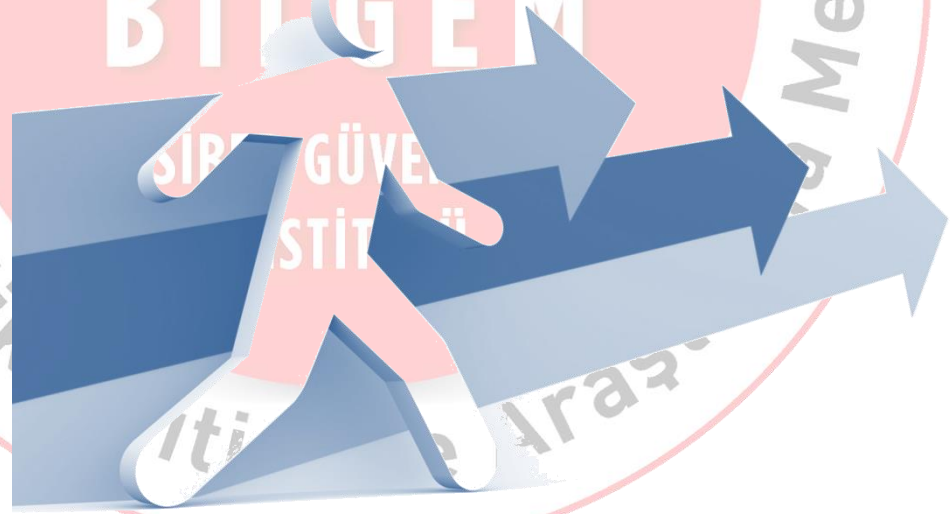
Web Uygulama Güvenliği

TÜBİTAK BİLGEM
Siber Güvenlik Enstitüsü

Eğitimin Amacı

- HTTP tabanlı uygulamaların çalışma ilkeleri
- HTTP tabanlı uygulamalarda bulunabilecek açıklıklar
- HTTP tabanlı uygulamalarda bulunan açıklıkları giderme yöntemleri
- Güvenli web uygulama geliştirme esasları

