2.5 Yedekleme/Geri yükleme sistemi

Yedeklemeler, her veri tabanı platformunun çok önemli bir parçasıdır. İster yazılım, ister donanım hatası olsun yedekleme yapmak sistemi geri yüklemek için kritik öneme sahiptir. Yedeklemeler, veri tabanının bir kopyasını içerir. Yedekleme, ayrı bir dosyada,

bir kasette veya ayrı bir yedekleme ünitesinde tutulabilir.

Veriler sıklıkla kaybolan veya sızdırılan yedekleme ünitelerinden çalınır. Veriler, genellikle dosyaya yazılırken veya yazıldıktan sonra tüm dosyanın şifrelenmesiyle korunur. Bu durumda, yedeklemenin düzgün bir şekilde korunması için şifreleme anahtarının saklanması önem kazanır. Burada önemli olan, şifreleme anahtarının verilerle birlikte doğru bir şekilde yedeklenmesini ve alınan yedeğin düzgün bir şekilde geri yüklenmesini sağlamaktır. Dosyalar geri yüklenemezse, yedekleme değersiz hale gelir ve geri yüklenemeyen yedekler işlerin aksamasına neden olur.

6.2.2.6 SQL deyimleri

Yapısal sorgulama dili olan SQL (Structured Query Language), ilişkisel veri tabanındaki verilere erişmek için kullanılır. Teknik olarak, SQL üç ayrı harfle 'S-Q-L' olarak telaffuz edilmelidir fakat 'sequel' telaffuzu yaygın olarak kullanılmakta ve bu da doğru olarak kabul edilmektedir. SQL set tabanlı bir dildir, yani bir kerede bir veri kümesi çalışır. Yordamsal (procedural) bir dil değildir, yani, while döngüleri, if deyimleri, for döngüleri vb. gibi yordamsal bileşenlere sahip değildir. Çoğu veri tabanı platformunda, yordamsal bileşenleri sağlamak için SQL uzantıları bulunur. Örneğin, Oracle PL/SQL diline, Sybase ve Microsoft SQL Server Transact-SQL diline sahiptir.

SQL deyimleri, veri tabanından verileri çekmek için kullanılır. SQL dört temel deyim etrafında oluşturulmuştur:

- SELECT bir tablodaki verileri görüntüleme
- INSERT bir tabloya yeni veri ekleme
- UPDATE bir tablodaki var olan veriyi değiştirme
- DELETE bir tablodaki verileri silme

En iyi anlaşılması gereken ifade 'SELECT' deyimidir. Temel sözdizimi şu şekildedir:

SELECT <sütun listesi> FROM <tablo ismi> WHERE <koşul>

Bu ifadede, <sütun listesi>, virgülle ayrılmış olarak gösterilecek sütun adlarının listesidir. Kısayol olarak, tüm sütunları görüntülemek için bir yıldız imi kullanılabilir. <tablo ismi> yerine, görüntülenmek istenen tablo adı yazılır. <koşul> ve WHERE kelimeleri isteğe bağlıdır. WHERE yan tümcesi belirtilmezse, tablodaki tüm satırlar döndürülür. WHERE yan tümcesini kullanarak, yalnızca eklemek istenilen satırlar seçilir.

Örneğin; on bin TL'den fazla kazanan tüm çalışanların adı ve soyadını getiren cümlecik şöyle yazılır;

SELECT AD, SOYAD FROM CALISANLAR WHERE MAAS > 10000

SELECT deyimleri bundan daha karmaşık hale gelebilir. Bununla birlikte denetim yapılırken genellikle çok karmaşık cümleciklere bakılmasına gerek yoktur.

6.2.2.7 Veri tabanı nesneleri

Bir veri tabanı, her biri benzersiz bir görev veya amaç taşıyan çeşitli nesneler içerir. Her bir nesneyi anlamak gerekli değildir fakat ortak nesne türlerinin kavranması gerekmektedir.

En sık kullanılan veri tabanı nesneleri aşağıda verilmiştir. Her bir veri tabanı platformu şemalar, kurallar, sıralar (sequence) gibi birçok özel nesneye sahiptir. Daha fazla ayrıntı için veri tabanı platformuna özgü özel dokümanlara bakılmalıdır.

Tablo, birbirleriyle ilgili sütunları olan satırlar bütünü

Görünüm, bir veya birkaç tablo ya da görünümden bilgiye erişmeyi sağlar. Görünümler, sütunların sayısını veya sırasını değiştirebilir, fonksiyonları çağırabilir ve

verileri çeşitli şekillerde değiştirebilir.

Saklı Yordam/Fonksiyon, veri tabanı içerisinde karmaşık işlevleri yerine getirmek için çağrılabilen yordamsal kodlardır. Fonksiyonlar değer döndürürken, saklı yordamlar değer döndürmezler. Saklı yordamlar veri erişimi için çok etkilidir.

Tetikleyici, bir tabloda değişiklik yapıldığında çağrılan yordamsal koddur. Veriler değiştirildiğinde, diğer tablolardaki değişiklikler de dahil olmak üzere herhangi bir işlemi gerçekleştirmek için kullanılabilir.

İndeks, verilerin hızlı aranmasını sağlayan mekanizmadır. İndeksler karmaşık nesnelerdir ve doğru ayarlanması veri tabanı performansı için kritik önem taşır.

6.2.2.8 Veri tabanı sözlüğü

Veri tabanı, veri sözlüğü veya bazen de sistem tabloları adı verilen kendisiyle ilgili meta verileri saklar. Meta veriler, veri tabanının kendi yapılandırması, kurulumu ve nesneleri hakkında bilgiler içerir. Meta veriler, veri tabanındaki bilgilerin içeriği hakkında bir şey söylemez, sadece veri tabanı biçimi hakkında bilgiler içerir. Veri sözlüğünün biçimi statiktir. Veri sözlüğü kendi yapısı hakkındaki meta verileri içerir ancak biçimi kolay değiştirilebilen bir şey değildir.

Veri sözlüğündeki meta veriler manipüle edilecek şekilde tasarlanmıştır. Nadiren veri sözlüğü doğrudan manipüle edilir. Bunun yerine, sistem tablolarını değiştirmek için karmaşık doğrulama mantığına sahip özel saklı yordamlar kullanılır. Sistem tablolarına doğrudan erişim tehlikelidir, çünkü küçük bir yanlış adım bile veri sözlüğünü bozabilir, bu da ciddi veri tabanı sorunlarına yol açabilir.

Veri sözlüğü; kullanıcılar, gruplar ve izinler gibi geri kalan veri tabanı nesnelerini tanımlar. Veri sözlüğü, fiziksel dosyaların diskte nerede saklandığını, tabloların adlarını, sütun türlerini ve uzunluklarını, saklı yordam, tetikleyici ve görünüm kodlarını da içeren

veri tabanı yapısını tanımlar.

6.3 Veri Tabanı Denetimi Adımları

Denetime geçilmeden önce bir kontrol listesine sahip olmak işimizi kolaylaştıracaktır. Aşağıda bahsedilecek olan temel kontrol listesini kullanarak denetim tamamlanabilir.

Veri tabanı yöneticisi ile denetimi görüşmek amacıyla bir toplantı yaparak denetime başlanır. Açıkçası veri tabanı yöneticisi, denetlenme fikri konusunda pek istekli olmayabilir, bu yüzden veri tabanı yöneticisine olabildiğince samimi bir şekilde yaklaşmak gereklidir. Veri tabanı yöneticisinin işini engellemek için değil, ona yardım etmek için orada olunduğunun anlamasından emin olunmalıdır.

Sisteme erişilecek süreyi optimize etmeye hazır olunmalıdır. Sistemde verilen hesapların yalnızca ihtiyaç duyulan izinlerle çalıştığından emin olunmalıdır. Herhangi bir iş tamamlandıktan hemen sonra veri tabanı yöneticisi yetki verilen hesabı kilitlemelidir. Hesap silinmemelidir, denetim tamamlanıncaya kadar hesap kilitlenmelidir. Böylece, daha çok bilgiye ihtiyaç duyulursa, hesap yeniden oluşturulmak yerine kolayca kilit kaldırılabilir.

Mümkün olduğunca çevrim dışı çalışılmalıdır. Sistem tabloları, şifre özetleri, dosya izinleri ve diğer tüm bilgiler yerel bilgisayara indirilip daha sonra, veri tabanından bağlantımızı kesip veri tabanını etkileyecek bir risk olmadan denetim adımları sürdürülmelidir. Örneğin, canlı veri tabanı üzerinde parolanın güçlü olup olmadığı test edilemez. Şifre özetleri bilgisayara indirilip parola gücü çevrim dışı olarak test edilebilir.

Denetçinin, veri tabanı yöneticisine, veri tabanında bu derece dikkatli davranmayı göstermesi, ondan profesyonel nezaket görmesini sağlar. Veri tabanı yöneticisi ile anlaşamamak, kuruluşa, denetimi değersiz kılabilir.

Buraya kadar veri tabanları hakkında verilen bilgiyle bir altyapı oluşturulmuştur. Şimdi, denetimi gerçekleştirmek için bir denetim planına ihtiyaç vardır. Burada ele alınan adımların çoğu, bir işletim sistemi veya ağ denetimi üzerinde gerçekleştirilen adımlara benzemektedir fakat bu adımların veri tabanı bağlamında yerleştirilmesi gerekir. Bahsedilecek bazı adımlar veri tabanına özgüdür.

Bu kısım beş ana başlık olarak ele alınıp her bir başlık altında denetim adımları anlatılacaktır. Bu başlıklar şu şekildedir:

- Kurulum ve Genel Kontroller
- İşletim Sistemi Güvenliği Kontrolleri
- Kullanıcı Hesap ve İzin Yönetimi Kontrolleri (Erişim Yönetimi)
- Veri Şifreleme
- İzleme ve Yönetim

Bu başlıkların altında kontrol listesi olarak verilecek adımların önce açıklaması yapılacak sonra nasıl yapılacağı anlatılacaktır. Ayrıca gerekli maddelerde ORACLE veri tabanında bu adımların nasıl yapılacağı sorgularıyla birlikte örnek olarak verilecektir.

a) Kurulum ve Genel Kontroller

1. Veri tabanı sürümünü edin ve politika gereksinimi ile karşılaştır. Veri tabanı sürümü desteğinin devam ettiğini doğrula.

2. Güvenlik yaması çıktığında politika ve prosedürlere göre zamanında uygulandığına bak. Onaylanan tüm yamaların veri tabanı yönetim politikasına göre kurulduğundan emin ol.

3. Yeni veri tabanı sistemleri için standart bir kurulum olduğunu kontrol et ve yeterli güvenlik ayarları olup olmadığına karar ver.

b) İşletim Sistemi Güvenliği Kontrolleri

4. İşletim sistemine erişimin doğru bir şekilde kısıtlandığından emin ol.

5. Veri tabanının yüklü olduğu dizine ve veri tabanı dosyalarına izinlerin doğru bir şekilde kısıtlandığından emin ol.

6. Veri tabanı tarafından kullanılan kayıt defteri anahtarları üzerindeki izinlerin doğru bir şekilde kısıtlandığından emin ol.

c) Kullanıcı Hesap ve İzin Yönetimi Kontrolleri

Veri Tabanı Kullanıcı Hesaplarını İnceleme

7. Kullanıcı hesabı oluşturma prosedürlerini gözden geçir ve hesapların sadece iş gereksinimlerine uygun şekilde oluşturulduğundan emin ol. Ayrıca, iş değişikliği veya işlemlerin sonlanma durumunda hesapların zamanında kaldırıldığını veya pasif edildiğini incele ve değerlendir.

