

1.1.1 Şifrelenmemiş Web.Config Dosyası (Unencrypted Web Config File) (CWE-312)

Açıklık Önem Derecesi: Düşük

Açıklığın Etkisi: Bilgi İfşası

Açıklığın Barındıran Dosyalar/Satırlar:

Proje Dosyası/Dosya Adı	Satır Numarası

Açıklığın Açıklaması:

ASP.NET web uygulamalarda web.config dosyaları hassas veriler içerebilmektedir. Örneğin servis hesap bilgileri veya bağlantı cümlecikleri gibi. Bu hassas veriler güvenli bir şekilde depolanmalıdır. Aksi takdirde yerel dosya sistemine sızan bir saldırgan bu bilgileri bulabilir ve saldırı derinliğini genişletebilir. Bu güvensiz yapılandırma dosyasına örnek olarak şu verilebilir:

Güvensiz Web.Config:

```
<!-- web.config File with Plain-Text Sensitive Content -->
<configuration>
  <connectionStrings>
    <add name="ServiceName" connectionString="[connection strings]" />
  </connectionStrings>
  <system.web>
    <machineKey validationKey="[validation key]" decryptionKey="[decryption key]" />
  </system.web>
</configuration>
```

Güvenli Web.config:

```

<!-- web.config File with Encrypted Sensitive Content -->

<configuration>
  <connectionStrings>
    configProtectionProvider="RsaProtectedConfigurationProvider">
      <EncryptedData Type="http://www.w3.org/2001/04/xmlenc#Element"
        xmlns="http://www.w3.org/2001/04/xmlenc#">
        <EncryptionMethod Algorithm="[Encryption Algorithm]" />
        <KeyInfo xmlns="http://www.w3.org/2000/09/xmldsig#">
          <EncryptedKey xmlns="http://www.w3.org/2001/04/xmlenc#">
            <EncryptionMethod Algorithm="[Encryption Algorithm]" />
            <KeyInfo xmlns="http://www.w3.org/2000/09/xmldsig#">
              <KeyName>RSA Key
                </KeyName>
            </KeyInfo>
            <CipherData>[Cipher Value]</CipherValue>
          </CipherData>
        </EncryptedKey>
      </KeyInfo>
      <CipherData>[Cipher Value]</CipherValue>
    </CipherData>
  </EncryptedData>
</connectionStrings>
<system.web>
  <machineKey configProtectionProvider="RsaProtectedConfigurationProvider">
    <EncryptedData Type="http://www.w3.org/2001/04/xmlenc#Element"
      xmlns="http://www.w3.org/2001/04/xmlenc#">
      <EncryptionMethod Algorithm="[Encryption Algorithm]" />
      <KeyInfo xmlns="http://www.w3.org/2000/09/xmldsig#">
        <EncryptedKey xmlns="http://www.w3.org/2001/04/xmlenc#">
          <EncryptionMethod Algorithm="[Encryption Algorithm]" />
          <KeyInfo xmlns="http://www.w3.org/2000/09/xmldsig#">
            <KeyName>RSA Key
              </KeyName>
          </KeyInfo>
          <CipherData>
            <CipherValue>[Cipher Value]</CipherValue>
          </CipherData>
        </EncryptedKey>
      </KeyInfo>
      <CipherData>
        <CipherValue>[Cipher Value]</CipherValue>
      </CipherData>
    </EncryptedData>
  </machineKey>
</system.web>
</configuration>

```

Güvensiz web.config dosyasında iki hassas veri vardır. Birincisi bağlantı cümlecği, ikincisi IIS makina anahtarı (Viewstate'leri şifreler). Bunlar açık metin (plain-text) yerine şifreli bir şekilde web.config dosyasında tutulmalıdır. Dolayısıyla güvenli web.config dosyası örneğinde bu iki hassas yapılandırma bloğunun şifreli hali konulmuştur.

Kurum ASP.NET uygulamasında web.config yapılandırma dosyasında "Şifrelenmemiş Web.config Dosyası (CWE-312) açıklığı tespit edilmiştir.

::::BULGU::::

Açıklığın Önemi:

Bu açıklığı önlemek için web.config dosyalarındaki hassas veriler şifrelenmelidir. Bunun için aspnet_regiis.exe aracı kullanılabilir. Bu işlem uygulamalı olarak gösterilecektir:

Web.Config Connection Strings Bloklarını Şifreleme (Encryption) ve Çözme (Decryption)

i) Web.Config Veri Bağlantılarını Şifreleme (Encryption)

Web.Config yapılandırma dosyasındaki blokları (bu senaryo için connection strings bloklarını) şifrelemede asp.net framework'ünün bir aracı olan aspnet_regiis.exe 'den faydalanılabilir. Bu aracı sorunsuz kullanabilmek için CMD penceresinin yönetici olarak çalıştırılması gerekir.

