VERİ TABANINDA ADAPTİF YETKİ YÖNETİMİ VE UYGULAMASI

Veli HAKKOYMAZ, Ömer MOLLARECEP
[1]Yıldız Teknik Üniversitesi, Mühendislik Fakültesi, Bilgisayar Mühendisliği Bölümü, İstanbul
vhakkoymaz@gmail.com, omer.mollarecep@hotmail.com

ÖZET

Erişim yetkilerinin gözden geçirilmesi, bilgi güvenliğinde önemli ve zahmetli olup, kurumlar için zaman ve iş gücü maliyeti yüksek çalışmalardır. Özellikle, karmaşık veri tabanı yapıları içerisinde erişim yetkilerinin gözden geçirilmesi, her zaman maliyet etkin bir şekilde yapılamadığından ihmal edilebilmekte veya yeterince nitelikli biçimde gerçekleştirilememektedir. Bu çalışmada, erişim yetkilerinin adaptif bir biçimde sistem tarafından otomatik olarak belirlenmesi amaçlanmıştır. Elde edilen sonuçlardan, aktarım (transaction) tabanlı yetki kullanım bilgilerinin tutulması yöntemi ile erişim yetkilerinin gözden geçirilmesi sürecinin daha nitelikli ve maliyet etkin şekilde gerçekleştirileceği göstermiştir. Bu makale, ISDFS 2015'de sunulmuş olup seçilerek bu dergide yayımlanmıştır.

Anahtar Kelimeler: Veri tabanı, Erişim yetkisi, Yetki denetimi, Yetki yönetimi

ADAPTIVE AUTHORITY MANAGEMENT AND APPLICATON IN DATABASES

ABSTRACT

Access rights control is one of the painful activities within the information security topic. The activities about access rights control, have usually high cost in terms of time and effort. Because of this, access rights control require more time and effort, especially on complicated database systems, sometimes it may be either omitted or not done properly. In this paper, a new method was proposed in that access rights are adaptively and automaticly defined by the system. By keeping transaction based rights usage information it is anticipated and observed that access right control processes will be more cost effective and qualified.

Keywords: Database, Access authorization, Authority control, Authorization management

I. GİRİŞ (INTRODUCTION)

Günümüzde sistem güvenliği çok katmanlı bir güvenlik anlayışını yansıtmaktadır. Bu anlayış verinin iletildiği bütün katmanlarda, kullanıcı ara yüzünden, iş mantığı uygulamasına ve verinin saklanma ortamına kadar, güvenlik gereksinimlerinin yerine getirilmesini gerektirmektedir [7,8]. Dolayısı ile verilerin saklandığı yer olan veritabanı içerisinde verilerin güvenli olarak saklanması hassas bir konudur.

Bir sistemin güvenliğinin sağlanmasında ana kontrolleri şu şekilde sıralayabiliriz [1,2] ;

- Erişim kontrolü (access control): Veriye yetkisiz erişimin engellenmesini sağlayan kontrol.
- Şifreleme (encryption): Hassas veya kritik verinin şifre kullanılarak değiştirilmesini, saklanmasını ve şifreli olarak iletilmesini sağlayan kontrol.
- Çıkarım kontrolü (inference control): Veri tabanına erişim hakkı olan ancak hassas bilgiye erişim yetkisi olmayan kullanıcının hassas veriyi veya bilgiyi dolaylı yollardan elde etmesinin engellenmesi kontrolüdür [5].
- Akış kontrolü (flow control): Veri veya bilginin yetkisiz kişilere ulaşma kanallarının kapatılması kontrolüdür.

Ülkemizde özellikle bankacılık ve telekomünikasyon sektöründe bilginin güvenliğinin sağlanması, yasalar ile teminat altına alınmıştır. Bu kapsamda söz konusu sektörlerde, gerek devlet kuruluşları (BDDK, SPK, TİB, BTK, vb.) tarafından gerekse, özel sektörden bağımsız denetçi kuruluşlar tarafından denetimler gerçekleştirilmektedir. Güvenlik konusunda yapılan denetimlerde kullanıcıların erişim yetkilerinin düzenli olarak gözden geçirilmesi önemli bir başlıktır [3,6].

