

IIS Kurulu Uygulama Sunumcu ile Veritabanı Arasındaki Bağlantının Güvenlik Esasları

Ünal Perendi
TÜBİTAK- Ulusal Elektronik ve Kriptoloji Enstitüsü
Gebze, KOCAELİ

Özet

Bu dokümanda IIS Kurulu olan bir web veya uygulama sunumcusundan veritabanına bağlanırken gerekli güvenlik gereksinimlerinden bağlantı güvenliği esaslarıyla ilgili gereksinimler anlatılmıştır.

Giriş

Web programlama dillerinin en önemli özelliklerinden biri veritabanlarıyla birlikte çalışabilmesidir. Web tabanlı uygulamalar geliştirilirken saklanacak ve erişilecek olan verilerin güvenliği konusuna dikkat edilmelidir. Bunu sağlamak için gerekli güvenlik işlemlerinden yetkilendirme seçimi ve bağlantı dizilerinin güvenli saklanması konuları hakkında bilgi verilmektedir.

1. Yetkilendirme Seçimi

OS (Windows, Unix) Yetkilendirmesi daha güvenli bir yetkilendirme sağlamasına rağmen güvenlik duvarları ve güvenilmeyen etki alanlarından dolayı veritabanının kendi yetkilendirmesine başvurulmaktadır. Eğer veritabanı yetkilendirmesi yapılacaksa güvenlik gereksinimlerine daha çok dikkat edilmelidir.

Eğer kullanılan uygulama veritabanı yetkilendirmesi yapıyorsa bağlantı dizilerindeki kullanıcı adı ve şifreleri açık halde saklanması gerekebilir.

Veritabanları ile uygulama sunumcuları arasında her açıdan güvenliği artırıcı yöntem olarak sertifikalar ile SSL uygulaması ve istemci bilgisayarı (Web sunucu veya uygulama sunucusu) ile veritabanı sunucu arası şifreleme için IPSEC kullanılması şiddetle tavsiye edilmektedir.

1.1 Windows Yetkilendirmesi

Windows yetkilendirmesi aşağıdaki sebeplerden dolayı veritabanı yetkilendirmesinden daha güvenlidir:

- Kimlik bilgileri ağ üzerinden transfer edilmez.
- Kullanıcı adlarının ve şifrelerin bağlantı dizilerinin içinde bulunmaması
- Şifre süresinin dolması periyodu, şifre uzunluğu ve hesap kilitlenmesi gibi ek özelliklerin bulunması. Sözlük saldırılarına daha dayanıklı olması.

Örnek: Windows Yetkilendirmesi ile SQL Sunumcu bağlantısı

- İstemci uygulaması içerisinde "*Trusted Connection=Yes*", veya "*Integrated Security=SSPI*" olan bağlantı dizinleri kullanır.

```
"server=MySQL; Integrated Security=SSPI; database=Istanbul"
```

1.2 SQL Yetkilendirmesi

Bazı uygulamalar Windows yetkilendirmesi kullanılmasını engellemektedir:

- Veritabanı istemcisi ile veritabanı sunucusu bir güvenlik duvarı ile ayrılmışa,
- Eğer uygulama bir veya birden çok veritabanına çoklu kimlikle bağlanması gerekiyorsa,
- SQL sunucu dışında bir veritabanına bağlanılıyorsa Windows Yetkilendirme kullanılmaz.

Bu durumlarda veritabanının kendi yetkilendirme mekanizması kullanılmaktadır ve aşağıdaki güvenlik önlemleri alınır:

- En düşük yetkili hesap ile veritabanına bağlanılır.
- Transfer edilen kimlik bilgileri güvenlik altına alınır.
- Kimlik bilgilerini içeren veritabanı bağlantı dizileri güvenlik altına alınır.