Başlat:

CMD (Sağ Tık -> Yönetici Olarak Çalıştır)

Ardından aspnet_regiis.exe dosyasının yer aldığı dizin bulunur.

CMD (as Administrator):

```
C:\Windows\system32\> dir /s c:\aspnet_regiis.exe
```

Sıralanan aspnet_regiis.exe araç dizin yollarından .NET framework versiyonu en yüksek olan dizin yoluna gidilir. Örn;

CMD (as Administrator):

```
C:\Windows\system32\> cd "C:\Windows\Microsoft.NET\Framework64\4.0.30319"
```

Ardından aspnet_regiis.exe aracı verilecek -pef parametresi ve arguman değerleriyle çalıştırılır:

CMD (as Administrator):

// Şifreleme (Encryption)

```
C:\Windows\Microsoft.NET\Framework64\4.0.30319> aspnet_regiis.exe -pef "connectionStrings" "C:\inetpub\wwwroot"
```

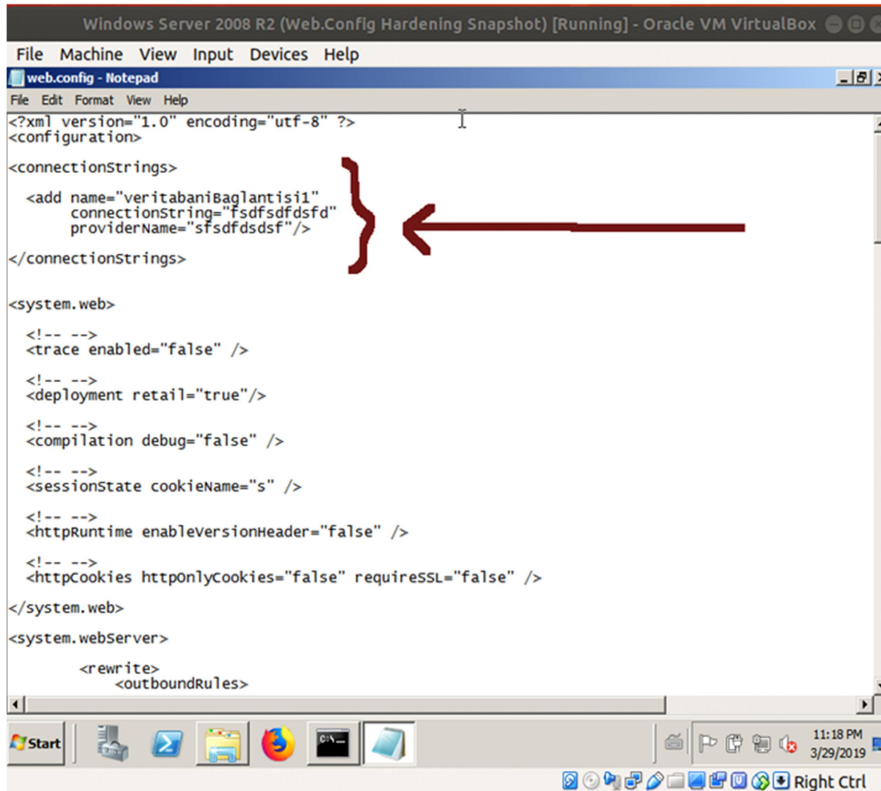
Çıktı:

Encrypting configuration section...

Succeeded!

Yukarıdaki kodlama ile C:\inetpub\wwwroot dizininde yer alan web.config dosyası içerisindeki <connectionStrings> bloğunun AES algoritması ile şifrenmesi sağlanır. Örneğin bir web.config dosyasında şifreleme öncesi bağlantı cümlecği paylaşılmıştır.