Bu çalışmada temel hedef, çok kullanıcılı bir sistemde, organizasyon şeması içerisinde var olan rollere uygun olarak yönetimce standart olarak atanmış erişim yetkilerinin kimi kullanıcılarca kullanılmadığı ortamlardır. Önerilen yöntem ile kullanıcı yetkilerinin adaptif bir şekilde sistem tarafından otomatik olarak olması gereken seviyeye taşınması hedeflenmektedir. Aynı organizasyon içerisinde çok sayıda ve değişik kullanıcıların kullandığı büyük birleşik veri tabanlarının yönetiminde, bu yöntem çok daha önemli olacaktır. Kurumsal veritabanı yönetim sistemlerinde çok sayıda kullanıcının ve kullanıcı rollerinin bulunması gibi nedenler ile yeni bir kullanıcı tanımlandığında, her zaman en az yetki prensibine (principle of least privilege) uygun şekilde tanımlamalar yapılamayabilmektedir. Yapılan çalışmada, bahsedilen kontroller ve denetleyici, düzenleyici kuruluşların denetimlerine uyumlu olma amacı ile, verilen yetkilerin kullanılıp kullanılmadığını tespit ederek erişim yetkilerinin adaptif bir şekilde olması gereken seviyeye indirilmesi amaçlanmıştır [11,12].

Bu makale 6 bölümden oluşmaktadır: Bu ilk bölümün ardından ikinci bölümde, veritabanı erişim kontrolü ile ilgili var olan yaklaşımlardan bahsedilmiştir. Üçüncü bölümde yetki yönetimi tartışılmış ve önerilen yöntem tanıtılmıştır. Dördüncü bölüm, böyle bir çalışmanın gerekliliği ile ilgili tartışmaya ayrılmıştır. Beşinci bölümde, geliştirilen yetki yönetim simülasyon uygulamasından kısaca bahsedilirken, sonuç ve değerlendirme ise altıncı bölümde verilmiştir.

II. VERİ TABANI ERİŞİM KONTROLÜ (DATABASE ACCESS CONTROL)

Veri tabanı güvenliğinin sağlanması aşağıdaki başlıklar altında incelenebilir:

- Veri erişimi ile ilgili etik konular: Konu, gizliliğin yanı sıra mahremiyetin sağlanması yönüyle .
- Yasal zorunluluklar: Yasa koyucu veya kurum tarafından uygulanan, kanun, politika, prosedür gibi kurallar yönünden.
- Sistem ile ilgili güvenlik kontrolleri: Örnek olarak donanım, işletim sistemi veya veri tabanı yönetim sistemi seviyesinde belirlenecek güvenlik kontrolleri yönüyle,
- Veri güvenlik seviyeleri: Organizasyon tarafından tanımlanması gereken veri gizlilik seviyeleri yönüyle. Örnek olarak tasnif dışı, gizli, çok gizli, kozmik gibi.

Bilgi güvenliği alanında erişim kontrolü, kullanıcıların kaynaklar ile etkileşimini yönetir. Erişim kontrolü kapsamında kullanıcıların belirli kaynaklara erişmesi açısından, söz konusu kullanıcıların tanımlandığı ve kimlik doğrulamasının yapıldığı var sayılır.