Parola Gücü ve Yönetimi Özellikleri

8. Varsayılan kullanıcı adlarını ve parolaları kontrol et.

9. Kolay tahmin edilebilen parolaları kontrol et.

10. Parola yönetim yeteneklerinin aktif olduğunu kontrol et.

Veri Tabanı Yetkilerini İnceleme

11. Veri tabanı izinlerinin gerekli yetki seviyesi için uygun şekilde verildiğini veya

kaldırıldığını doğrula.

- Tablolara SELECT Yetkisi Dışında Yetkisi Olan Kullanıcıların Tespiti
- Önemli Rollere Sahip Olan Kullanıcıların Tespiti
- Sistemde Veri Tabanı Yöneticisi Kadar Yetkiye Sahip Kullanıcıların Tespiti
- Tablo Alanlarına (Tablespace) Sınırsız Yetkisi Olan Kullanıcıların Tespiti;

12. Gruplar veya roller yerine bireysel verilen veri tabanı yetkilerini gözden geçir.

13. Veri tabanı izinlerinin dolaylı olarak yanlış verilmediğinden emin ol.

- With Admin Option Yetkisi Verilen Kullanıcıların Tespiti

14. Saklı yordamlarda çalıştırılan dinamik SQL'leri incele.

15. Tablodaki verilere satır düzeyinde erişimin doğru bir şekilde uygulandığından emin ol.

- Görünümler (view) ve Yetki Kontrolleri

16. Gereksiz olan PUBLIC yetkilerini iptal et.

d) Veri Şifreleme

17. Ağ şifrelemesinin uygulandığını doğrula.

18. Durağan verilerin şifrelemesinin, gerektiğinde uygulandığını doğrula.

e) İzleme ve Yönetim

19. Veri tabanı faaliyet izleme (monitoring) ve denetiminin (auditing) doğru yapıldığını doğrula.

- Hangi Yetkilerin Denetlendiğinin (Audit) Tespiti

20. Var olan ve beklenen iş gereksinimlerini için veri tabanı kapasitesinin nasıl yönetildiğini değerlendir.

21. Veri tabanı ortamında var olan ve beklenen iş gereksinimlerini karşılamak için performansın nasıl yönetildiğini ve izlendiğini değerlendir.

22. Yedeklerin başarılı bir şekilde alındığını kontrol et.

23. Yedekleme ortamlarının saldırılara karşı önlem alındığını kontrol et.

6.3.1 Kurulum ve genel kontroller

1) Veri tabanı sürümü edinilir ve şirket politikası gereksinimi ile karşılaştırılır. Veri tabanı yazılım sürümünün güncel olup olmadığı ve sağlayıcı tarafından desteklenip desteklenmediği kontrol edilir.

Politikalar, bir çevreyi daha güvenli, kolay yönetilebilir ve denetlenebilir kılmak için yazılır ve onaylanır. Veri tabanının, organizasyonun politikasına uygun olduğundan emin olmak için temel yapılandırma ayarları kontrol edilir. Eski veri tabanları, ortamın yönetimini zorlaştırır ve birbirinden farklı veri tabanı sürümleri yöneticinin iş yükünü artırır. Standart geliştirmeleri ve yamaları güncel tutmak, veri tabanlarını yönetme sürecini kolaylaştırır. Buna ek olarak, pek çok eski veri tabanı, veri tabanı sağlayıcısının artık desteklemediği yazılım sürümlerinde çalışır. Bu durum, güvenlik açığı ortaya çıkarabilir ve sağlayıcı tarafından eski sürümlere yama sağlanmadığından sorun

düzeltilemeyebilir.

Nasıl Yapılır?

Veri tabanı yöneticisi ile yapılan görüşmeler ve organizasyonun BT standart ve politikalarının incelenmesiyle, şirket tarafından veri tabanı sürümü ve platformu olarak ne önerildiği ve desteklendiği belirlenir. Veri tabanı sağlayıcısı tarafından hangi sürüm ve platformun desteklendiği ve yeni güvenlik sorunları için yamaların sağlanıp sağlanmadığı doğrulanır. Veri tabanında çalışan sürümlerin envanteri çıkarılır ve desteklenmeyen sürümler işaretlenir. İdeal olan, veri tabanını desteklenen güncel sürümde tutmaktır.

Örnek:

Oracle veri tabanında, veri tabanı sürümünü öğrenmek için aşağıdaki sorgu yazılır. Sürüm bilgisi 'v$version' tablosunda tutulmaktadır. Bu tabloda Oracle, PL/SQL vb. versiyon bilgileri tutulmaktadır.

```
SELECT * FROM v$version;
```

2) Onaylanan tüm yamaların veri tabanı yönetim politikasına göre kurulduğundan emin olunmalıdır.

Çoğu veri tabanı sağlayıcısı düzenli olarak yamaları piyasaya sürmektedir. Bu yamalar için hazırlıklı olunmalıdır ve böylece yükleme ve test etme aşamaları gereğine uygun olarak yapılır. Tüm veri tabanı yamaları yüklü değilse, yaygın olarak bilinen güvenlik açıkları ortaya çıkabilir.

Nasıl Yapılır?

Veri tabanı yöneticisi ile görüşülür ve yamaları uygulamak için atılan adımlar nelerdir

ve yamalar, canlı veri tabanına uygulanmadan yeteri kadar test edilmiş mi sorularına cevap aranır.

En son yamalar hakkında olabildiğince bilgi edinilir ve bu yamaların hangi güvenlik açıklarını kapadığı belirlenir. Mevcut yamalar uygulanan yamalar ile karşılaştırılır. Yamalar zamanında uygulanmadığında olası riski azaltmak için atılan adımlar hakkında veri tabanı yöneticisi ile görüşülür. Çoğu veri tabanı yöneticisi güvenlik açığı bulunan sistem bileşenlerini kaldırarak yama gereksinimini azaltmaya çalışır. Bu, güvenlik riskini azalttığı için iyi bir uygulama olmasına rağmen, uzun dönemde yama yükleme işleminin devam etmesi tavsiye edilir.

Veri tabanları, yamalama konusunda çoğu organizasyon için ikilem oluşturmaktadır. Birçok veri tabanı 7/24 kesintisiz çalışmakta, bu yüzden kesintiye tahammül edilmemektedir. Bu, yamaları uygulamak için veri tabanının hiçbir kesintiye uğramaması gerektiği anlamına gelmektedir.

Veri tabanı yama işlemleri için bir diğer önemli sorun, yeni çıkan yamaların testi için geçen sürenin 3-6 ay arası sürmesidir. Veri tabanları çok kritik yazılımlar olduğundan, yamalar test edilmeden uygulanamaz. Veri tabanı yöneticileri için sadece yamaların uygulanması tam zamanlı bir iş olabilmektedir. Bunun çözümü bu yamaların uygulanması için sadece bu işle ilgilenecek kişiler atanmasıdır.

Yama uygulanırken kesinti yaşanmaması için kümeleme kullanılmaktadır. Kümeleme sisteminde, tek bir sunucu çevrim dışı yapılır, yama uygulanır, tekrar çevrim içi yapılır.

Örnek:

Oracle veri tabanında, hangi yamaların uygulandığı bilgisi, işletim sisteminde aşağıdaki komutu yazarak öğrenilebilmektedir.

$ORACLE_HOME/OPatch/opatch lsinventory

3) Yeni veri tabanı sistemleri için standart bir kurulum olup olmadığını ve temelinin yeterli güvenlik ayarlarına sahip olup olmadığını belirlemek.

Yeni sistemler test veya canlı ortamlara alınmadan önce doğru bir şekilde kurulmalıdır.

Nasıl Yapılır?

Sistem yöneticisi ile yapılan görüşmeler neticesinde, yeni bir sistemi geliştirmek veya uygulamaya almak için kullanılan metodoloji belirlenir.

Örnek:

Veri tabanı yönetimi prosedürüne göre standart bir kurulum şöyle olmaktadır;

- İlgili proje ya da uygulamanın, güvenlik, performans, bütçe durumuna göre konsolide veri tabanı sunucuları, bu işe dedike sunucu ya da sanal sunucu üzerine kurulum planlaması yapılır.
- Yeni sunucu kurulumu olacaksa ve uygulama için herhangi bir koşul yoksa o an için en güncel işletim sistemi versiyonu kullanılır. İşletim sistemi için genel güvenlik kuralları uygulanır.
- Yeni sunucu üzerine kurulumda veri tabanı için yine herhangi bir uygulama kısıtı yoksa ilgili veri tabanının en güncel sürümü kurulur.
- Mevcut sunucular üzerine konumlandırılma durumunda var olan konsolide sunuculardaha önce bu amaçla açılmış instance' ler mevcut ise bu instance' ye veri tabanı ya da şema konumlandırılır. Eğer böyle bir instance mevcut değilse konuyla ilgili bir isimlendirme ile yeni bir instance oluşturulur.
- Kurulumdan sonra sisteme ait kullanılmayan yüksek yetkili kullanıcılar (MSSQL:sa) kilitli değilse kilitlenir ve kullanıma kapatılır.
- Oluşturulan yeni veri tabanı/sunucu izleme/monitoring ekiplerine sunucunun izlenmesi ve sorunlu durumlarda alarm üretmesi için iletilir.
- Yeni veri tabanı/sunucu, veri sahiplerinin belirleyeceği kurallara göre

yedeklenmek üzere yedeklemeden sorumlu ekiplere iletilir.

- Veri tabanı yöneticilerine ait ilgili envanter dosyası güncellenir.

6.3.2 İşletim sistemi güvenliği kontrolleri

Bu kontroller işletim sistemi kontrollerine girdiği için burada kısaca bahsedilecektir. Güvenli olmayan bir veri tabanından işletim sistemine sızılabileceği gibi güvenli olmayan bir işletim sisteminden de veri tabanına sızma işlemi gerçekleşebilir. Birini korunaklı hale getirmek ikisini de güvenli hale getirmez. Veri tabanları en değerli hedef olduğundan dolayı yine de en çok odaklanılması gereken sistemlerdir.

4) İşletim sistemine erişimin doğru bir şekilde sınırlandırıldığından emin olunur.

İşletim sisteminin yalnızca veri tabanına ayrılması en iyi durumdur. Veri tabanı yöneticileri dışındaki hiçbir kullanıcı, SSH, FTP veya uygulama dışından herhangi bir yöntemle işletim sistemine bağlanma yetkisine sahip olmamalıdır. Çoğu uygulama için kullanıcılar veri tabanını doğrudan güncellememelidir. (uygulama dışından). Güncellemeler uygulama aracılığıyla gerçekleştirilmelidir. Uygulamanın dışında, verilerin doğrudan güncellenmesi veri tabanını bozabilir. Kullanıcıların genellikle uygulama dışından verileri güncellemek için sebepleri vardır. Bu, kullanıcı adına veri tabanına yapılan güncellemeleri gerçekleştiren (uygulama içindeki kullanıcının yetkisine bağlı olarak), uygulama için ortak bir veri tabanı kullanıcısıyla (generic user id) gerçekleştirilir.

Nasıl Yapılır?

İşletim sistemine yapılan tüm erişimlerin sadece veri tabanı yöneticisi ile sınırlı olduğu doğrulanır. Herhangi bir erişimin SSH gibi güvenli protokoller üzerinden yapıldığı doğrulanır. İşletim sisteminde kaldırılması gereken hesaplar kontrol edilir.

Örnek:

Yukarıda belirtilen güvenli bağlantılar dışında Oracle veri tabanının kullandığı işletim sistemi üzerindeki dizinlerde erişimin sadece Oracle kullanıcısı olduğundan emin olunmalıdır. (Linux işletim sistemi için dizin yetkisi kritik olanlar için 600 diğerleri için 750 olarak ayarlanabilir.)

5) Veri tabanının yüklü bulunduğu dizine ve veri tabanı kendi dosyalarına izinlerin düzgün bir şekilde kısıtlanmış olduğundan emin olunur.