Örnek Bağlantı Dizileri:

Eğer SQL veritabanı kullanılıyorsa;

```
SqlConnectionString = "Server=YourServer;  
Database=YourDatabase;  
uid=YourUserName;pwd=YourStrongPassword;"
```

Eğer Oracle veritabanı kullanılıyorsa;

```
SqlConnectionString = "Provider=MSDAORA;Data  
Source=YourDatabaseAlias;  
User ID=YourUserName;Password=YourPassword;"
```

2. Güvenli İletişim

Veritabanının kendi yetkilendirme mekanizması kullanıldığında bağlantı dizilerinin açık halde gönderilmesini engellemek için SSL(veritabanı sunumcusu üzerinde) ve IPSEC mekanizmaları kullanılmalıdır.

Bunun yanında veritabanı bağlantı dizilerinin güvenli saklanması ile ilgili farklı yaklaşımlar mevcuttur:

- DPAPI ile şifrelemek
- Web.config veya Machine.config dosyası içerisinde açık olarak tutmak,
- UDL dosyaları kullanmak,
- Kütük kaydı
- COM+ kataloğu

2.1 DPAPI Kullanımı

Windows 2000 ve sonrası işletim sistemlerinde veri şifreleme ve çözme işlemleri için Win32® Data Protection API (DPAPI) kullanılmaktadır. DPAPI, Cryptography API (Crypto API)'nin bir parçası olup Crypt32.dll içerisinde geliştirilmiştir. CryptProtectData ve CryptUnprotectData. Olarak iki metottan oluşmaktadır.

DPAPI tam bir güvenlik sağlamasa da kriptoloji kullanan uygulamalardaki anahtar değişim sorununu çözmektedir. Şifreleme ile verinin güvenliğini sağladıktan sonra anahtarın güvenliği içinde DPAPI kullanılabilir. DPAPI fonksiyonlarını çağıran kod şifre ve kullanıcı adı ile ilişkilendirilerek DPAPI sayesinde uygulamanın değil işletim sisteminin anahtarı yönetmesi sağlanır.

2.2 Web.config ve Machine.config Kullanımı

Açık olarak saklanan şifrelerin web.config de saklanması tavsiye edilmemektedir.

HttpForbiddenHandler dosyaların indirilmesi ve görüntülenmesini engellemektedir. Buna rağmen ilgili dizinlere erişim hakkı olan herkes konfigürasyon dosyaları içindeki kullanıcı adı ve şifreleri görebilir.

Machine.config Web.config dosyasına göre daha güvenli bir saklama yeri olarak kabul edilmektedir. Çünkü Web uygulamasının sanal dizininin dışında sistem dizininin altında erişim kontrol listeleri tarafından varsayılan olarak korunmaktadır.

2.3 UDL Dosyalarının Kullanımı

The OLE DB .NET Data Provider UDL dosyalarının bağlantı dizilerinde kullanılmasını desteklemektedir

UDL dosyalarının diğer uygulama dosyaları ile beraber sanal bir dizinde tutulması tavsiye edilememektedir. UDL dosyaları işletim sisteminin bulunduğu sabit disk bölmesinden farklı bir bölmede, Web uygulamasının sanal izin hiyerarşisinin dışında, dosyanın ve dosyanın bulunduğu dizinin güvenliği sağlanarak saklanmalıdır.

2.4 UDL dosyaları için (ACL) erişim kontrol listeleri

UDL dosyalarına ait erişim kontrol listeleri Machine.config dosyasının varsayılan erişim kontrol listelerinden dahi daha katı olarak ayarlanabilir. Örneğin Machine.config dosyasına ait varsayılan erişim kontrol listesi aşağıdaki gibidir:

```
MachineName\ASPNET:R  
BUILTIN\Users:R  
BUILTIN\Power Users:C  
BUILTIN\Administrators:F  
NT AUTHORITY\SYSTEM:F
```

UDL dosyalarında sadece *Administrators*, the “*System*” account, and the *ASP.NET process* account (okuma hakkını sağlayan) hakları bulunması yeterlidir.