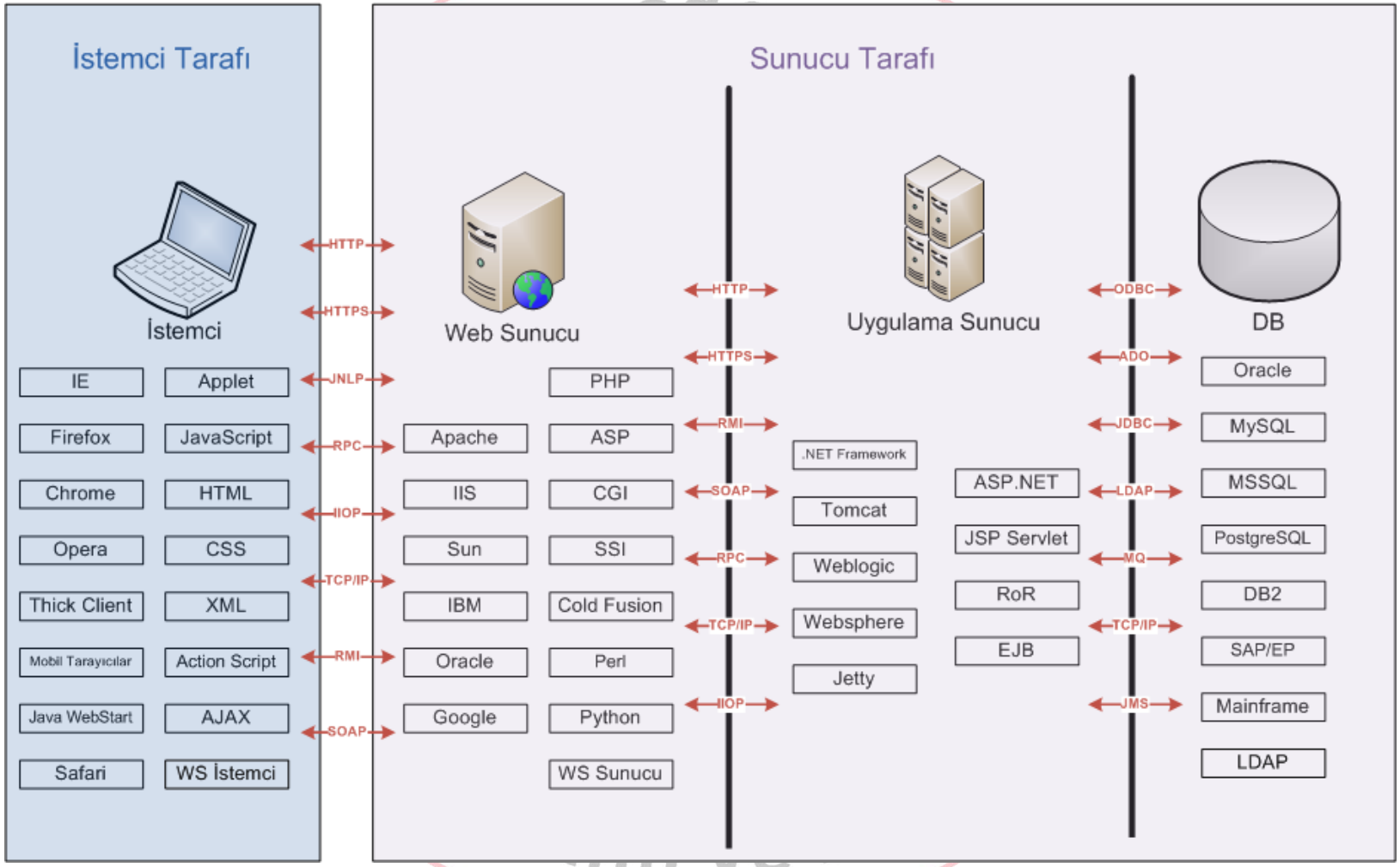


Sunum İçeriği

- Genel Bakış
- Bilgi Toplama
- Girdi & Çıktı Denetimi
- Oturum Yönetimi
- Kimlik Doğrulama & Yetkilendirme
- İş Mantığı Problemleri
- Ayar Yönetimi

Genel Bakış

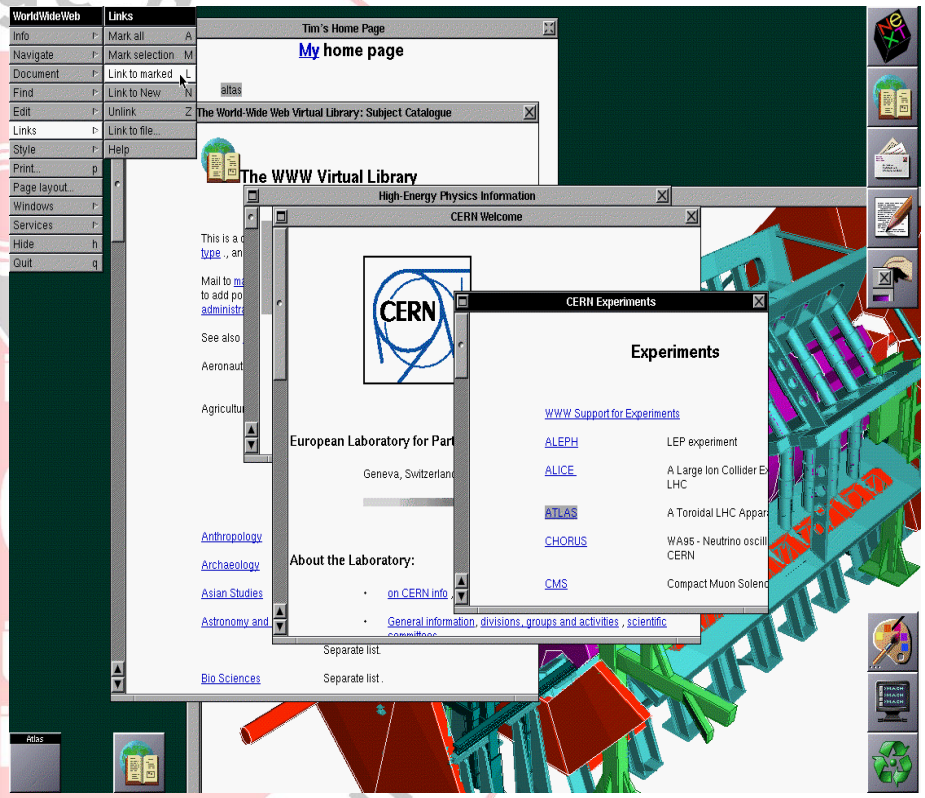
Genel Bakış



Genel Bakış

WWW

- **Ne:** WWW
- **Ne zaman:** 1989 Yılında
- **Nerede:** CERN'de
- **Kim:** Tim Berners-Lee tarafından geliştirildi
- **Neden:** Doküman paylaşımı
- **Nasıl:**
  - Oluşturmak ve biçimlendirmek için bir dil: **HTML**
  - Bulmak / erişmek için bir adresleme: **URL**
  - Transfer etmek için bir protokol: **HTTP**