~ Şifreleme Öncesi ~



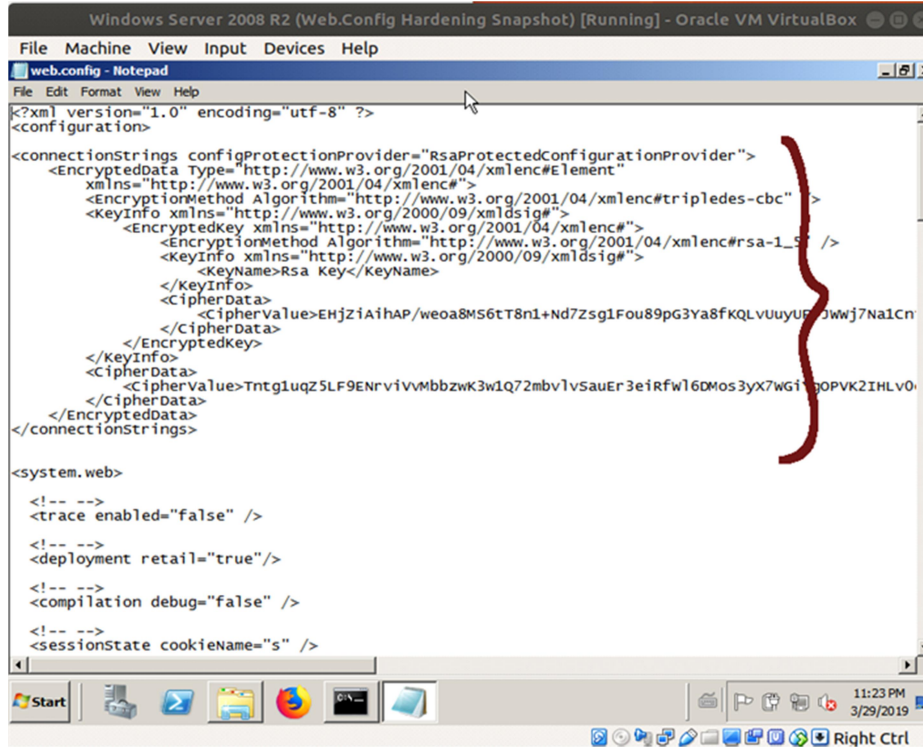
```
Windows Server 2008 R2 (Web.Config Hardening Snapshot) [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
web.config - Notepad
File Edit Format View Help
<?xml version="1.0" encoding="utf-8" ?>
<configuration>
<connectionStrings>
  <add name="veritabaniBaglantisil"
        connectionString="fsdfsdfsfd"
        providerName="sfsdfsdfs" />
</connectionStrings>
<system.web>
  <!-- -->
  <trace enabled="false" />
  <!-- -->
  <deployment retail="true"/>
  <!-- -->
  <compilation debug="false" />
  <!-- -->
  <sessionState cookieName="s" />
  <!-- -->
  <httpRuntime enableVersionHeader="false" />
  <!-- -->
  <httpCookies httpOnlyCookies="false" requireSSL="false" />
</system.web>
<system.webServer>
  <rewrite>
    <outboundRules>
```

Şekil 3. Yapılandırma Dosyasında Açık Tutulan Bir Bağlantı

String Bloğu Gösterimi

Ardından bu bağlantı cümlecığının şifrelenmiş hali gösterilmiştir:

~ Şifreleme Sonrası ~



```
<?xml version="1.0" encoding="utf-8" ?>
<configuration>
  <connectionStrings configProtectionProvider="RsaProtectedConfigurationProvider">
    <EncryptedData Type="http://www.w3.org/2001/04/xmlenc#Element"
      xmlns="http://www.w3.org/2001/04/xmlenc#"
      >
      <EncryptionMethod Algorithm="http://www.w3.org/2001/04/xmlenc#tripledes-cbc" />
      <KeyInfo xmlns="http://www.w3.org/2000/09/xmldsig#"
        >
        <EncryptedKey xmlns="http://www.w3.org/2001/04/xmlenc#"
          >
          <EncryptionMethod Algorithm="http://www.w3.org/2001/04/xmlenc#rsa-1_5" />
          <KeyInfo xmlns="http://www.w3.org/2000/09/xmldsig#"
            >
            <KeyName>Rsa Key</KeyName>
            <CipherData>
              <CipherValue>EHjziAiHAP/weoa8MS6tT8n1+Nd7Zsg1Fou89pG3Ya8fKQLvUuyUF0Jwvj7Na1Cn
            </CipherData>
            </EncryptedKey>
          </KeyInfo>
          <CipherData>
            <CipherValue>Tntg1uqZ5LF9ENrviVvMbbzwK3w1Q72mbv1vSauer3eiRfw16DMos3yx7Wgi0OPVK2IHLV0
          </CipherData>
          </EncryptedData>
        </connectionStrings>
      </EncryptedData>
    </connectionStrings>
  </configuration>
  <system.web>
    <!-- -->
    <trace enabled="false" />
    <!-- -->
    <deployment retail="true"/>
    <!-- -->
    <compilation debug="false" />
    <!-- -->
    <sessionState cookieName="s" />
  </system.web>
</configuration>
```