Erişim kontrol modellerinden Discretionary Access Control (DAC) (*İhtiyari Erişim Kontrolü*), Mandatory Access Control (MAC) (*Zorunlu Erişim Kontrolü*) ve Role Based Access Control (RBAC) (*Görev Bazlı Erişim Kontrolü*) genel kabul görmüş olan üç erişim kontrol modelidir. Bunlardan:

- DAC: Veriye erişimin veri sahibi tarafından kullanıcılara belirlenen yetkiler ve politikalar çerçevesinde gerçekleştiği,
- MAC: hem veri hem de kullanıcıların güvenlik seviyelerine göre sınıflandırıldığı ve verilere erişimin buna uygun olarak kurumsal güvenlik politikaları çerçevesinde gerçekleştiği,
- RBAC: veriye erişim yetkilerinin organizasyondaki tanımlı olan roller çerçevesinde gerçekleştiği

modellerdir [1, 9].

Bu kontrollerden erişim yetkilendirmesi başlığı altında, sistem içerisinde yer alan verilerin gizlilik sınıflarına göre;

- Hangi veriye (yani Ne sorusu)
- Hangi kullanıcılar (yani Kim sorusu)
- Hangi yetkiler ile (yani Nasıl sorusu)
- Süre uzunluğu ile (yani Hangi Süre sorusu)

erişeceği konuları yer alır. Bu konular özellikle büyük miktarda ve farklı kaynaklardan gelen veriyi işleyen organizasyonların veri tabanı yönetim sistemlerinde daha karmaşık bir hal almaktadır. Söz konusu kurumların veri tabanı yönetim sistemleri incelendiğinde her bir veri tabanı nesnesine erişim yetkisinin hangi seviyede olması gerektiğinin kontrolünün tek tek gözden geçirilmesi çok zor olmaktadır. Ayrıca verilen yetkilerin tamamının kullanılmadığı da görülmektedir.

Buna ilaveten bilgi güvenliği yönetim sisteminde uluslararası bir standart olan ISO IEC 27001 standardında erişim yetkilerinin düzenli olarak gözden geçirilmesine ilişkin kontroller yer almaktadır [6] .

III. YETKİ YÖNETİMİ (AUTHORITY MANAGEMENT)

Güvenlik bakış açısı ile veritabanında yer alan verilere erişebilmesi için kullanıcılara veya uygulamalara işini yapabilecek en az yetki prensibine (*principle of least privilege*) uygun olarak erişim tanımlanmalıdır. Bu prensibe göre kullanıcılar veya uygulamalar işlerini yapabilecekleri en düşük yetki seviyelerinde çalışmaları gerekmektedir. Bir veri tabanı yönetim sistemi içerisinde hangi verilerin şifreleneceği, hangi kullanıcıların hangi verilere hangi yetkiler ile ne sürede erişeceği konusu en mükemmel şekilde tasarlansa bile, muhtemelen yaşayan sistem içerisinde yeni oluşturulacak verilerin ve kullanıcıların bahsedilen konulardaki seviyeleri iş yaşantısının yoğun temposu içerisinde her zaman yeterince doğru tespit edilememektedir.

Somut olarak güvenlik denetimlerinde ve çalışmalarında kullanıcıların erişim izinlerinin gözden geçirilmesi önemli ve zahmetli bir konudur. Birçok ticari veri tabanı yönetim sistemi içerisinde kullanıcıların erişimlerinin yukarıda bahsedilen kapsamda gözden geçirilmesini sağlayacak bir modül bulunmamaktadır. Bu makalede önerilen yöntem ile erişim yetkilerinin denetlenmesi ve kullanıcıların doğru yetki seviyelerine atanması süreci daha zahmetsiz olarak gerçekleştirilecektir. Bu yöntem bir alt alt başlıkta tanımlanmıştır.

A. Önerilen Yöntem

Veritabanı yönetim sistemi içerisinde hareketlere (transactions) ilişkin birçok detaylı kayıtlar yer almaktadır. Söz konusu detaylı kayıtlar veritabanlarında özellikle felaket kurtarma işlemlerinde hayati öneme sahiptir. Bir kullanıcıya bir nesne için herhangi bir yetki verilmesi konusu da bir harekettir. Önermiş olduğumuz yöntemin uygulamasında;

- Her bir nesne için, kullanıcıların yetkilerinin yer aldığı Yetki Matrisi (Capability Matrix-CM, Bakınız Tablo I) veya her bir kullanıcı için nesnelere olan yetki matrisi yer almaktadır.