Veri tabanı dosyalarına uygun olmayan erişimler ve güncellemeler veri tabanına büyük hasarlar verebilir. Örneğin, veri tabanı dosyalarını içeren işletim sistemi veri dosyaları üzerinden yapılacak herhangi bir doğrudan değişiklik veri tabanını bozabilir. Ayrıca, Oracle veri tabanında, bulunan redo log dosyaları, veri tabanının çökmesi durumunda işlenmemiş (uncommitted) verilerin kurtarılmasına olanak sağlar ve kontrol dosyaları bu gibi işlemleri yapmak için veri tabanı tarafından kullanılır. Bu dosyaların işletim sistemi üzerinden doğrudan güncellenmesi veri tabanı işlevselliğine zarar verebilir. Her veri tabanı yönetim sisteminin kendine özgü başlangıç, kayıt ve yapılandırma dosyaları vardır ve veri tabanının devamlılığını ve bütünlüğünü sağlamak için bu dosyaların korunması çok kritiktir.

Nasıl Yapılır?

Veri tabanının yüklü olduğu dizinde izinlerin mümkün olduğunca kısıtlayıcı ve yetkili veri tabanı yöneticisi hesabına ait olduğu doğrulanır.

Windows' ta benzer önlemler alınır. Veri tabanının yüklü olduğu dizinin dosya izinleri veri tabanı altında çalışan hesabın izinleri düzeyine kısıtlanmalıdır. 'Everyone' ve 'Anonymous' yetkileri veri tabanı dosyaları üzerinde herhangi bir yetkiye sahip olmamalıdır. Buna ek olarak, veri tabanı dosyalarını saklamak için kullanılan tüm sürücülerin NTFS kullandığından emin olunur.

İdeal olan, veri tabanı yöneticisinin bile temel işletim sistemi dosyalarına izninin olmamasıdır. Bununla birlikte, veri tabanı yöneticisi, veri tabanı dosyaları ve yedeklemeler üzerinde çalıştığı, yama işlemleri ve diğer işlemleri yapması gerektiği için işletim sistemi dosyalarında bazı erişimlere ihtiyaç duyacaktır. İşletim sistemine erişimi gerekmeyen ayrıcalıklı kullanıcılara bu konuda izin verilmemelidir.

İşletim sistemine bağlanarak veya denetçinin kendisinin çekerek ya da yöneticiden edinilerek, tüm veri tabanı dosyalarındaki ve bunların bulunduğu dizinlerin dosya izinlerinin bir listesi alınır. Yüksek ayrıcalıklı yetkileri bulmak için bu liste gözden geçirilir. Unix' te izinlerin 770 izninden fazla olmaması gerekir. Eğer 'Everyone' yetkisine sahip tüm kullanıcıları kilitlersek, çoğu programın çalışması durabilir, bu yüzden dikkatli olunmalıdır. Sıkı güvenlik ayarları iyi bir hedeftir fakat bunun için bazı istisnalar koyulması gerekir ve bu istisnaların nedenlerinin belgelendirildiğinden emin olunmalıdır. Windows'ta, 'Everyone' yetkisi verilmediğinden emin olunmalıdır. En ideali, sadece erişimi gerektiren veri tabanı kullanıcılarına yetki vermektir.

Örnek:

Oracle veri tabanında, aşağıdaki dosyalara yetkisi olan kullanıcı ve gruplar temin edilip sadece gerekli olanların yetkisi olduğu kontrol edilir:

Tablo Alanı (tablespace) veri dosyaları: Oracle; tabloları, paketleri, prosedürleri vb. tablo alanlarında saklar. Tablo alanı dosyalarının uzantıları '.dbf' şeklindedir. Şu klasörde tutulur:

$ORACLE_BASE/oradata/<sid>

Başlangıç Dosyası: Başlangıç parametreleri başlangıç dosyasında saklanır. Oracle veri tabanı çalıştırıldığında varsayılan olarak çalışan başlangıç parametrelerini yapılandırmak için bu dosya kullanılır. Bu dosya:

INIT.<veri tabanı>.ORA

Bu dosyada yapılan değişikliklerin aktif olması için veri tabanı yeniden başlatılmalıdır.

Kontrol Dosyaları: Veri tabanı fiziksel yapısının değişimini kontrol etmek için kullanılan dosyalardır. Veri tabanı başlangıcında ve kurtarımında kullanılır. Bulunduğu konum aşağıdadır:

INIT.ORA dosyasında, **CONTROL_FILES** parametresinde listelenir.

Yapılandırma Dosyası: Bu dosyalarda sistem güvenliği hakkında önemli bilgiler bulunur.

CONFIG.ORA dosyası yapılandırma ayarlarını içeren dosyadır.

Veri Tabanı Dinleyici Dosyası: Veri tabanı dinleyicisinin bilgilerinin tutulduğu dosyadır. Dinleyici servisi veri tabanına erişim sağlamak için çok kritik öneme sahiptir. Bu yüzden 1521 olan varsayılan portu değiştirilmeli, bu servise parola koruması konulmalıdır.

Oracle Parola Dosyası: Veri tabanı yöneticisi parolası veri tabanında saklanmaz çünkü instance başlatılmadan önce Oracle veri tabanına erişemez. Bu yüzden, Veri tabanı yöneticisinin kimlik doğrulaması veri tabanının dışında gerçekleşmelidir. Orapwd dosyası yüksek yetkiye sahip kullanıcıların parolalarını şifreli biçimde (hashed) tutmaktadır. Varsayılan yolu aşağıdaki gibidir:

Unix işletim sisteminde,

$ORACLE_HOME/dbs/orapw$ORACLE_SID,

Windows işletim sisteminde,

%ORACLE_HOME%\database\PWD%ORACLE_SID%.ora

6) Veri tabanı tarafından kullanılan kayıt defteri anahtarları üzerindeki izinlerin doğru şekilde kısıtlandığından emin olunur.

Windows ortamında çalışan veri tabanı platformları için, veri tabanı tarafından kullanılan kayıt defteri anahtarları düzgün bir şekilde güvence altına alınması gerekir. Kayıt defteri anahtarları, veri tabanının güvenli bir şekilde çalışması için önemli olan yapılandırma değerlerini tutmak için kullanılır. Kayıt defteri anahtarlarını düzenlemek, silmek, oluşturmak veya görüntülemek için verilen izinlerin yalnızca veri tabanı altında çalışan hesaba verildiğinden emin olunur.

Nasıl Yapılır?

Yöneticiden bilgiler edinilerek kayıt defteri düzenleyicisi aracılığıyla güvenlik izinleri gözden geçirilir. İzinlerin tam listesi alındıktan sonra, aşırı yetkinin olup olmadığı kontrol edilir.

Örnek:

Windows işletim sisteminde çalışan Oracle veri tabanlarında kayıt defterinde bulunan dosyalar ve içindeki anahtarların yapısı bozulmamalıdır. Uygun olmayan değerler verilmesi sistemin kullanılamaz hale gelmesine sebep olabilir. Bu nedenle, yalnızca ileri düzey kullanıcılar kayıt defterini düzenlemeli ve kayıt defterinde herhangi bir değişiklik yapmadan önce sistem yedeklenmelidir. Windows kayıt defterinde Oracle veri tabanının parametreleri aşağıdaki yollarda bulunur;

HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\KEY_HOME_NAME

HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE

HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services

6.3.3 Kullanıcı hesap ve izin yönetimi kontrolleri

Veri Tabanı Kullanıcı Hesaplarını İnceleme

Veri tabanının karmaşıklığı, kullanıcı oluşturma, yönetme, yetkilendirme, kimlik doğrulama ve denetleme gibi işlemlerin zamanında yapılması gerekliliği herhangi bir düzeyde hesap yönetimini zor hale getirmektedir. Bu zorluklara, veri tabanında saklanan hassas verilerin doğasında olan riskler de eklenince özellikle denetimin bu kısmı çok önemli hale gelmektedir.

Gerçek ortamlarda kullanıcılar oluşturulurken ve yetkileri verilirken ilgili bölümlerde onay aşamalarından geçmesi gerekmektedir. Bu durum aşağıdaki tüm maddeler için geçerlidir. Denetçiler, görevler ayrılığı prensibine aykırı bir yetkiyle karşılaşırsa bunun neden verildiğinin kanıtını ister ve bunun gösterilebiliyor olması gerekir.

7) Kullanıcı hesapları oluşturma prosedürleri gözden geçirilir ve hesapların sadece iş gereksinimine uygun olarak oluşturulduğu kontrol edilir. Ayrıca, kullanıcı hesaplarının, iş değişimi veya bir işin sonlanması durumunda zamanında kaldırılması veya pasif edilmesi kontrol edilir.

Veri tabanına erişimi sağlamak veya kaldırmak, veri tabanı kaynaklarına gereksiz erişimi sınırlamak için etkin kontroller mevcut olmalıdır.

Nasıl Yapılır?

Veri tabanı yöneticisi ile görüşülür ve kullanıcı hesabı oluşturma prosedürleri incelenir. Bu işlemin içinde kullanıcının sadece işiyle ilgili yetkilere sahip olduğu da olmalıdır. Veri tabanı yöneticisi düzeyinde hesapların veya yetkilerin en aza indirgenmesi sağlanır.

Hesapların oluşturulmadan önce onay mekanizmasından geçip geçmediği incelenir. Sistemdeki her kullanıcının kendi kullanıcı hesabına sahip olduğundan emin olunur. Hiçbir misafir veya grup hesabı olmamalıdır. Çok sayıda veri tabanı hesabı varsa, nedenleri sorgulanır. Uygulama kullanıcıları, doğrudan veri tabanına erişmek yerine uygulama aracılığıyla erişmelidir.

Erişime ihtiyaç duyulmadığında, hesapları kaldırma işlemleri de kontrol edilir. Ayrıca, iş değişikliği ve işlerin sonlanması durumunda hesapların nasıl kaldırıldığı süreci kontrol edilir. Bu süreçte, örnek kullanıcı hesapları alınıp bunların aktif çalışanlara ait oldukları ve hesapların oluşturulduğu zamandan bu yana iş pozisyonlarının değişmediği doğrulanır.

Örnek:

Oracle veri tabanında, tüm kullanıcılar dba_users tablosunda tutulur. Kullanıcı listesini alırken ihtiyaca göre bu tablodan ilgili sütunlar seçilerek sorgular yazılabilir. Örnek olarak;

SELECT username, account_status FROM dba_users;

Sorgusu yazılarak kullanıcı adları ve hesabın durumu listelenebilir.

Ayrıca veri tabanına belli bir süre erişim yapmayan kullanıcıların da listelenmesi gerekebilir. Bunu yapabilmenin çeşitli yolları vardır. Bunlardan biri logon trigger adlı tetikleyiciyi kullanmaktır. Bu tetikleyici kullanılarak hangi kullanıcının, ne zaman, hangi makineden bağlandığı vb. bilgileri öğrenilebilir. Ayrıca denetim (audit) parametrelerini kullanarak aşağıdaki sorgudaki gibi 50 gündür bağlanmayan kullanıcıları listeleyebiliriz;

SELECT username FROM dba_audit_trail WHERE action_name = 'LOGON' GROUP BY username HAVING MAX(timestamp) < sysdate – 50;

Parola Gücü ve Yönetimi Özellikleri

Çoğu veri tabanı platformu kendine özgü kimlik doğrulama mekanizmasına sahiptir. Parolaların ve kimlik doğrulama mekanizmasının zayıflığı kontrol edilir.