Şekil 4. Yapılandırma Dosyasında Şifreli Halde Tutulan

Bir Bağlantı String Bloğu Gösterimi

Web.Config dosyasına şifrelenmiş connection strings'i c# kodlaması içerisinde kullanabilmek için

C# :

```
ConnectionStringsSection encryptedConnectionStrings =
WebConfigurationManager.OpenWebConfiguration().GetSection("connectionStrings")
as ConnectionStringsSection;

ConnectionStringSettings decryptedConnectionStrings =
encryptedConnectionStrings.SectionInformation.UnprotectSection() as
ConnectionStringSettings;

SqlConnection con = new SqlConnection(decryptedConnectionStrings);
```

şeklinde sırasıyla web.config yapılandırma dosyasının açılması, connectionStrings bloğunun cımbızlanması, sonra bu bloğun çözülmesi (decrypt edilmesi) ve son olarak çözülen connection string'in sql bağlantı kurma metoduna koyulması gerekmektedir.

ii) Web.Config Veri Bağlantılarını Çözme (Decryption)

Web.Config yapılandırma dosyasında şifrelenmiş (encrypt edilmiş) connection strings bloklarını düzenlemek / modifiye etmek maksadıyla çözmek için (decrypt etmek için) yine aspnet_regiis.exe aracı kullanılabilir:

Not: Şifrelenmiş (encrypt edilmiş) bir web.config dosyası sadece şifrelendiği makinada çözülebilir (decrypt edilebilir).

Başlat:

CMD (Sağ Tık -> Yönetici Olarak Çalıştır)

Ardından aspnet_regiis.exe dosyasının yer aldığı dizin tespit edilir:

CMD (as Administrator):

```
C:\Windows\system32\> dir /s c:\aspnet_regiis.exe
```

Sıralanan aspnet_regiis.exe araç dizin yollarından .NET framework versiyonu en yüksek olan dizin yoluna gidilir. Örn;

CMD (as Administrator):

```
C:\Windows\system32\> cd "C:\Windows\Microsoft.NET\Framework64\4.0.30319"
```

Ardından aspnet_regiis.exe aracı verilecek -pdf parametresi ve argüman değerleriyle çalıştırılır:

CMD (as Administrator):

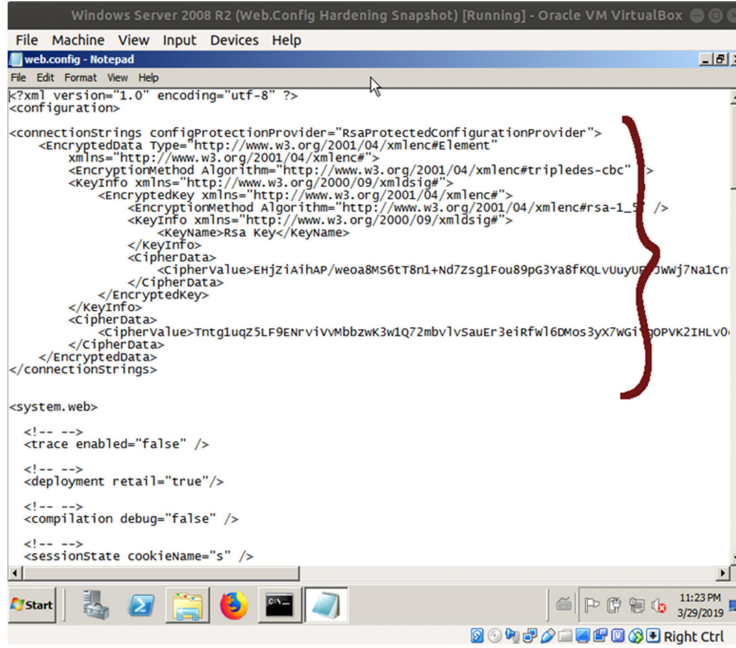
```
C:\Windows\Microsoft.NET\Framework64\4.0.30319> aspnet_regiis.exe -pdf "connectionStrings" "C:\inetpub\wwwroot"
```