TABLO 1: YETKİ MATRİSİ

Capability Matrix for Object O					
Users/ Yetki	***Select***	***Insert***	***Delete***	***Update***	**…**
User 1	*P1				
User 2	P2	P3			
User 3	-	P4	-	-	
User 4	P5		P6	P7	
…					

* P ilgili kullanıcının nesne üzerindeki yetkisini ifade eder

- Bunun yanı sıra yetki kullanım zamanı ve kullanılan yetkinin ne olduğunu gösterir ayrı bir matris Kullanım Matrisi (Usage Matrix-UM, bakınız Tablo II) yer almaktadır.

TABLO II: KULLANIM MATRİSİ

Usage Matrix for Object O						
User - Yetki	***P1***	***P2***	***P3***	***P4***	***P5***	**…**
User 1	X*		X	X		
User 2		X		X	X	
User 3	X		X	X		
User 4	X	X		X		
…						

* X ilgili kullanıcının P1 yetkisini kullanım sayısını ifade eder

Bir güvenlik uygulaması ortamında yukarıda bahsedilen CM ile UM matrislerinin yardımı ile basitten karmaşık olana doğru aşağıdaki aksiyonlar gerçekleştirilebilir:

- Yetki geri alma: CM ve UM matrislerinin karşılaştırması yapılarak kullanılmayan yetkilerin kullanıcılardan geri alınması (revoke işlemi) yapılır. Bunun için CM matrisinde güncelleme yapılması yeterlidir.
- İstatistiksel değerlendirme: Her bir kullanıcı için nesne üzerinde yetkilerin kullanım sıklıklarının ve profillerinin ortaya çıkarmak mümkündür. Bunun için UM Matrisi üzerinde okuma ve değerlendirme yapmak yeterli olacaktır.
- Zaman boyutlu istatistik: UM matrisi benzeri bir matris kullanarak her kullanım için bu matrise zaman damgası (time-stamp) konarak zaman boyutunda kullanım istatistiği çıkarılır.

Söz konusu kullanıcıya ilişkin verilen yetki çerçevesinde hangi nesne ne zaman eriştiği bilgisinin tutulması ile söz konusu kullanıcının nesnelere erişim yetkisinin dönemsel olarak izlenmesi mümkün olacaktır. Böylece eğer bir kullanıcıya verilen yetkiler içerisinde kullanmadığı veya gerçekten ihtiyaç duymadığı yetkiler varsa, bu durumda uyarlamalı olarak (adaptif) yetki kısıtlaması yapılacaktır.

B. Uygulama Senaryosu

Bu durumun bizlere sağlayacağı kolaylık gerçek bir uygulama senaryosu ile daha kolay anlaşılabilir. Kullanıcılara ve uygulamalara veri tabanı içerisinde herhangi bir erişim kontrolü uygulanmadığını düşünelim. Böyle bir durumda "en az yetki prensibi" gereği verilerin ve kullanıcıların iş ihtiyaçları analiz edilmelidir. Bu durum yukarıda da bahsedildiği üzere zahmetli bir süreçtir. Kullanıcı erişimlerinin belirli bir dönem kayıt altına alındığı sistemde ise kullanıcıların (uygulama kullanıcıları da dahil olmak üzere) iş ihtiyaçları gereği günlük erişimleri rahatlıkla gözlemlenebilecektir. Kullanıcının sadece belirli dönemlerde (örnek olarak muhasebe kapanış, denetim dönemi gibi) ihtiyacı olacak olan erişimler de bu sırada kolaylıkla ortaya çıkartılabilecektir. Eğer söz konusu erişim yetkisinde "en az yetki prensibine" uygun olmayan yetkiler söz konusu ise, bu yetkilerin aşamalı (gradual) ve uyarlamalı (adaptif) olarak geri alınması gerekmektedir. Önerilen yöntemler yardımı ile tüm kullanıcıların ideal yetki seviyesine ulaşması sağlanmış olacaktır.