Diğer veri tabanı platformları kimlik doğrulamayı sağlamak için işletim sistemleri ile entegre çalışır veya güvenlik alt sistemleri vardır. Mesela, DB2 for z/OS veri tabanı, işletim sisteminin kullanıcı adı ve şifresini kullanmak yerine RACF yazılımını kullanarak ayrı bir kimlik doğrulama sağlar. Microsoft SQL Server, Windows modunda Windows kimlik doğrulamasını kullanır. Bu, kullanıcıların veri tabanında tutulmadığı anlamına gelmez. Kullanıcıların gruplara, izinlere ve diğer veri tabanı ayarlarına eşleştirilmesi yapılır. Fakat, kimlik doğrulama, veri tabanı yerine işletim sistemi düzeyinde yapılır.

Veri tabanı platformlarında entegre işletim güvenliği kullanmanın avantaj ve dezavantajları vardır. Avantajları;

- İşletim sistemi kimlik doğrulaması genellikle veri tabanı doğrulamasından daha güvenlidir.
- İşletim sistemi kimlik doğrulaması genellikle daha fazla parola yönetim özelliği içerir.

Dezavantajları;

- Kimlik doğrulama veri tabanı yöneticisinin elinde değildir.
- İşletim sistemi hesabı olan bir kullanıcı düzgün yapılandırılmadıysa veri tabanının işletim sistemine erişme olasılığı vardır.

8) Varsayılan kullanıcı adları ve parolaları kontrol edilir.

Denetimin ilk temel öğesi varsayılan kullanıcı adı ve parolaları kontrol etmektir. Veri

tabanı solucanları varsayılan kullanıcı adlarını ve parolalarını kullanarak veri tabanlarında çoğalmaktadır. Çizelge 6.1, varsayılan kullanıcı adlarını ve parolaları birkaç kategoriye ayırmaktadır.

Nasıl Yapılır?

Tüm varsayılan kullanıcı adları ve parolaların kaldırıldığını, kilitlendiğini veya parolaların değiştirildiği doğrulanır. Bunu yapmak için ücretsiz ve ticari yazılımlar, araçlar mevcuttur.

Çizelge 6.1 Varsayılan Parolalar

Kategori	Açıklama
Varsayılan veri tabanı parolaları	Standart bir veri tabanı kurulumunda oluşturulur. Yeni sürümlerin çoğu artık varsayılan şifreleri kaldırmıştır fakat eski sürümlerde sorun olmaya devam etmektedir.
Örnek parolalar	SQL komut dosyalarında bulunan test veya örnek hesaplardır.
Varsayılan uygulama parolaları	Üçüncü parti ürünler bir veri tabanına kurulduğunda, ürün genelde veri tabanına kendi kullanıcı adı ve parolasıyla bağlanır. Bunlar bilgisayar korsanları tarafından bilinir.
Kullanıcı tanımlı varsayılan parolalar	Yeni bir hesap oluşturulduğunda, parolanın bir başlangıç değeri olur ve ilk kullanımda sıfırlanır. Parola ilk oluştuğunda ve hiç kullanılmadığında problem olur, bunu önlemek için yeni hesapların parolaları rastgele ve güçlü olarak verilir.

Örnek:

Oracle veri tabanı kurulumu ile birlikte bazı kullanıcılar varsayılan olarak gelir. Bu varsayılan kullanıcıların hesapları kilitlenmeli veya mümkünse silinmelidir. Varsayılan kullanıcı listesi Oracle veri tabanının versiyonuna göre farklılık gösterebilmektedir. Aşağıdaki komut yazılarak varsayılan kullanıcılar elde edilebilir:

SELECT * FROM DBA_USERS_WITH_DEFPWD;

9) Kolayca tahmin edilebilen parolalar kontrol edilir.

Kullanıcılar genellikle programlar veya bilgisayar korsanları tarafından kolayca tahmin edilebilen parolaları seçerler. Şifrelerin atak yapılacak bir sözlükte veya kolay tahmin edilebilir olmamasına dikkat edilmesi çok önemlidir.

Nasıl Yapılır?

Parolaların kolay tahmin edilmesini sınamak için parola özetleri (hashes) üzerinde gücü test edilir. Eğer parolanın bir sözlükte bulunabilir veya tahmin edilebilir olduğu tespit edilirse, veri tabanı yöneticisi ile kullanıcı farkındalığı ve güçlü parola kontrolleri hakkında konuşulur. Bir sonraki adımda anlatılacak olan güçlü parolayı oluşturmak için sistem yapılandırma ayarlarına bakılabilir.

Örnek:

Bir sonraki maddede anlatılacak olan password verify fonksiyonu Oracle veri tabanında kullanılarak parola değiştirilirken sekiz karakterden küçük olan, ardışık sayılardan oluşan, kullanıcı adıyla aynı olan gibi zayıf girilen parolaları kontrol ederek girilmesini engelleyebilir. Password verify fonksiyonu örneği **EK 1**'de verilmiştir.

10) Parola yönetim yeteneklerinin aktif edildiği kontrol edilir.

Çoğu veri tabanı platformu zengin parola özellikleri desteği sağlar. Genellikle aşağıdaki özelliklerle sahiptirler:

- Parola gücü doğrulama fonksiyonları
- Parola kullanım süresinin dolması
- Parolanın yeniden kullanım limiti
- Parola kullanım süresi
- Parola kilitlenmesi
- Parola kilit sıfırlama

Bu özellikler yapılandırılmazsa, herhangi bir ilave güvenlik sağlanmaz. Varsayılan olarak, bu özellikler genellikle pasiftir.

Nasıl Yapılır?

Veri tabanından yapılandırma değerleri seçilir. Her bir parola yönetim özelliğinin aktif olduğu doğrulanır ve şirket politikalarına, çevreye göre uygun ayarlar seçilerek yapılandırılır. Bu özelliklere nasıl bakılacağı ve bunların getirilmesi için gereken komutlara veri tabanı dokümanlarından bakılır.

Örnek:

Oracle veri tabanında, Password Verify isimli fonksiyon, kullanıcı parolalarını kontrol etmek için kullanılır. Bu fonksiyon, sys şeması altında oluşturulur. Kullanıcı, parolasını değiştirdiğinde yeni parola bu fonksiyon tarafından kontrol edilir ve fonksiyon, parola gereksinimlerini karşılayıp karşılamadığının kontrolünü yapar. Password Verify fonksiyonu aşağıdaki yolda bulunur:

$ORACLE_HOME/rdbms/admin/utlpwdmg.sql

Bu fonksiyon 3 tane parametre kullanır, bunlar; kullanıcı adı, parola ve eski paroladır ve ‘bool’ değer döndürür ki ‘true’ döndüğü takdirde parola geçerli anlamına gelir. Fonksiyon, ihtiyaca göre özelleştirilebilir.

Veri Tabanı Yetkilerini İnceleme

Veri tabanı yetkileri işletim sistemi izinlerinden biraz farklıdır. Yetkiler, ‘GRANT’ ve ‘REVOKE’ ifadeleri kullanarak yönetilir. Örneğin, aşağıdaki SQL cümleciği KULLANICI1’e MAAS tablosu için SELECT yetkisi verir:

```
GRANT SELECT ON MAAS TO KULLANICI1;
```

‘REVOKE’ ifadesi verilen yetkiyi kaldırmak için kullanılır:

```
REVOKE SELECT ON MAAS FROM KULLANICI1;
```

‘GRANT’ deyimi, SELECT, UPDATE, DELETE veya EXECUTE gibi izinleri vermek için kullanılır. Tablodaki verileri okumak için erişim izni verebilirken tabloyu değiştirmeyi sınırlamak için de kullanılabilir. Bu deyimler sütun bazında da kullanılabilir.

Uygulama kullanıcılarında istisnai durumlar vardır. Uygulama kullanıcılarının, uygulamanın altındaki şemada tüm tablolara yetkisi olabilir. Bunun kontrolü, sunucu üzerinden gelen uygulama kullanıcılarının ilgili IP adresinden gelip gelmediği ile yapılır. Başka bir IP’ den gelmesinin engellenmesi gerekli tetikleyici yardımıyla yapılabilir. Diğer yandan, uygulama kullanıcısı veri tabanı yöneticisi kadar tam yetkiye sahip olmamalıdır. Sistemde, veri tabanı kullanıcısı dışında kimlerin tam yetkiye sahip olduğuna bakarken uygulama kullanıcıları dikkate alınırken, ‘SELECT’ yetkisi dışında kimlerin yetkisi olduğuna bakarken bu kullanıcılar hariç tutulur.

11) Veri tabanı izinlerinin gerekli düzeyde verilmiş veya kaldırılmış olduğu kontrol edilir.

Veri tabanı izinleri doğru şekilde kısıtlanmazsa, kritik verilere uygun olmayan erişimler olabilir. Güvenlik için ideal olan, yetkilerin sadece ihtiyaca uygun olarak verilmesidir. Yetki, özel olarak gerek duyulmuyorsa, verilmemelidir.

Nasıl Yapılır?

Veri tabanı yöneticisi ile yapılan görüşmeyle, hangi kullanıcının hangi verilere erişmesi gerektiği belirlenir. Bazı hesaplar verilere erişmek için web uygulaması tarafından, bazı hesaplar toplu (batch) işlerde kullanılabilir. Erişim gerektirmeyen hesaplar kilitlenmeli, pasif edilmeli hatta silinmelidir.

Tablolara SELECT Yetkisi Dışında Yetkisi Olan Kullanıcıların Tespiti

İdeal olan, veri tabanı yöneticisi dışında kalan kullanıcıların update, delete, alter gibi kritik yetkilerinin olmamasıdır. İlgili sorguyla gelen sonuçta böyle yetkilere sahip kullanıcılar çıkarsa denetçiyi ikna edecek şekilde gerekçeler açıklanmalıdır.

Örnek:

Oracle veri tabanında, aşağıdaki sorgu ile ‘SELECT’ yetkisi dışındaki kullanıcılar ve bu kullanıcıların hangi tabloya ne yetkisi olduğu listelenebilir.

SELECT grantee, table_name, privilege FROM dba_tab_privs WHERE privilege NOT IN('SELECT')

Önemli Rollere Sahip Olan Kullanıcıların Tespiti

Bu roller genelde ‘DROP ANY TABLE’, ‘ALTER SYSTEM’, ‘CREATE ANY

TABLE', 'DBA', 'SELECT CATALOG ROLE' gibi yetkilerdir. Bu rollere sahip kötü niyetli bir kullanıcı veri tabanının bütünlüğüne zarar verebileceği için bu yetkilere sahip olan kullanıcıların listesi alınıp nedenleri sorulmalıdır.

Örnek:

Yukarıda bahsedilen rollere sahip kullanıcılar Oracle veri tabanında, aşağıdaki sorgu ile bulunabilir.

```
SELECT * FROM dba_role_privs
```

Sistemde Veri Tabanı Yöneticisi Kadar Yetkiye Sahip Kullanıcıların Tespiti

Sysdba, Oracle veri tabanında, en yetkili kullanıcıdır. Sistemde bu yetkiye veya diğer platformlarda bu kullanıcıya karşılık gelen yetkilerin sürekli kontrol edilmesi gerekir. Veri tabanı yöneticisi dışında, tespit edilmesi halinde böyle bir kullanıcının yetkisi kısıtlanmalıdır.

Örnek:

Oracle veri tabanında, yukarıda belirtilen yetkili kullanıcıların listesi aşağıdaki sorgu ile çekilebilir.

```
SELECT * FROM v$pwfile_users;
```

Tablo Alanlarında (Tablespace) Sınırsız Yetkisi Olan Kullanıcıların Tespiti;

Tablo alanı bir veri tabanında mantıksal veri dosyaları grubudur. İçinde tablolar, indeksler, tetikleyiciler gibi nesneler tutulur. Veri dosyaları olan klasör gibi düşünülebilir. Gerçek ortamlarda tablo alanlarına sadece uygulama kullanıcısının yetkisi olması beklenmektedir. Ancak istisna durumlar söz konusu olduğunda bu yetkilerin

dikkatli verilip alınması gerekir.