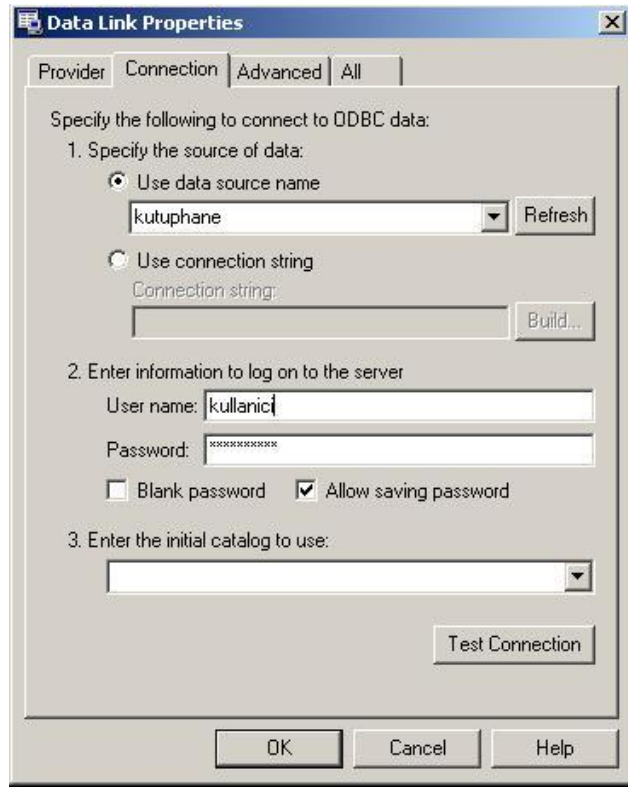
```
BUILTIN\Administrators:F  
MachineName\ASPNET:R  
NT AUTHORITY\SYSTEM:F
```

UDL dosyalarının her türlü ADO.NET istemci uygulaması tarafından değiştirilebilir olmaları her bağlantı açıldığında UDL dosyasına referans eden bağlantı dizileri metin dosyalarının işlenmesini gerektirir. Eğer performans ihtiyacı önemliyse UDL dosyalarının kullanımı yerine statik bağlantı dizileri önerilmektedir.

Yeni bir UDL dosyası yaratmak için:

1. Erişim kontrol listeleri uygun dizin içerisinde yeni bir metin dosyası oluşturulur. Dosya uzantısı “*.udl*” olarak değiştirilir.

2. Çift tıklanarak açıldığında ilgili parametreler doldurulur.

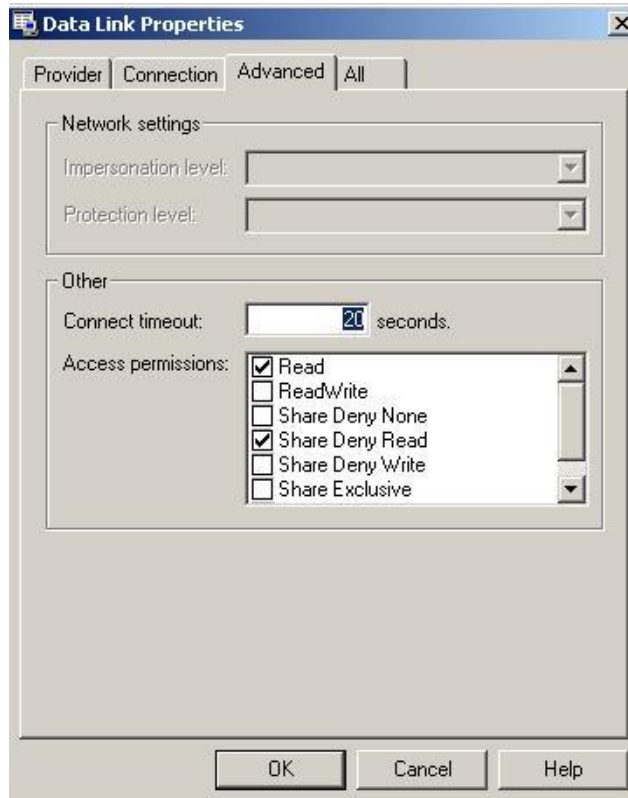


The image shows the 'Data Link Properties' dialog box with the 'Connection' tab selected. The dialog is titled 'Data Link Properties' and has four tabs: 'Provider', 'Connection', 'Advanced', and 'All'. The 'Connection' tab is active. The main area contains the following fields and options:

- Specify the following to connect to ODBC data:
- 1. Specify the source of data:
 - Use data source name: A dropdown menu shows 'kutuphane' and a 'Refresh' button is next to it.
 - Use connection string: A 'Connection string:' label is above an empty text box, with a 'Build...' button to its right.
- 2. Enter information to log on to the server:
 - User name: A text box containing 'kullanici'.
 - Password: A text box with masked characters '*****'.
 - Blank password Allow saving password
- 3. Enter the initial catalog to use: An empty dropdown menu.
- A 'Test Connection' button is located at the bottom right of the main area.