IV. TARTIŞMA (DISCUSSION)

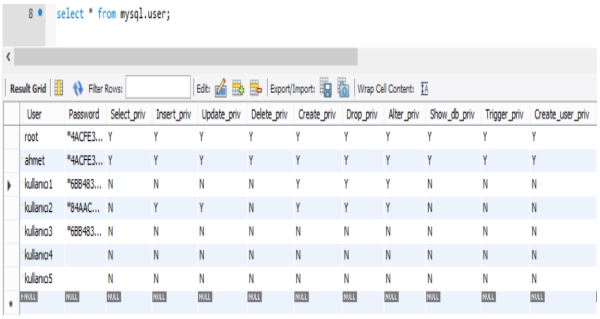
Veritabanı güvenliği konusunda TÜBİTAK tarafından yayınlanmış olan “Oracle Veri Tabanı Güvenliği Kılavuzu” bulunmaktadır [10]. Söz konusu doküman içerisinde kullanıcı oluşturulması süreci hakkında “Yeni kullanıcı oluşturma ve hali hazırda kullanımda olan kullanıcı hesaplarının yetki değişimleri vs. gibi durumlarda, veritabanı yöneticisi ile birlikte güvenlik birim sorumlularına anlık bilgilendirme yapabilecek sistemler kurulmalıdır. Sınırlı haklara sahip bir kullanıcının haklarını genişletmesi (zararlı kod çalıştırma gibi. teknikler ile) durumunda anlık sorgulama ve araştırma yapılmasına imkân tanıyan alarm mekanizmaları büyük öneme haizdir.”

denilmektedir. Yine aynı kılavuz içerisinde veri tabanı yönetim sisteminin kurulumu ile birlikte sistemde tanımlı olarak gelen varsayılan kullanıcılar ile ilgili "Veritabanı kullanıcıları incelenmeli ve veritabanının kendi oluşturduğu veya sonradan oluşturulan ama kullanılmayan kullanıcı hesapları (dormant accounts) belirlenmelidir. Kullanılmayan veritabanı kullanıcı hesapları kilitlenmiş olmalıdır. Diğer bir değişle bu hesapların account_status değeri locked olmalıdır. Aksi halde saldırganların bu kullanıcıları kullanarak sisteme sızma olasılıkları artacaktır." denilmektedir. Çalışmamızın söz konusu durum hakkında, sistemde account_status değeri locked olmayan kullanıcıların kolaylıkla tespit edilebilmesine de katkısı olacaktır. Söz konusu doküman içerisinde kullanılmayan kullanıcı hesaplarının tespit edilmesi hakkında "Pasif kullanıcı hesaplarını belirleme esnasında dba_users tablosunda en son sisteme giriş tarihleri incelenebilir. Örneğin SQL*Plus'ta aşağıdaki sorgu kullanılarak, son 30 gün içerisinde kullanılmayan kullanıcı hesapları bulunabilir:" denilmekte ancak uyarlamalı (adaptif) bir yöntem önerilmemektedir. Önerdiğimiz yöntem yetkilerin uyarlamalı olarak gereken seviyeye taşınması sureti ile diğer yapılan çalışmadan farklılaşmaktadır.