Örnek:

Oracle veri tabanında, sınırsız tablo alanı yetkisi dba_sys_privs tablosunda tutulur. Sorgusu aşağıdaki şekilde olabilir;

SELECT grantee, privilege FROM dba_sys_privs WHERE privilege = 'UNLIMITED TABLESPACE';

12) Gruplar veya roller yerine bireylere verilen veri tabanı yetkileri kontrol edilir.

İdeal olarak, veri tabanında gruplar veya roller oluşturularak yetkiler bunlara verilir. Bireysel yetkiler verilirken, hesaplar bu grup ve rollere dahil edilir. Yetkiler için rolleri ve grupları kullanmak yönetimsel hataları azaltır ve güvenlik kontrolleri yönetimini kolaylaştırır. Yeni yetkilerin verilmesi gerektiğinde, birçok hesaba yetki vermek yerine tek bir gruba verilebilir. Kullanıcının işi değiştiğinde, bağlı olduğu gruptan çıkarılarak yeni kişilerin buraya dahil edilmesi sağlanır.

Nasıl Yapılır?

Veri tabanı sözlüğünden yetkilerin listesi çıkarılır. Bir hesaba veya kullanıcıya verilen yetkiler incelenir. Rollere veya gruplara verilen yetkiler kontrol edilir. Bireysel kullanıcıları, gerektiğinde rol veya gruplara atayarak yetkiler verilebilir.

Ayrıca hangisine yetki verileceğini belirlemek için roller/gruplar ve kullanıcı/hesaplar listesinin indirilmesi gerekecektir. Kullanıcıların ve grupların listesi veri tabanı sözlüğünde bulunmaktadır.

Örnek:

Oracle veri tabanındaki kullanıcılara ve rollere verilmiş yetkiler aşağıdaki sorgu ile listelenebilir;

```
SELECT * FROM DBA_ROLE_PRIVS
```

13) Veri tabanı yetkilerinin dolaylı olarak yanlış verilmediği kontrol edilir.

Veri tabanı izinleri birçok kaynaktan verilebilir. Örneğin, bir nesnenin sahibi, veri tabanındaki bu nesne üzerinde tam kontrol imkanı verebilir. 'SELECT ANY TABLE' gibi ayrıcalıklar, tüm verilere erişim sağlar ve verilere yetkisiz erişime neden olabilir. Veri tabanında dolaylı olarak nasıl yetki verildiği tam olarak anlaşılmazsa, izinler planlandığı şekilde verilmeyebilir.

Nasıl Yapılır?

Kullanılan veri tabanı platformunun özel yetki modelleri gözden geçirilir ve yetkilerin uygun olarak devredildiği doğrulanır. Ayrıca, veriye erişim sağlayan, 'SELECT ANY TABLE' gibi sistem yetkileri incelenir. Bu gibi izinlerin kullanılmasına izin verilememesi için izinler dokümante edilir.

Örnek:

Oracle veri tabanında yukarıda bahsedilen ayrıcalıklı yetkiler aşağıdaki sorgu ile listelenebilir;

```
SELECT * FROM DBA_SYS_PRIVS
WHERE
PRIVILEGE='CREATE USER' OR
PRIVILEGE='BECOME USER' OR
```

PRIVILEGE='ALTER USER' OR
PRIVILEGE='DROP USER' OR
PRIVILEGE='CREATE ROLE' OR
PRIVILEGE='ALTER ANY ROLE' OR
PRIVILEGE='DROP ANY ROLE' OR
PRIVILEGE='GRANT ANY ROLE' OR
PRIVILEGE='CREATE PROFILE' OR
PRIVILEGE='ALTER PROFILE' OR
PRIVILEGE='DROP PROFILE' OR
PRIVILEGE='SELECT ANY TABLE' OR
PRIVILEGE='CREATE ANY TABLE' OR
PRIVILEGE='ALTER ANY TABLE' OR
PRIVILEGE='DROP ANY TABLE' OR
PRIVILEGE='INSERT ANY TABLE' OR
PRIVILEGE='UPDATE ANY TABLE' OR
PRIVILEGE='DELETE ANY TABLE' OR
PRIVILEGE='CREATE ANY PROCEDURE' OR
PRIVILEGE='ALTER ANY PROCEDURE' OR
PRIVILEGE='DROP ANY PROCEDURE' OR
PRIVILEGE='CREATE ANY TRIGGER' OR
PRIVILEGE='ALTER ANY TRIGGER' OR
PRIVILEGE='DROP ANY TRIGGER' OR
PRIVILEGE='CREATE TABLESPACE' OR
PRIVILEGE='ALTER TABLESPACE' OR
PRIVILEGE='DROP TABLESPACES' OR
PRIVILEGE='ALTER DATABASE' OR
PRIVILEGE='ALTER SYSTEM'

With Admin Option Yetkisi Verilen Kullanıcıların Tespiti

Grant komutu ile kullanıcılara yetki verilirken sonuna with admin option koşulu konulduğunda verilen yetki başka bir kullanıcıya devredilebilir. Bu durumun tespiti için

gerekli komutlar yazılır ve bu yetkiye sahip kullanıcılar bulunur.

Örnek:

Oracle veri tabanında, yukarıda bahsedilen durumdaki kullanıcılar aşağıdaki sorgu ile listelenebilir;

SELECT grantee FROM dba_role_privs WHERE admin_option = 'YES' UNION SELECT grantee FROM dba_sys_privs WHERE admin_option = 'YES'

14) Saklı yordamlarda çalıştırılan dinamik SQL'ler kontrol edilir.

Dinamik SQL, çalışma zamanında dinamik olarak SQL ifadeleri oluşturmayı sağlayan bir programlama tekniğidir. Dinamik SQL, standart (veya statik) SQL'den farklı olarak, program ifadelerinin otomatik oluşturulmasını ve yürütülmesini kolaylaştıran gelişmiş bir SQL biçimidir. Bu, değişken veri tabanlarına, koşullara veya sunuculara göre ayar yapabilmek için kod yazmak gerektiğinde yardımcı olabilir. Ayrıca birçok kez tekrarlanan görevleri otomatikleştirmeyi kolaylaştırır. Programcı tarafından girilebilir veya programın kendisi tarafından oluşturulabilir, ancak statik SQL ifadelerinin aksine, kaynak programa gömülü değildir. Ayrıca statik SQL ifadelerinin aksine, dinamik SQL ifadeleri bir çalışmadan (execution) diğerine değişebilir. Dinamik SQL deyimleri, nispeten az programlama tecrübesi olan kişiler tarafından yazılabilir, çünkü program asıl kodun oluşturulmasının çoğunu yapar. Çok fazla dinamik SQL çalışırsa performans kayıpları meydana çıkabilir. Bunun nedeni çalışma planının (execution plan) tekrar tekrar oluşturulmasıdır. Ayrıca, SQL enjeksiyonu saldırılarına karşı fırsat verebilir. Bununla birlikte dinamik SQL içinde referans verilen tablolara genellikle erişim izni verilmelidir ki bu da kaçınılması gereken bir durumdur.

Bir nesneye erişim, saklı yordamları veya fonksiyonları çalıştırarak da elde edilebilir. Microsoft SQL Server' da, saklı yordam sahibi tarafından kod çalıştırıldığında başka bir nesneye erişimi izin verilir. Oracle' da saklı yordamı çalıştırmak, nesnelere saklı

yordam sahibi gibi erişime izin verir. Bu gibi durumlar tehlikeli olabilir.

Nasıl Yapılır?

Veri tabanı yöneticisinin yardımı ile saklı yordamlara göz atılır. Özellikle, SQL enjeksiyonu veya dinamik SQL gibi konulara bakılır. Yüksek ayrıcalıklarla çalışan yordamlarda dinamik SQL kullanımı kısıtlanır. Buna ek olarak, bu yüksek ayrıcalıklarla saklı yordamlara erişimin loglandığından emin olunmalıdır.

15) Tablolara satır düzeyinde erişimin doğru uygulandığı kontrol edilir.

İlişkisel veri tabanlarında yetkiler tablo veya sütun bazında verilmektedir. Satır bazında yetkiler ise, bir kullanıcıya 'SELECT' yetkisi verildiğinde kullanıcı tablodaki tüm satırları okuyabilmektedir.

Bu sorunu çözmeye yardımcı çeşitli teknolojiler kullanılmaktadır. Örneğin, Oracle belirli satırlara erişimi sınırlamak için kullanabileceğimiz özel sanal veri tabanları (VPDs) sunmaktadır. Ayrıca, görünümleri (view) kullanarak satır sınırlaması yapmak mümkündür. Tablolara erişim için saklı yordamları kullanmak da pratik bir yaklaşımdır.

Nasıl Yapılır?

Bu durum veri tabanı yöneticisi ve uygulama sahibi arasında ortak bir çalışmayla özellikle daha büyük ortamlar için yapılabilir. Uygun yöneticilerle satır bazlı yetki kontrollerinin metodu hakkında görüşülür. Eğer uygulama veya saklı yordam erişim sağlıyorsa, kullanıcının uygun bir yetkilendirme olmadan bir tabloda bulunan verilere erişemediğinden emin olunmalıdır. Kullanıcının istendiği gibi yetkilere sahip olduğunu doğrulamak için veri tabanına kullanıcının hesabı üzerinden erişilir.

Görünümler (view) ve Yetki Kontrolleri

Veri tabanında tanımlanan görünümlerin hangi verileri görüntülediğini, bu görünümleri kimlerin görüntülediğini kontrol etmek gereklidir.

Örnek:

Veri tabanında bulunan tüm görünümleri listelemek için aşağıdaki sorgu yazılabilir;

SELECT view_name, owner FROM all_views;

Bu görünümlere kimlerin ne yetkisi olduğu şu sorgu ile bulunabilir;

SELECT v.view_name, p.privilege, p.grantee FROM dba_views v JOIN dba_tab_privs p ON p.table_name = v.view_name;

Bu sorgudaki dba_tab_privs tablosundaki table_name sütunu tablo, görünüm, indeks gibi herhangi bir nesne olabilir.

16) Gereksiz PUBLIC olan yetkilerinin iptal edildiği kontrol edilir.

Bir veri tabanındaki saklı yordamlar veya fonksiyonların birçoğu varsayılan olarak PUBLIC grubuna dahildir. Her veri tabanının PUBLIC yetkileri uygulamada biraz farklılık gösterse de genelde veri tabanında herkes görebilir şeklindedir, yani, PUBLIC olarak verilen izinlerin herkes için geçerli olduğu anlamına gelmektedir.

Bu durum, veri tabanında birçok güvenlik sorununa yol açmaktadır. Gömülü yazılımların birçoğu tehlikeli görünmeyebilir ve sıradan kullanıcılar burada pek kullanılmaz. İdeal olan, gerekmediği sürece tüm erişimleri kısıtlamaktır ve zorunlu olmadığı sürece herhangi bir kullanıcıya yetki verilmemelidir. PUBLIC yetkisi için de bu kurallar geçerlidir.

Gerekli olan izinlerin iptal edilmesi, programların çalışmasını aksatabileceği unutulmamalıdır. Bilinçsizce tüm PUBLIC yetkilerin iptal edilmesi programların çalışmasını durdurabilir.

Nasıl Yapılır?

Tüm izinlerin listesi toplanarak başlanır ve özellikle PUBLIC yetkilerine bakılır. Veri tabanının hangi prosedür ve özelliklerinin kullanıldığını veya kullanılabileceğini veri tabanı yöneticisi ile görüşüp, ardından, gerekli olmayan nesnelerden izinlerin iptal edilmesinin ne kadar risk getireceği belirlenir. İzinlerin iptal edilmesi herkes tarafından uygunsa, iptal edilmesi sağlanır. Beklenmedik bir durumla karşı karşıya kalma ihtimalini düşünüp yedekleme alınmalı veya bu yetkileri geri almak için komut dosyası hazırlanmalıdır.

