Formları FİLTRELEME

Web uygulamalarında sıklıkla kullanılan formlar , kullanışlı olmasının yanı sıra bir o kadarda güvenlik konusunda hassastır. Oluşturulan formlara göre kişisel bilgiler, ödeme bilgileri , gizli mesajların yer aldığı bu formları bazı filtreler ekleyerek daha güvenli hale getirebiliyoruz.

Web Güvenliğinde bahsedeceğimiz XSS açıklarının bir çoğuda FORMlarda bulunur.

Güvenlik açığını ortadan kaldırmak için formlara girilen bilgilerin belli bir süzgeçten geçirilerek veri tabanına kaydedilmesi gerekir.

PHP 5.2 sürümü ile filtreleme fonksiyonları kullanıma sunulmuştur.

Birçok satırdan oluşan kodları yazmak yerine birkaç satır kod ile form denetimlerini yapmak mümkündür.

Formdan giden iframe ya da javascript kodu -XSS-Cross Site Scripting saldırısına maruz kalabilir. Kısaca paylaşımın olduğu sayfalarda üyelerin oturum bilgileri,kişisel ve banka bilgileri ele geçirilebilir.

Filtreleme Fonksiyonları

filter_input() - Metoda göre tek alanı filtreler.

filter_input_array() - Metoda göre birden fazla alanı filtreler.

FILTER_SANITIZE_STRING - HTML etiketlerini ve özel karakterleri temizler.

FILTER_SANITIZE_SPECIAL_CHARS - HTML etiketlerini özel karakterlerle kodlayarak kullanIcı tarafında görülmesini sağlar.

FILTER_SANITIZE_ENCODED - Bütün özel karakterleri % ön eki etkisiz yapar.

FILTER_SANITIZE_EMAIL - Eposta adresinde bulunan istenmeyen !#$%&*’+-/=^_{|}.[]@ gibi karakterleri temizler.

FILTER_SANITIZE_URL - Site adresinde bulunan ve adrese uymayan bütün özel karakterleri temizler.

FILTER_REQUIRE_ARRAY - Veriyi dizi olmaya zorlar.

filter_input() ile başlayan fonksiyonların ilk parametresi veri gönderim metodu sabiti olmalıdır. İkinci parametre ise filtre edilmek istenen form adı anahtar adı ve bu anahtarla gelen bilginin filtrelenmesi için kullanılacak sabit değişkendir.

formguvenlik.php:

```
<?php
if(!$_POST){


?>
<style>
h4{
	margin:0;
}
input {
	display:block;
}
</style>
<h3>Yardım FORMU</h3>
<form action="" method="post">
<h4>Konuyu yazın</h4>
<input type="text" name="konu"/>
<h4>Mesajınız...</h4>
<textarea rows="3" cols="30" name="mesaj"></textarea>
<input type="submit" value="Gönder" />
</form>
<?php
}
elseif ($_POST){
```

```
        $secilen = array ('konu' => FILTER_SANITIZE_STRING,
                    'mesaj' => FILTER_SANITIZE_STRING);


        $giris = filter_input_array(INPUT_POST, $secilen);
      echo "Konu alanından gelen bilgi : ".$giris['konu'];
                echo "<br>";
                echo "Mesajdan gelen bilgi : " .$giris['mesaj'];
}
?>
```

Burada görüldüğü gibi tehlikeli javascript gibi kodlar filtrelerle temizlendi.

Sadece js değil aynı zamanda HTML etiketleride temizlenebilir.

Bu örnekte, FILTER_SANITIZE_STRING yerine FILTER_SANITIZE_SPECIAL_CHARS sabiti ile HTML etiketlerini sadece kullanıcı tarafında okunabilir kod olarak görünmesini sağlayabiliriz.

$secilen = array('konu' =>FILTER_SANITIZE_STRING, 'mesaj' => FILTER_SANITIZE_SPECIAL_CHARS);

Form kısmına XSS saldırısı deneyin aradaki farkı göreceksiniz.

---

filter_var_array() - Birden çok değeri filtreler.

filter_var() - Tek giriş için özel değerleri filtreler.

```
<?php
//E postanın doğru yazılmasını kontrol eder.
$email = 'deneme@test.com';
if (filter_var($email, FILTER_SANITIZE_EMAIL)) {
        echo "Eposta doğru formatta yazılmıs";
}else{
        echo "Hatalı eposta adresi ! ";
```

```
}
?>
```

```
<?php
//IP adresinin doğru yazıldığını kontrol eder.
$ip = '127.0.0.1';
if (filter_var($ip, FILTER_VALIDATE_IP)) {
        echo "IP doğru";
}else{
        echo "IP adresi yanlış";
}
//Aynı şekilde FILTER_VALIDATE_URL urlyi kontrol eder.
//
?>
```

PROGRAMA DIŞARDAN DOSYA DAHİL ETMEK

Var olan programa HTML veya PHP dosyası dahil etmek istenebilir. PHP de kod yazdıkça denetim yapmak ve değişiklik yapmak zorlaşabilir.

Dosyalar dahil ederek bütünlüğü sağlayabiliriz.

Bazı fonksiyonları kullanarak dışardan dosya dahil edebiliriz.

require - Belirtilen dosyayı programa ekler,dosyayı bulamazsa programı durdurur.

include - Belirtilen dosyayı ekler, bulamazsa ekrana uyarı yazar.

require_once - Dosyayı ekler, aynısının aynı anda birden fazla yüklenmesine izin vermez.

include_once - require_once ile aynı şeyi yapar.

---

dosyaekleme.php

```
<?php
include('a.php');

echo $a;
echo "<br>";
echo $b;

include('b.php');
?>
```

---

a.php

```
<?php
echo '<h3>Dosya ekleme</h3>';
$a = 'Yıl';
$b = 2020;
?>
```

---

b.php

```
<?php
echo "<h3>Web sitesi</h3>";
echo "<a href='http://toros.edu.tr'>Buraya Tıkla</a>";
?>
```

---

sayfa.php

```
<table border="1" style="width:500pt; height:250pt; border-collapse:collapse;">
<tr>
 <td valign="top" colspan="2">
  <?php include("ust.php"); ?>
  </td></tr>
  <tr>
  <td width="200" valign="top">
  <?php include("menu.php"); ?>
  </td>
  <td valign="top">
  <?php echo"Content/İçerik kısmı burada olacak";?>
  </td>
</tr>
<tr>
<td colspan="2">
<?php include("footer.php"); ?>
</td>
</tr>
</table>
```

---

ust.php

```
<?php
echo "<h2>Başlık Kısmı</h2>";
?>
```

---

menu.php

```
<?php
echo "<b>1.Menü </b> <br>";
echo "<b>2.Menü </b> <br>";
echo "<b>3.Menü </b> <br>";
echo "<b>4.Menü </b> <br>";
echo "<b>5.Menü </b> <br>";
?>
```

footer.php

```
<?php

echo "<b>&copy; Footer kısmı 2020..</b>";

?>
```