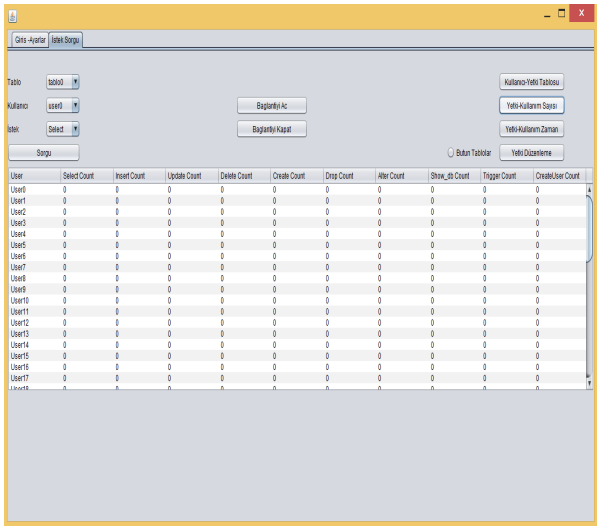
V. YETKİ YÖNETİM UYGULAMASI (AUTHORITY MANAGEMENT APPLICATION)

Veritabanı Yetki Yönetimi, simülasyon uygulaması olarak gerçekleştirilmektedir. Veritabanında en önemli güvenlik açıklarından biri yanlış kişilere yanlış yetki verilmesidir. Yanlış bir yetki verilmesinde veritabanında ciddi güvenlik açıkları ortaya çıkabilir. Yanlış yetki dağıtımının önüne geçebilmek için düzenli bir şekilde periyodik olarak (örneğin aylık periyotlarla) yetki kontrolü yapılması gereklidir. Bu uygulamada kullanıcıların yetkilerini kontrol edebilmek, yanlış verilen yetkiler sonucu oluşabilecek güvenlik açıklarının önüne geçebilmek ve ideal yetki dağıtımının yapılabilmesini sağlamak amaçlanmıştır. Kullanıcıların dönemsel olarak kullandıkları yetkiler ve kullanım sıklıkları incelenerek analiz edilmiştir. Sonuçta, kullanıcıların kullanmadığı yetkileri elinden alınıp, ideal yetki dağıtımının yapılması sağlanarak veritabanında bu konuda oluşabilecek güvenlik açıklarının önüne geçilmiştir. Örnek olarak, Mysql veritabanı kullanıcı yetki kullanım tablosu (Bkz. Şekil 1) incelenmiş yetkilerin nasıl saklandığı görülmüştür.

Geliştirilen uygulamaya, giriş yapıldıktan sonra simülasyonda kullanılmak istenen kişi sayısı ve tablo sayısı yazılıp oluşturacaktır. Tabloları oluşturduktan sonra istediği tablodan istediği kullanıcının yetkilerini görebilecek ve o yetkiyi o kişinin kaç defa kullandığını görebilecektir. Dönemsel olarak kayıt altına alındıktan sonra matrisler analiz edilecektir. Tabloları oluşturduktan sonra kullanıcıların o nesneye ait yetkileri, kullanım sayıları veya kullanım zamanları ekrana getirilebilir (Bkz. Şekil 2).



Şekil 1. Kullanıcı Yetki Tablosu (MySQL)



Şekil 2. Yetki Kullanım Uygulama Ekran Görüntüsü

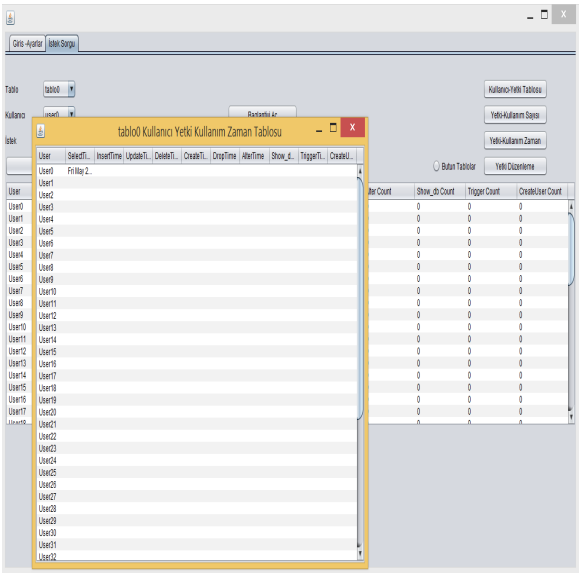
TabloOlustur fonksiyonu, programdan tablo sayısını ve kullanıcı sayısını alacak ve her bir tablo için kullanıcı yetkilerini (kullanıcı rolüne göre) rastgele olarak oluşturacaktır. TabloGosterYetki fonksiyonu, seçilen tabloya göre kullanıcı yetki matrisini ekrana getirecektir. TabloGosterYetkiKullanım fonksiyonu, seçilen tabloya göre kullanıcı yetki kullanım matrisini ekrana getirecektir. TabloGosterYetkiKullanımZaman fonksiyonu, seçilen tabloya göre kullanıcı yetki kullanım zaman matrisini ekrana getirecektir (Bkz. Şekil 3).