Örnek:

Oracle veri tabanında varsayılan olarak EXECUTE yetkisi PUBLIC olan haklar şunlardır;

SYS.DBMS_EXPORT_EXTENSION, UTL_FILE, UTL_SMTP, UTL_HTTP, UTL_TCP, SYS.DBMS_RANDOM

Bu belirtilen paketler kullanılarak veri tabanı yöneticisi (DBA) yetkisine sahip kullanıcı yaratılması gibi riskler doğabilir. Bu yüzden bu paketlerin yetkileri REVOKE ile alınmalıdır. Bu yetkilerin alınırken neleri etkileyeceği de iyi sorgulanmalıdır.

6.3.4 Veri şifreleme

Veri şifreleme üç farklı bölgeye veya duruma uygulanır. *Hareket halindeki veriler*; ağ üzerinden aktarılan veriler anlamına gelir ve genellikle SSL gibi protokolleri kullanarak şifrelenirler. *Duran veriler*; bir veri tabanındaki gibi depolama halinde duran veriler

anlamına gelir ve AES gibi algoritmalarla şifrelenebilir. *Kullanılan veriler*; uygulamalarda veri işleme anlamına gelmektedir.

17) Ağ şifrelemesinin uygulandığı kontrol edilir.

Ağ şifreleme iki temel amaca hizmet eder: kimlik bilgilerinin ağ üzerinden geçerken korunması ve veri tabanındaki gerçek verilerin ağ üzerinde dolaşırken korunması. Ağ güvenli bir ortam değildir; IP adresleri sahte olabilir, ağ trafiği yönlendirilebilir ve algılanabilir (sniffed). Ağ trafiğinin yalnızca dış ağ üzerinden değil aynı zamanda intranet üzerinden de şifrelenmesi kritik önem taşır.

Nasıl Yapılır?

Ağ ve istemci sürücülerinin, SSL gibi protokolleri kullanarak ağ trafiğini şifrelemeyi destekleyecek şekilde yapılandırıldığı doğrulanır. Hem istemcide hem de veri tabanındaki ayarlar doğrulanır. Ayrıca güvenlik duvarı kullanımıyla erişimler port veya IP bazlı engellenebilir, uygulama katmanında ağ trafiği incelenerek daha güvenli bağlantı oluşması sağlanabilir. Dışarıdan veri tabanına erişilmek isteniyorsa bu erişimler VPN kullanılarak daha güvenli ortamda yapılır.

18) Duran verilerin şifrelenmesinin, gerektiğinde uygulandığı kontrol edilir.

Duran verilerin şifrelenmesi, verilerin veri tabanında tutulduğu şekliyle şifrelenmesini içerir. Diskteki veya veri tabanındaki verilerin ömrü ağdaki verilerin ömründen çok daha uzun olduğundan, kuşkusuz duran verilerin şifrelenmesi diğer şifreleme durumlarından daha önemlidir. Zaten verilerin en çok nereden çalındığına bakılırsa, ağ geçişi sırasında değil, doğrudan veri tabanından olduğu görülür.

Nasıl Yapılır?

Şifrelenmesi gereken verinin doğru şifrelendiği kontrol edilir. Aynı zamanda, şifreleme

anahtarlarının nerede saklandığına bakılır çünkü şifrelemenin gücü şifreleme anahtarını koruma gücüne bağlıdır. Eğer şifreleme anahtarı şifrelenen veriyle birlikte tutuluyorsa, saldırgan şifreleme anahtarını elde ederek güvenliği rahatça aşabilir.

Şifreleme anahtarı yönetiminin felaket kurtarma planına dahil edildiğinden emin olunur. Veri tabanı yöneticisi, yedekleme prosedürlerine şifreleme özelliklerini eklerken şifre anahtarı yönetimini de dahil etmelidir. Anahtarların yedeklenmemesi, yedeklenen veri tabanının kurtarılamamasına neden olabilir.

Örnek:

Oracle veri tabanı verileri bir dosya sisteminde tutmaktadır. Hassas verilerin veri tabanında şifreli olarak tutulması yetkisiz erişimlere engel olmak için gerekli olabilir. Şifreleme için Oracle Advanced Security; ağdaki, depolama ortamındaki ve veri tabanı içindeki hassas verileri yetkisiz erişimlerden korumaya yardımcı olur. Oracle Advanced Security' nin önemli bir bileşeni olan Transparent Data Encryption, uygulamalarda değişiklik yapmadan hassas bilgileri korumak için gelişmiş veri tabanı şifreleme çözümünü sunmaktadır. Bu çözüm, veriyi diskte şifreli olarak tuttuğu için veri SQL sorguları ile çekildiğinde düz metine çevrilir. Bu yüzden veri yine veri tabanı yöneticileri tarafından görüntülenebilir. Fakat disklerin veya dosyaların çalınması gibi yetkisiz erişimlere karşı koruma sağlar. Şifreleme yapılmasının sistem kaynaklarını tüketmesi maliyeti de bulunmaktadır.

6.3.5 İzleme ve yönetim

Mevzuatlar, hassas verilere erişimin düzgün bir şekilde izlenmesi gerektiğini söyler. Sarbanes-Oxley gibi bu düzenlemeler, hassas verileri barındıran şirketler üzerinde olumlu ve önemli bir etkiye sahiptir.

19) Veri tabanı faaliyet izleme (monitoring) ve denetiminin (auditing) doğru yapıldığı kontrol edilir.

Veri tabanında saklanan veriler şirketler için çok kritik olduğundan, kötü niyetli saldırıları ve verilerin uygunsuz kullanımını belirlemek için izleme yapılmalıdır.

Faaliyetleri izlemek için aşağıdaki yöntemler kullanılabilir:

- Veri tabanına ait olan denetim sistemini etkinleştirme
- Veri tabanı faaliyetlerinin ağ trafiğini izleme
- İşlem günlüklerini (transaction logs) gözden geçirme

Her bir yöntemin zayıf ve güçlü yönleri vardır. Örneğin, ilk yöntem nispeten ucuzdur çünkü veri tabanına entegredir. Diğer yöntemler daha pahalıdır fakat ilk yöntemin yeterli olmadığı durumlarda ve gereksinimleri karşılamada diğer yöntemler etkili olabilmektedir.

Nasıl Yapılır?

Denetim birçok biçimde olabilir:

- **Erişim ve kimlik doğrulama denetimi:** Hangi sistemlere kimin, nasıl, ne zaman eriştiğini belirleme
- **Kullanıcı ve yönetici denetimi:** Yöneticiler ve kullanıcılar tarafından gerçekleştirilen faaliyetleri belirleme
- **Şüpheli faaliyetlerin denetimi:** Hassas verilere veya kritik sistemlere şüpheli, olağan dışı veya anormal erişimleri belirleme ve işaretleme
- **Güvenlik açığı ve tehditlerin denetimi:** Veri tabanındaki güvenlik açıklarını tespit etme ve bunlardan yararlanmaya çalışan kullanıcıları izleme
- **Değişiklik denetimi:** Veri tabanı, yapılandırma ayarları, şemalar, kullanıcılar, ayrıcalıklar için temel ilkeler oluşturulur ve bu ilkelerden sapmalar bulunur ve

takip edilir.

Uygulanan izleme yöntemleri veri tabanı yöneticisi ile gözden geçirilir ve verilerin hassasiyeti görüşülür. Faaliyet izleme, veri tabanında saklanan bilgilerin ticari değeri ile ve kuruluşun politikaları, gereksinimleri ile uyumlu olmalıdır.

Veri tabanındaki hassas verilerin listesi gözden geçirilir ve denetimin, hassas veriler için düzgün bir şekilde aktif olduğu doğrulanır. İzleme sisteminin bu tür olayları denetleme kabiliyeti olduğunu göstermek için belirli bir süre hassas işlemlere (transaction) bakılır.

Örnek:

Veri tabanlarının kendilerine özgü denetim parametreleri vardır. Oracle veri tabanında bu audit_trail ve audit_sysoperations parametreleridir ve bunlar aktif edilerek sys kullanıcısının yapmış olduğu tüm işlemler audit kapsamına alınmış olur. Burada parametrelerin durumlarına bakmak için aşağıdaki sorgu kullanılarak bu parametreler yönetilebilir.

SELECT name, value FROM v$parameter WHERE NAME LIKE '%audit%'

Hangi Yetkilerin Denetlendiğinin (Audit) Tespiti

Denetim parametreleri açık olabilir fakat nelerin denetlendiğini bilmemiz gerekir. Hangi yetkilerin denetlendiğinin (audit) tespiti gerekli sorgu yazılarak bulunur.

Şu unutulmamalıdır ki her şeyi denetlemek (audit) doğru bir yaklaşım değildir. Bunun sonucunda elimizde binlerce veri olabilir. Mesela Alter any index yetkisinin denetlenmesine birçok organizasyonda gerek yoktur. İndeksi değiştirmekle verinin üzerinde bir değişiklik yapılamayacağından bunu denetlememize gerek olmayabilir. Hangi verilerin, hangi detayda logunun tutulacağı, bu log raporlarının kimlere iletileceği

konuları organizasyon tarafından verilecek bir karardır.

Ayrıca yetkilerden başka hangi nesnelerin ve komutların denetlendiğinin (audit) tespit edilmesi gereklidir. Bunlara örnek olarak; table, stored procedure, alter any table, create user gibi nesne ve komutlar verilebilir.

Örnek:

Oracle veri tabanında, hangi yetkiler kullanıldığında denetleniyor bilgisi şu sorguyla yapılabilir;

```
SELECT * FROM sys.dba_priv_audit_opts;
```

Hangi nesnelerin denetleniyor bilgisi şu sorguyla yapılabilir;

```
SELECT * FROM sys.dba_obj_audit_opts;
```

Veri tabanında çalıştırılan komutların hangilerinin denetlendiği bilgisi de şu sorguyla yapılabilir;

```
SELECT * FROM dba_stmt_audit_opts;
```

20) Var olan ve beklenen iş gereksinimleri için veri tabanı kapasitesinin nasıl yönetildiği kontrol edilir.

Veri tabanları için teknik gereksinimler ve iş gereksinimleri hızlı bir şekilde değişmektedir. Bunlara; altyapı değişiklikleri, iş ilişkileri, müşteri ihtiyaçları, mevzuat gereksinimleri gibi ihtiyaçlar örnek verilebilir. Yetersiz veri tabanı altyapısı, işletmenin önemli verilerini kaybetmesine ve kritik işlerinin aksamasına sebep olabilir.

Nasıl Yapılır?

Kapasite gereksinimlerinin dokümante edildiği ve müşteri isteklerini karşıladığı doğrulanır. Kapasite kullanımını izlemek için süreçler gözden geçirilir, tanımlanan eşik değerleri (threshold) aşıldığında not alınır. Gereksinimler, depolama ortamından sorumlu olan aynı ekip tarafından kısmen değerlendirilebilir veya tutulabilir. Kapasite kullanımı belirlenen eşikleri aştığında yanıt vermek ve gereğini yapmak için süreçler değerlendirilir. Mevcut veri tabanı gereksinimleri ve beklenen büyümeyi belirlemek için kullanılan yöntemler görüşülür. Yönetici ile büyüme planları görüşülür ve donanımın; performans, kapasite ve diğer gereksinimleri karşılayabildiği doğrulanır.

21) Veri tabanı ortamında var olan ve beklenen iş gereksinimlerini desteklemek için performansın nasıl yönetildiği ve izlendiği kontrol edilir.