Çıktı:

```
Decrypting configuration section...
```

```
Succeeded!
```

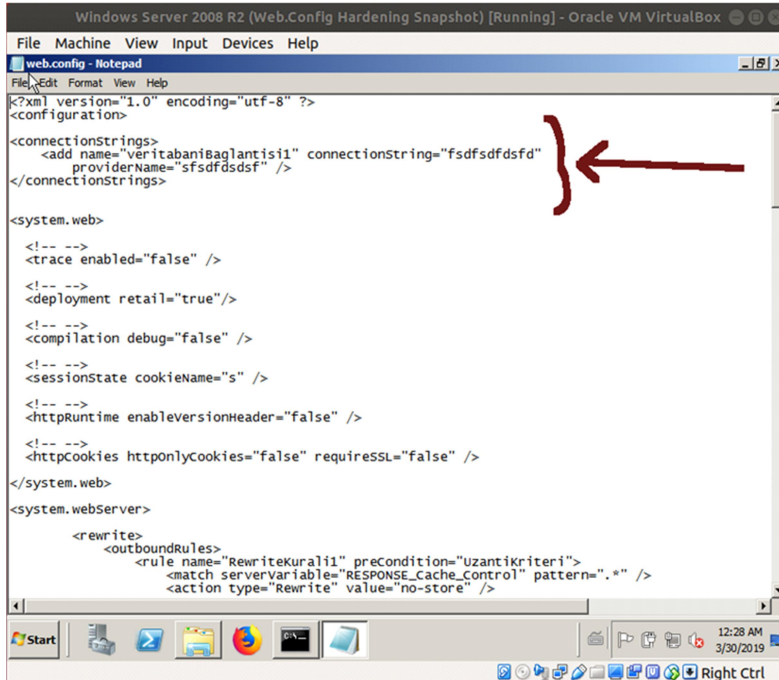
~ Şifrelenmiş Bloğu Çözmeden Önce ~



```
Windows Server 2008 R2 (Web.Config Hardening Snapshot) [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
web.config - Notepad
File Edit Format View Help
<?xml version="1.0" encoding="utf-8" ?>
<configuration>
<connectionStrings configProtectionProvider="RsaProtectedConfigurationProvider">
  <EncryptedData Type="http://www.w3.org/2001/04/xmlenc#Element"
    xmlns="http://www.w3.org/2001/04/xmlenc#">
    <EncryptionMethod Algorithm="http://www.w3.org/2001/04/xmlenc#tripledes-cbc" />
    <KeyInfo xmlns="http://www.w3.org/2000/09/xmldsig#">
      <EncryptedKey xmlns="http://www.w3.org/2001/04/xmlenc#">
      <EncryptionMethod Algorithm="http://www.w3.org/2001/04/xmlenc#rsa-1_5" />
      <KeyInfo xmlns="http://www.w3.org/2000/09/xmldsig#">
        <KeyName>Rsa Key</KeyName>
      </KeyInfo>
      <CipherData>
        <CipherValue>EhJZiaIhAP/weoa8MS6tT8n1+Nd7Zsg1Fou89pg3Ya8fKQLvUuyUfjwWj7Na1Cn
        </CipherValue>
      </CipherData>
      </EncryptedKey>
    </KeyInfo>
    <CipherData>
      <CipherValue>Tntg1uq25LF9ENrviVvmbzkw3w1Q72mbvlvSauer3eiRfWl6Mos3yx7Wg100PVK2IHLV0
      </CipherValue>
    </CipherData>
  </EncryptedData>
</connectionStrings>
</configuration>
<system.web>
  <!-- -->
  <trace enabled="false" />
  <!-- -->
  <deployment retail="true"/>
  <!-- -->
  <compilation debug="false" />
  <!-- -->
  <sessionState cookieName="s" />
</system.web>
```

Şekil 5. Yapılandırma Dosyasında Şifreli Halde Tutulan Bağlantı String Bloğu

~ Şifrelenmiş Blok Çözüldüğünde ~



```
Windows Server 2008 R2 (Web.Config Hardening Snapshot) [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
web.config - Notepad
File Edit Format View Help
<?xml version="1.0" encoding="utf-8" ?>
<configuration>
<connectionStrings>
  <add name="VeritabaniBaglanti1" connectionString="fsdfsdfsdf"
    providerName="sfsdfsdf" />
</connectionStrings>
</configuration>
<system.web>
  <!-- -->
  <trace enabled="false" />
  <!-- -->
  <deployment retail="true"/>
  <!-- -->
  <compilation debug="false" />
  <!-- -->
  <sessionState cookieName="s" />
  <!-- -->
  <httpRuntime enableVersionHeader="false" />
  <!-- -->
  <httpCookies httpOnlyCookies="false" requireSSL="false" />
</system.web>
<system.webServer>
  <rewrite>
    <outboundRules>
      <rule name="RewriteKural1" precondition="uzantikriteri">
        <match serverVariable="RESPONSE_cache_control" pattern=".*" />
        <action type="Rewrite" value="no-store" />
      </rule>
    </outboundRules>
  </rewrite>
</system.webServer>
```