YetkiDüzenle fonksiyonu, kullanıcıların yetkileri kullanım sayılarına bakarak kullanıcıların kullanılmayan yetkilerini elinden alma işlemidir.

VI. SONUÇ (CONCLUSION)

Bu çalışmada (1) veri tabanı güvenlik kontrolleri ve bu kontrollerden özellikle erişim yetkileri, (2) erişim yetkilerinin yönetimi (gözden geçirilmesi), (3) erişim izinlerinin gözden geçirilmesi sürecinin zahmetli bir süreç olduğu, (4) bu durumun kurumlar açısından zorunluluğuna dikkat çekilmiştir. Gelecekte erişim izinlerinin ya da yetkilerinin gözden geçirilmesine ilişkin yapılacak iyileştirme çalışmasının adaptif

olarak sistem tarafından otomatik olarak gerçekleşmesi için bir çerçeve belirlenmiştir.



Şekil 3. Yetki Kullanım Zaman Matrisi Görüntüleme

Bu çalışmada önerilen yaklaşım ile veritabanlarında güvenliğin daha da arttırılabileceği değerlendirilmektedir.

VII. KAYNAKLAR

[1] R. Elmasri, S. B. Navathe, Database Systems: Models, Languages, Design and Application Programming, 6th Edition, Pearson, 2011

[2] Teknik Rapor: Felaketten Kurtarma Ve Depolama Çalışma Grubu Raporu, Türkiye Bilişim Derneği, 2009.

[3] Bağımsız Denetim Kuruluşlarınca Gerçekleştirilecek Banka Bilgi Sistemleri ve Bankacılık Süreçlerinin Denetimi Hakkında Yönetmelik, BDDK, 2010

[4] COBIT (Control Objectives for Information and Related Technology) Framework: User Account Management, ISACA, 2012

[5] C. Farkas, S. Jajodia, The Inference Problem: A Survey, ACM SIGKDD Explorations., v.4(2), 2002

[6] ISO-27001 Bilgi Güvenliği Yönetim Sistemi Standardı, 2013

[7] J.B.D. Joshi, W.G. Aref, A. Ghafoor, and E.H. Spafford, Security Models for Web-based Applications, Comm. of ACM, v.44(2), Feb 2001

[8] Y. Yang, Y. Li, R.H. Deng, F. Bao, Shifting Inference Control to User Side: Architecture and Protocol, IEEE Trans. on Dependable and Secure Computing, v.7, n.2, 2010

[9] R.S. Sandhu, P. Samarati. Access Control: Principle and Practice. Communications Magazine, IEEE, 32(9):40–48, September 1994.

[10] T. Türköz, A. Sezer, Y. Çankaya, E. Çalışkan, Oracle Veritabanı Güvenliği Kılavuzu, Sürüm 2, Doküman Kodu: BGT-500, Kasım 2012

[11] V. Hakkoymaz, İ. Alan, An Adaptive Security Policy Design and Management for Distributed Systems, Proc. of ISCTurkey: Information Security and Cryptology Conference, pp.238-243, December 2007.

[12] V. Hakkoymaz, İ. Alan, Design of an Adaptive Security Manager for Distributed Systems, International Journal of Web Applications, v.2, n.2, pp.137-148, June 2010.