Veri tabanı performansı; fiziksel depolama ortamı, iletişim protokolleri, ağ, veri boyutu, işlemci, bellek, depolama mimarisi, şifreleme stratejileri ve diğer özellikler de dahil olmak üzere çeşitli faktörlere bağlıdır. Yetersiz veri tabanı altyapısı, işletmeyi önemli verileri kaybetme riski altına sokar ve daha iyi performans göstermesini engeller.

Nasıl Yapılır?

Veri tabanı mimarisinde işlemci, bellek ve G/Ç ağ bant genişliği yüklerinin periyodik performans incelemeleri, mimaride artan gerilimi tanımlamak için yapılmalıdır. Performans gereksinimlerinin dokümante edildiği ve müşterilere cevap verebildiği doğrulanır. Performansı izlemek için süreçler gözden geçirilir ve performans belirlenmiş eşik değerinin altına düştüğünde not edilir. Performans, belirlenen eşik değerinin altına düştüğünde yanıt vermek ve gereğini yapmak için süreçler değerlendirilir. Mevcut performans gereksinimleri ve beklenen değişiklikleri belirlemek için kullanılan yöntemler tartışılır.

Denetimde, kapasite yönetimini ve performans planlamasını gözden geçirmek kritik

adımlardır. Yöneticinin uygun bir kapasite yönetim planına sahip olduğundan emin olunur ve performans ihtiyaçlarının kuruluş için uygun olduğu doğrulanır.

22) Yedeklerin başarılı bir şekilde alındığı kontrol edilir.

Veri tabanı yöneticisi, veri tabanlarının hangi yöntemle yedekleneceğini, yedeklemeyi yapacak sorguların hazırlanmasını sağlamakla; yedeklemelerden sorumlu BT çalışanı ise veri tabanı yöneticisinin hazırladığı yapı ve sağladığı bilgiler doğrultusunda veri tabanlarını düzenli olarak kartuşlara yedeklemekle yükümlüdür.

Nasıl Yapılır?

Organizasyonun yedekleme ve yedekten dönme prosedürleri incelenip nasıl yapılması gerektiğine bakılır. Bunun için araçlar kullanılıyorsa bu aracın loglarından veya gerekli sorgular yazılarak loglara ulaşılabilir. Özellikle üretim ortamında alınan yedeklerin durumu başarılı olarak gözükmesi gerekir.

Örnek:

Oracle veri tabanında yedekleme yapılmasını ve yedekten dönülmesini sağlayan araçlardan en önemlisi RMAN (Recovery Manager)'dır. Standart olarak bir yedekleme şöyle alınır;

- Oracle veri tabanının RMAN aracı kullanılarak veri tabanının en tutarlı ve performanslı şekilde yedeklenmesi sağlanır.
- RMAN kullanımı için katalog veri tabanı ya da yedeklenen veri tabanının kontrol dosyaları kullanılabilir. Katalog kullanımı yedekten dönme işlemlerini ve takibi kolaylaştırdığından tercih edilmelidir.
- Yedeklenecek veri tabanı tüm değişikliklerin çevrimiçi olarak kaydedilmesini sağlayacak olan 'ARCHIVELOG' durumuna getirilir.
- Veri tabanının tüm içeriğinin ve değişikliklerin istenilen nitelikte yedeklenmesi

için RMAN'ın kendi sorgu dilinde yedeklemeyi yapacak olan sorgu, veri tabanı yöneticisi tarafından hazırlanır.

- Hangi tip (Level 0,1,2,…) RMAN yedeği alınacağını belirleyen ve buna göre RMAN sorgularını otomatik olarak yedekleme başlamadan önce hazırlayan işletim sistemi sorguları veri tabanı yöneticisi tarafından hazırlanır.
- Sunucuda yeterli disk alanı varsa, hazırlanan RMAN sorgusu ile bu diske yedekleme yapacak, sonucunu veri tabanı yöneticilerine mail yoluyla bildirecek sorgu veri tabanı yöneticileri tarafından hazırlanır ve istenilen zamanda çalışacak şekilde zamanlanır. Diske alınan bu yedek BT yedekleme çalışanı tarafından kartuşa yedeklenir.

Oracle veri tabanında yedeklerin günlük olarak doğru alındığını öğrenebilmek için gerekli sorgu aşağıdaki gibi olabilir;

```
SELECT *
FROM (  SELECT start_time,
                end_time,
                output_device_type,
                input_type,
                status,
                output_bytes_display,
                time_taken_display
           FROM V$RMAN_BACKUP_JOB_DETAILS
           ORDER BY 1 DESC);
```

23) Yedekleme ortamlarının saldırılara karşı önlem alındığı kontrol edilir.

Verilerin ifşa edilmesi çeşitli faktörlerden kaynaklanıyor olabilir. Örneğin, veri yedekleme cihazlarının çalınması veya hassas verilerin kopyalarına (yedeklerine) yetkisiz kişilerce erişilmesi mümkündür. Veri ifşasının ana nedeni, veri tabanı kopyalarını korumak için güvenli bir mekanizmanın bulunmamasıdır. Çoğu durumda, bu olayı yaşayan organizasyonlar, tüm veri tabanlarının ve ilgili yedeklemelerinin bir

envanterine sahip değildir. Her veri tabanı bir bilgi madenidir ve erişim kontrolünü gerektiren hassas veriler içerirler. Güncel olmayan veri tabanları, organizasyonlar için zayıf noktalardan biridir. Çok sayıda güvenlik denetimi, veri tabanlarının eski kopyasını içeren ortamların uygun şekilde yönetilmediğini ortaya çıkarmaktadır.

Nasıl Yapılır?

Bir organizasyon, altyapısındaki tüm veri tabanlarını ve içerdiği verileri tanımlamalıdır; her arşiv için erişim ayrıcalıklarını tanımlamalı ve oluşturulmasından depolama için kullanılan medya desteğinin elden çıkarılmasına kadar, kullanım ömrü boyunca her faaliyetin izini sürmelidir.

Veri sınıflandırması, arşivlerdeki hassas bilgileri tanımlayan bir zorunluluktur. Aynı bilginin zararsız görünmesine rağmen, eğer bir araya getirildiklerinde hassas verileri açığa çıkarsalar bile, gizli bilgileri açığa çıkarabilecek veri kombinasyonlarını ortaya çıkarmak da şiddetle önerilmektedir. Hassas veri ve erişim kontrollerinin yer aldığı doğru bir veri tabanı envanteri, kurumsal veri erişim politikalarına uygun olarak ayarlanmalıdır.

Yedekleme verilerinin ifşa edilmesini önlemek için yedeklemeleri şifrelemek de gereklidir.

Örnek:

Veri tabanını atlatma tehditleri, işletim sistemi dosyalarını ve yedekleme ortamlarını hedefler. Bu ortamları hedeflemek, saldırganın işini kolaylaştırır. Herhangi bir veri tabanı erişimi gerekmez, varsa daha az denetim kaydı oluşturulur ve uygulama erişim kontrollerinin yanı sıra ilgili veri tabanları da tamamen atlatılır. Veri tabanı atlatma tehditlerine karşı kullanılan en yaygın teknolojilerden biri şifrelemedir.

Şifreleme için Oracle Advanced Security' nin önemli bir bileşeni olan Transparent Data

Encryption, gelişmiş veri tabanı şifreleme çözümünü sunmaktadır. Bu çözüm, veri şifreleme bölümü altında da anlatılmıştır.

Beş ana başlık altında verilen kontroller, tezin üçüncü bölümünde anlatılan en çok rastlanan veri tabanı güvenlik tehditlerini önleyen kontrolleri de kapsamaktadır. Bu tehditler ve kontrol eşleştirmeleri Çizelge 6.2' de verilmiştir. Çizelgede kontrollerin ana başlıkları verilmiş, bunların altında kontrol maddelerinin numaraları yazılmıştır.

Çizelge 6.2 Risk ve Kontrol Eşleşmeleri

Tehditler	**Kontroller**				
	Kurulum ve Genel Kontroller	İşletim Sistemi Güvenliği Kontrolleri	Kullanıcı Hesap ve İzin Yönetimi Kontrolleri	Veri Şifreleme	İzleme ve Yönetim
Aşırı ve Kullanılmayan Yetkiler			7, 8, 9, 11		
Yetki Suistimali			7, 11		
SQL Enjeksiyonu	1, 2, 3	4	10, 14, 16	17	
Kötü Niyetli Yazılımlar	1, 2			17	
Zayıf Denetim İzi					19
Depolama Ortamı İfşası				18	22, 23
Zayıf, Yanlış Yapılandırılmış Veri Tabanlarının Suistimali	1, 2, 3	5, 6	8		
Yönetilmeyen Hassas Veriler	3				22, 23
Hizmet Durdurma Saldırısı				17	20, 21

7. SONUÇ

Bilgi toplumu olmak ve bilgi çağında yaşamak bilgiye ulaşımı kolay hale getirirken, bilginin amacına uygun olarak korunmasını bir o kadar zorlaştırmaktadır. Bilgiye duyulan ihtiyacın ve verilen önemin artması, bilgi teknolojilerinin gelişimini desteklerken, ilgili teknolojilerin ve bilgi sistemlerinin güvenilirliğini tartışma konusu haline getirmektedir.

Organizasyonlar, kurumsal iş süreçlerini gerçekleştirirken, rekabet ortamına ayak uydurmak ve faaliyet gösterdikleri sektörde farkındalık yaratmak için bilgi teknolojilerini kullanarak kurumsal bilgi varlıklarını korumayı hedeflemektedir. Bu noktada kurumsal iş süreçlerine ve hacmine bağlı olarak, bilgi işleme faaliyetleri artmakta dolayısıyla kullanılan bilgi sistemleri de karmaşık bir hal almaktadır. Söz konusu sistemlerin karmaşıklığı arttıkça, organizasyonlar kullandıkları bilgi sistemlerinin ne kadar güvenilir olduğu konusunda tereddüt yaşamakta ve kullanılan sistemlerin organizasyonun diğer iş süreçlerinde yeni risk faktörleri oluşturmasını engellemek, sistemlerin güvenilirliğinden emin olmak istemekte ve bu çerçevede bilgi sistemleri denetimi kaçınılmaz bir ihtiyaç olarak karşımıza çıkmaktadır.

BT altyapısının bir parçası olan BT genel kontrollerinin denetimi ise BT denetiminin temelini oluşturmaktadır. Çünkü verinin fiziksel olarak bulunduğu yer burasıdır. Bu nedenle hassas verileri güvenli bir şekilde saklamak, veri sızıntılarını önlemek ya da neden kaynaklandığını saptamak için BT altyapı kontrolleri denetimi ihtiyaç duyulan durumlarda yapılmalıdır. Denetimin amacına göre uygulanması gereken işlemler değişiklik gösterebilir. Bu çalışmadaki BT altyapı kontrollerinde yer alan veri tabanının denetimlerinde temel olarak uygulanması gerekli olduğu düşünülen konular ele alınmıştır.

Veri tabanı sistemleri, bilgi teknolojilerinin yoğun olarak kullanıldığı günümüzde verilerin saklandığı en önemli araçlarından biridir. Veri tabanları, verilerin, kendine has sorgu dilleriyle getirilmesi, silinmesi, güncellenmesi, eklenmesi gibi işlemleri yapmak amacıyla kullanılarak elektronik ortamda planlı bir şekilde tutulduğu sistemlerdir. Veri

tabanlarının etkili veri getirme ve saklama özellikleriyle hassas verileri hızlı ve doğru bir şekilde erişilebilir kılması, günümüzde bu sistemlerin kullanılmasını neredeyse zorunlu hale getirmiştir. Diğer yandan bu sistemlerin güvenliği ikinci planda kalabilmektedir. Veri tabanlarının, bir uygulamanın görünmeyen arka tarafında bulunması organizasyonlara, dışarıdan ulaşılması güç olarak hissettirmekte ve etkili güvenlik mekanizmalarının oluşmasının göz ardı edilmesine neden olabilmektedir. Teknolojinin gelişmesiyle yeni saldırı teknikleri de gelişmekte ve olası saldırılara karşı gerekli tedbirlerin alınmaması veya alınsa bile bunların uygulanmasının sistemin bütünlüğünü bozacağı kaygısı güvenlik açıklarına neden olabilmektedir.