Şekil 6. Yapılandırma Dosyasında Şifresi Çözülmüş

Bağlantı String Bloğu Gösterimi

Ek:

a. Web.Config İçerisinde Birden Fazla Veri Bağlantı String'i Tanımlama

Web uygulamanız içerisinde birden fazla çeşitte (rolde) veritabanı bağlantısı kuruyor olabilirsiniz. Bu durumda web.config yapılandırma dosyasında her biri için ayrı ayrı connection strings (veri bağlantı string'leri) tanımlaması yapılabilmektedir ve C# ile bu bağlantılar çekilebilmektedir.

Örneğin web.config yapılandırma dosyasına 3 adet veri bağlantısı string tanımlamasının eklendiği örneği aşağıda görmekteyiz.

```
<configuration>
  <connectionStrings>
    <add name="conn1"
      providerName="Veri Sağlayıcısı İsmi"
      connectionString="Geçerli Bir Veri Bağlantısı String'i 1" />
    <add name="conn2"
      providerName="Veri Sağlayıcısı İsmi"
      connectionString="Geçerli Bir Veri Bağlantısı String'i 2" />
    <add name="conn3"
      providerName="Veri Sağlayıcısı İsmi"
      connectionString="Geçerli Bir Veri Bağlantısı String'i 3" />
  </connectionStrings>
</configuration>
```

Bu durumda C# kodlamada aşağıdaki gibi bir kullanımda bulunulması gerekir.

```
var conn1 = ConfigurationManager.ConnectionStrings["conn1"].ConnectionString;
var conn2 = ConfigurationManager.ConnectionStrings["conn2"].ConnectionString;
var conn3 = ConfigurationManager.ConnectionStrings["conn3"].ConnectionString;
```

b. Harici Yapılandırma Dosyası Kullanma

Web.Config içerisine yerleştirilen bloklar dilerse harici bir dosyaya konulup web.config içerisinde dahil edilmek suretiyle kullanılabilir. Bunun avantajı deploy edilen web uygulamalarının web.config dosyası içerisinde değişiklik yapılması ihtiyacı duyulduğunda değişikliğin uygulanabilmesi için uygulamanın tekrar deploy edilip sunucuya konulmasını mahal bırakmadan dinamik olarak yapılan her değişikliğin deploy edilmiş web uygulamasında

uygulanabilmesini sağlamasıdır. Örneğin veri bağlantı string'leri zaman zaman değiştirilme ihtiyacı duyulan bir yapılandırma ayarıdır. Bu nedenle bu yapılandırma ayarının (<connectionStrings> ... </connectionStrings> bloklarının) harici bir yapılandırma dosyasına taşınıp web.config içerisine dahil edilmek suretiyle kullanımı tercih edilebilmektedir.

Harici yapılandırma dosyasına connection string'lerinizi taşımak ve o şekilde kullanmak isterseniz hazırlayacağınız harici yapılandırma dosyası web.config'deki gibi tanımlamaları içermeyen yalnızca dahil edilmek istenen bloğu içerecek şekilde hazırlanmalıdır. Örn;

C:\inetpub\wwwroot\connections.config İçeriği:

```
<connectionStrings>
  <add name="Name"
        providerName="Veri Sağlayıcı İsmi"
        connectionString="Geçerli Bir Veri Bağlantısı String'i;" />
</connectionStrings>
```

Bu harici yapılandırma dosyası web.config ana yapılandırma dosyası içerisine ise şu şekilde dahil edilmelidir:

C:\inetpub\wwwroot\web.config İçeriği:

```
<?xml version='1.0' encoding='utf-8'?>
<configuration>
  ... <!--size ait var olan kodlar -->
  <connectionStrings configSource="connections.config"/>
  ... <!--size ait var olan kodlar -->
</configuration>
```

Referanslar:

1. <https://cwe.mitre.org/data/definitions/312.html>
2. <https://gist.github.com/marcbarry/47644b4a43fbfb63ef54>
3. <https://stackoverflow.com/questions/550210/uses-for-machinekey-in-asp-net>
- 4.