At the bottom of the dialog are three buttons: 'OK', 'Cancel', and 'Help'.

3. İlgili haklar atanır, bağlantı kopma süresi belirlenir.



The image shows the 'Data Link Properties' dialog box with the 'Advanced' tab selected. The dialog is titled 'Data Link Properties' and has four tabs: 'Provider', 'Connection', 'Advanced', and 'All'. The 'Advanced' tab is active. The main area contains the following fields and options:

- Network settings:
 - Impersonation level: A dropdown menu.
 - Protection level: A dropdown menu.
- Other:
 - Connect timeout: A text box containing '20' followed by 'seconds'.
 - Access permissions: A list box with the following items:
 - Read
 - ReadWrite
 - Share Deny None
 - Share Deny Read
 - Share Deny Write
 - Share Exclusive

At the bottom of the dialog are three buttons: 'OK', 'Cancel', and 'Help'.

Kod içerisinde ise *.udl* dosyaları aşağıdaki gibi kullanılır.

```
Dim objConn
Dim connStr
Set objConn = Server.CreateObject("ADODB.Connection")
connStr = "File Name=K:\da\conn.udl; "
objConn.Open(connStr)

Set objConn = Server.CreateObject("ADODB.Connection")
connStr = "File Name=K:\da\conn.udl; "
objConn.Open(connStr)
```

2.5 Kütük Değeri Kullanımı

Windows kullanıldığında bağlantı dizisi *HKEY_LOCAL_MACHINE (HKLM)* or *HKEY_CURRENT_USER (HKCU)* altında bir kütük değeri olarak saklanabilir.

Regedt32.exe kullanılarak bu kütük değerinin erişim kontrol listeleri minimum haklar kuralına göre ayarlanmalıdır.

2.6 COM+ Kataloğu Kullanımı

Eğer Web uygulamanız servis bileşenleri içeriyorsa bağlantı dizileri COM+ katalogunda saklanabilir ve Component Services Tool aracılığı ile yönetilebilirler.

COM+ katalogu yüksek seviye bir güvenlik sağlamaz. Çünkü bilgi şifrelenmemiştir. Fakat ekstra işlem basamağı yarattığından yapılandırma dosyalarındaki bağlantı dizilerine göre daha güvenlidir.

Component Services Tool aracılığıyla da Kataloga erişimi sınırlandırmak için System uygulaması üzerindeki *Administrator* and *Reader* rollerine sahip kullanıcılar tanımlanır.

Aşağıda servis bileşeninden bir nesne yapıcısının nasıl elde edildiğini gösteren küçük bir örnek mevcuttur:

```
[ConstructionEnabled(Default="Default Connection String")]
public class YourClass : ServicedComponent
{
    private string _ConnectionString;
    override protected void Construct(string s)
    {
        _ConnectionString = s;
    }
}
```

Sonuç

Uygulama sunumcuları ile kullandıkları veritabanları arasındaki bağlantının güvenliği için sistem gereksinimlerine uygun yetkilendirme tercihi (OS yetkilendirmesi veya Veritabanı yetkilendirmesi) yapıldıktan sonra kullanıcı adı ve şifrenin transferi sırasındaki güvenlik için SSL veya IPSec,, saklanması sırasındaki güvenliği için ise kodun içersinde bulunmayacak şekilde bir kullanım ile birlikte UDL dosyasının içersinde, kütükte, Com+ kataloğunda, machine.config dosyasının içersinde kullanılması ve her biri için kendisine ait gerekli ek güvenlik önlemlerinin alınması gerekmektedir.