İlk bölümde bu güvenlik açıklarının etkilerine, dünya çapında yaşanan en büyük veri çalınma olayları anlatılarak değinilmiştir. Bu olayların sonucunda, organizasyonlar sadece maddi hasar görmemiş aynı zamanda itibar kaybına da uğramıştır. Verilerin ve bunların tutulduğu veri tabanı sistemlerinin güvenliği sağlanmadıkça bu olaylar artarak devam etmektedir. Her ne kadar bu olaylara tam anlamıyla engel olunamasa da bu çalışmanın ikinci bölümünde bahsedilen en sık rastlanan veri tabanı güvenlik tehditlerini anlayarak ve bunların çözümüne yönelik atılacak adımlarla bu sistemlerin güvenliğini makul seviyeye çekmek mümkün olabilmektedir. Bilgi sistemlerinin ana varlığı olan verilerin gizliliği, bütünlüğü ve erişilebilirliği diğer bölümde veri güvenliği başlığı altında anlatılarak veri güvenliği kapsamının kavranması hedeflenmiştir.

Denetim yapılmadan önce doğrulanması gereken kontrollerin listesine sahip olunması denetimi daha etkin kılmaktadır. Veri tabanlarına ilişkin kontroller bu çalışmanın en önemli bölümünü oluşturmaktadır ve bu kontrollerin, bilgi sistemlerinde veri tabanlarını denetleyecek olan denetçiler, veri tabanı yöneticileri, sistem üzerinde değerlendirme yapacak kişiler veya veri tabanı güvenliği konusunda bilgi sahibi olmak isteyen kişiler tarafından kullanılması amaçlanmıştır. Veri tabanları diğer katmanlardan bağımsız çalışamayacağından uygulama, işletim sistemi ve ağ katmanlarının işleyişi hakkında da bilgi sahibi olunmalıdır. Zaten kontroller uygulanmaya başlandığında bu katmanların birbirleriyle ne kadar bağlantılı olduğu anlaşılmaktadır. Bu çalışmada bahsedilen kontroller sınıflandırılarak beş gruba ayrılarak bir metodoloji oluşturulmuştur. Ayrıca, anlatılan kontrollerin tamamı, her organizasyona uygulanamayabilir. Çünkü

organizasyonlar kendine özgü farklı sistemlere sahip olabilmektedir.

Türkiye'de yapılan bağımsız denetimlerde, bilgi sistemlerine ilişkin kontrollere bakılırken, bu çalışmada belirtilen her bir kontrolün uygulanması beklenmese de veri tabanlarına yönelik saldırıların her geçen gün artması, organizasyonlarda bulunan verilerin büyümesi ve değerlenmesi nedenleriyle gelecekte bu alanda yapılacak denetimlerin yaygınlaşması öngörülmektedir. Diğer taraftan herhangi bir otorite tarafından böyle kapsamlı kontrollere ilişkin hali hazırda bir yönetmelik yayımlanmamış olması nedeniyle bu tezin bu alanda yapılacak çalışmalara bir çerçeve oluşturması açısından önem arz etmesi beklenmektedir.

KAYNAKLAR

(2014). *2014 Cost of Data Breach Study.* Michigan: Ponemon Institute.

(2018). tom's guide: https://www.tomsguide.com/us/pictures-story/872-worst-data-breaches.html adresinden alındı

Coronel, C., & Morris, S. (2016). *Database Systems Design, Implementation and Management.*

IDC. (2011). *Worldwide Security Products 2011–2014 Forecast.*

Imperva. (2015). *Top Ten Database Security Threats.*

Kamu Bilgi Teknolojileri Denetim Rehberi. (2014). Ankara: İç Denetim Koordinasyon Kurulu.

Martin, J. (1977). *Managing the Database Environment.* New Jersey: Prentice-Hall.

McKendrick, J. (2014). *2014 Ioug Enterprise Data Security Survey.* Independent Oracle User Group.

Oluwatosin, H. S. (2014). Client-Server Model. *IOSR Journal of Computer Engineering* .

Pawar, R. G. (2015). SQL Injection Attacks. *1st International Conference (Special Issue)* .

Sayıştay. (2013). *Bilişim Sistemleri Denetimi Rehberi.* Ankara.

Yıldırım, S. (2017). *Bilgi Sistemleri Denetim Süreçleri.* İstanbul: Sermaye Piyasası Kurulu.

EK 1 PASSWORD VERIFY FUNCTION

```
CREATE OR REPLACE FUNCTION SYS.pass_verify_function
(username varchar2,
 password varchar2,
 old_password varchar2)
 RETURN boolean IS
  n boolean;
  m integer;
  differ integer;
  isdigit integer;
  islowercasechar  integer;
  isuppercasechar  integer;
  ispunct integer;
  isturkce boolean;
  digitarray varchar2(20);
  punctarray varchar2(25);
  lowercasechararray varchar2(26);
  uppercasechararray varchar2(26);
  turkcearray varchar2(40);

BEGIN
  digitarray  := '0123456789';
  lowercasechararray   := 'abcdefghijklmnopqrstuvwxyz';
  uppercasechararray   := 'ABCDEFGHIJKLMNOPQRSTUVWXYZ';
  punctarray  := '!"#$%&()``*+,-/:;<=>?_';
  turkcearray := 'üğışçöÜĞİŞÇÖ';
```

```
-- Check if the password is same as the username
IF NLS_LOWER(password) = NLS_LOWER(username) THEN
  raise_application_error(-20001, 'Password same as or similar to user');
END IF;

 -- Check for the minimum length of the password
IF length(password) < 8 THEN
   raise_application_error(-20002, 'Password length less than 8');
END IF;

-- Check for the space char in password
IF instr(password,' ') > 0 THEN
   raise_application_error(-20003, 'Password should not contain space character');
END IF;

-- Check if the password is too simple. A dictionary of words may be
-- maintained and a check may be made so as not to allow the words
-- that are too simple for the password.
 IF NLS_LOWER(password) IN ('welcome', 'database', 'account', 'user', 'password',
'oracle123', 'computer', 'abcd','asdf','12345678','akbank','akb123','akbank123',
'12345abc','abc12345',
 'abcde123','123abcde','ab123456','123456ab','abcdef12','12abcdef','abcd1234','1234abcd',
'Akbank12') THEN
   raise_application_error(-20004, 'Password is too simple');
END IF;

-- check for turkish char
```

```
isturkce:=true;
m := length(password);
FOR i IN 1..length(turkcearray) LOOP
  FOR j IN 1..m LOOP
    IF substr(password,j,1) = substr(turkcearray,i,1) THEN
      isturkce:=false;
    END IF;
  END LOOP;
END LOOP;

IF isturkce = FALSE THEN
  raise_application_error(-20005, 'Password should not contain Turkish chars');
END IF;

-- 1. Check for the digit
isdigit:=0;
m := length(password);
FOR i IN 1..10 LOOP
  FOR j IN 1..m LOOP
    IF substr(password,j,1) = substr(digitarray,i,1) THEN
      isdigit:=1;
       GOTO findlowercasechar;
    END IF;
  END LOOP;
END LOOP;

-- 2. Check for the lowercase character
```

```
<<findlowercasechar>>
islowercasechar:=0;
FOR i IN 1..length(lowercasechararray) LOOP
  FOR j IN 1..m LOOP
    IF substr(password,j,1) = substr(lowercasechararray,i,1) THEN
      islowercasechar:=1;
      GOTO finduppercasechar;
    END IF;
  END LOOP;
END LOOP;

-- 2. Check for the uppercase character
<<finduppercasechar>>
isuppercasechar:=0;
FOR i IN 1..length(uppercasechararray) LOOP
  FOR j IN 1..m LOOP
    IF substr(password,j,1) = substr(uppercasechararray,i,1) THEN
      isuppercasechar:=1;
       GOTO findpunct;
    END IF;
  END LOOP;
END LOOP;

-- 3. Check for the punctuation
<<findpunct>>
ispunct:=0;
FOR i IN 1..length(punctarray) LOOP
```

```
  FOR j IN 1..m LOOP
    IF substr(password,j,1) = substr(punctarray,i,1) THEN
      ispunct:=1;
       GOTO checking;
    END IF;
  END LOOP;
END LOOP;

<<checking>>
 IF (isdigit + islowercasechar + isuppercasechar + ispunct) < 3 THEN
    raise_application_error(-20006, 'Password should contain  3 of 4 character set in \
    (1.digit  2.lowercase letter  3.uppercase letter 4.punctuation mark)');
 END IF;

<<endsearch>>
 -- Check if the password differs from the username by at least
-- 3 letters
  differ := length(username) - length(password);
  IF abs(differ) < 3 THEN
    IF length(password) < length(username) THEN
      m := length(password);
    ELSE
      m := length(username);
    END IF;

    differ := abs(differ);
    FOR i IN 1..m LOOP
```

```
        IF NLS_LOWER(substr(password,i,1)) != NLS_LOWER(substr(username,i,1)) THEN
          differ := differ + 1;
        END IF;
      END LOOP;

      IF differ < 3 THEN
        raise_application_error(-20007, 'Password should differ from username by at \
        least 3 characters');
      END IF;
    END IF;

-- Check if the password differs from the previous password by at least
  -- 3 letters
  IF old_password IS NOT NULL THEN
    differ := length(old_password) - length(password);

    IF abs(differ) < 3 THEN
      IF length(password) < length(old_password) THEN
        m := length(password);
      ELSE
        m := length(old_password);
      END IF;

      differ := abs(differ);
      FOR i IN 1..m LOOP
        IF NLS_LOWER(substr(password,i,1)) != NLS_LOWER(substr(old_password,i,1))
THEN
```

```
          differ := differ + 1;
        END IF;
      END LOOP;

      IF differ < 3 THEN
        raise_application_error(-20008, 'Password should differ from previous password by at \
        least 3 characters');
      END IF;
    END IF;
  END IF;
  -- Everything is fine; return TRUE ;
  RETURN(TRUE);
END;
```

ÖZGEÇMİŞ

Adı Soyadı : Sercan CANBEY
Doğum Yeri : Bursa
Doğum Tarihi : 03/07/1986
Medeni Hali : Bekar
Yabancı Dili : İngilizce

Eğitim Durumu

Lise : Tavşanlı Anadolu Lisesi (2004)
Lisans : Eskişehir Osmangazi Üniversitesi – Mühendislik-Mimarlık Fakültesi - Bilgisayar Mühendisliği Bölümü (2009)
Vilnius Gediminas Technical University (Eylül 2007 – Şubat 2008) – Bilgisayar Mühendisliği (Erasmus Öğrenci Değişimi Programı)
Yüksek Lisans : Ankara Üniversitesi - Fen Bilimleri Enstitüsü - Bilgisayar Mühendisliği Anabilim Dalı (Eylül 2014 – Eylül 2019)

Çalıştığı Kurumlar

Sosyal Güvenlik Kurumu / Bilgi Sistemleri ve Siber Güvenlik Daire Başkanlığı – Sözleşmeli Programcı (Haziran 2011 - Nisan 2015)
Kamu Gözetimi Muhasebe ve Denetim Standartları Kurumu - Uzman Yardımcısı (Nisan 2015 - Şubat 2019)
Kamu Gözetimi Muhasebe ve Denetim Standartları Kurumu - Uzman (Şubat 2019 - Devam ediyor)