HTML

Hyper Text Markup Language

- Web tarayıcılarının dokümanlardaki yazı ve grafik biçimlerini yorumlayabilme standardı.
- Web sayfalarının temelini oluşturur.
- Temel yapı taşı etiketlerdir. (Tag)

```
<html>
    <head>
        <title>SGE TEST PAGE</title>
    </head>
    <body>
        <h1>TUBITAK BILGEM</h1>
        <p>Siber Guvenlik Enstitusu</p>
    </body>
</html>
```

HTML Görünümü

Tarayıcı Görünümü

URL

Uniform (Universal) Resource Locator

- Universal Document Identifier - Uniform Resource Identifier
- Web uygulamalarına erişim için adresleme standardı
- ! * ’ ( ) ; , : @ & = + $ / ? % # [ ] Karakterleri rezerve karakterlerdir.
- Protokol://host:port/doküman_yolu?parametre=değer#fragment

Protokol (şema-schema)	http, https, ftp,
host:port	www.bilgiguvenligi.gov.tr:443
doküman_yolu	/, /index.php, /home/default.aspx
?parametre=değer	?haberno=3, ?id=manager
#fragment	doc.pdf#page=23, index.html#elementid

HTTP

Hyper Text Transfer Protocol

- Uygulama katmanı (Layer 7) protokolüdür.
- Taşıma Katmanında (Transport Layer – Layer4) paketler TCP protokolü üzerinden taşınır.

HTTP Mesajları

- HTTP Talebi (HTTP Request)
  - GET, POST, HEAD, PUT, DELETE, TRACE, OPTIONS ...
- HTTP Yanıtı (HTTP Response)
  - 200 OK, 302 Found, 404 Not Found, 500 Internal Serve Error ...

HTTP Talebi

GET

- Bir kaynağın (web sayfası, imaj, betik) çağırılması
- En çok kullanılan HTTP talebi
- Parametreler URL'de gönderilir

```
<img src=http://www.example.com/index.php?param=1 />
```

```
<iframe src=http://www.example.com/index.php />
```

```
GET /pages.php?pageid=4&article=3 HTTP/1.1
Host: www.mywebpage.com
User-Agent: Mozilla/5.0 Gecko/20100101 Firefox/12.0
Accept: text/html
Accept-Language: en-us,en;q=0.5
Accept-Encoding: gzip, deflate
Proxy-Connection: keep-alive
Cookie:PHPSESSIONID=c466a0bfe95ddd1e25ffe31
```

demo

HTTP Talebi

POST

- Veri göndermek için
- Parametreler gövde (body) kısmında gönderilir.

```
<form name=form action=login.php method=post >
        <input name=Username type=text value=myusername />
        <input name=Password type= text value=mysecretpass />
</form>
```

```
POST /accounts/login.php HTTP/1.1
Host: www.mywebpage.com
User-Agent: Mozilla/5.0 Gecko/20100101 Firefox/12.0
Accept: text/html
Proxy-Connection: keep-alive
Cookie:PHPSESSIONID=c466a0bfe95ddd1e25ffe31

Username=myusername&Password=mysecretpass
```

demo

HTTP Yanıtı

Yanıt Kodları

- 1xx: Bilgi Verme Amaçlı
- 2xx: Başarılı İstek
- 3xx: Yönlendirme
- 4xx: İstemci Taraflı Hata
- 5xx: Sunucu Taraflı Hata

```
HTTP/1.1 200 OK
Date: Tue, 08 Apr 2012 07:18:18 GMT
Server: Apache-Coyote/1.1
Pragma: No-cache
Cache-Control: no-cache,no-store,max-age=0
Content-Length: 18413
Connection: close

<html xmlns="http://www.w3.org/1999/xhtml">
<body>
<a href=http://www.tubitak.gov.tr>TUBITAK</a>
...
```

Kritik Kavramlar

DOM (Domain Object Model)

- Programların ve betik kodlarının dokümanların
  - İçeriğine
  - Yapısına
  - Biçim özelliklerine

  erişimini ve kullanımını sağlayan bir arabirim.
- Platform ve dilden bağımsızdır.

DOM Document, DOM Events, DOM Elements, DOM Anchor, DOM Area, DOM Body…

```
<script>window.location=http://www.bilgiguvenligi.gov.tr</script>

<script>alert(document.cookie)</script>
```

Kritik Kavramlar

SOP (Same Origin Policy)

- Farklı kaynakların (origin), birbirlerinin metodlarına ve özelliklerine erişmelerini engeller
- Tarayıcı tarafında çalışacak betik (scripting) dillerinin güvenlik sınırlarını belirleyen en önemli kural
- Same Origin = Domain + Protokol + Port

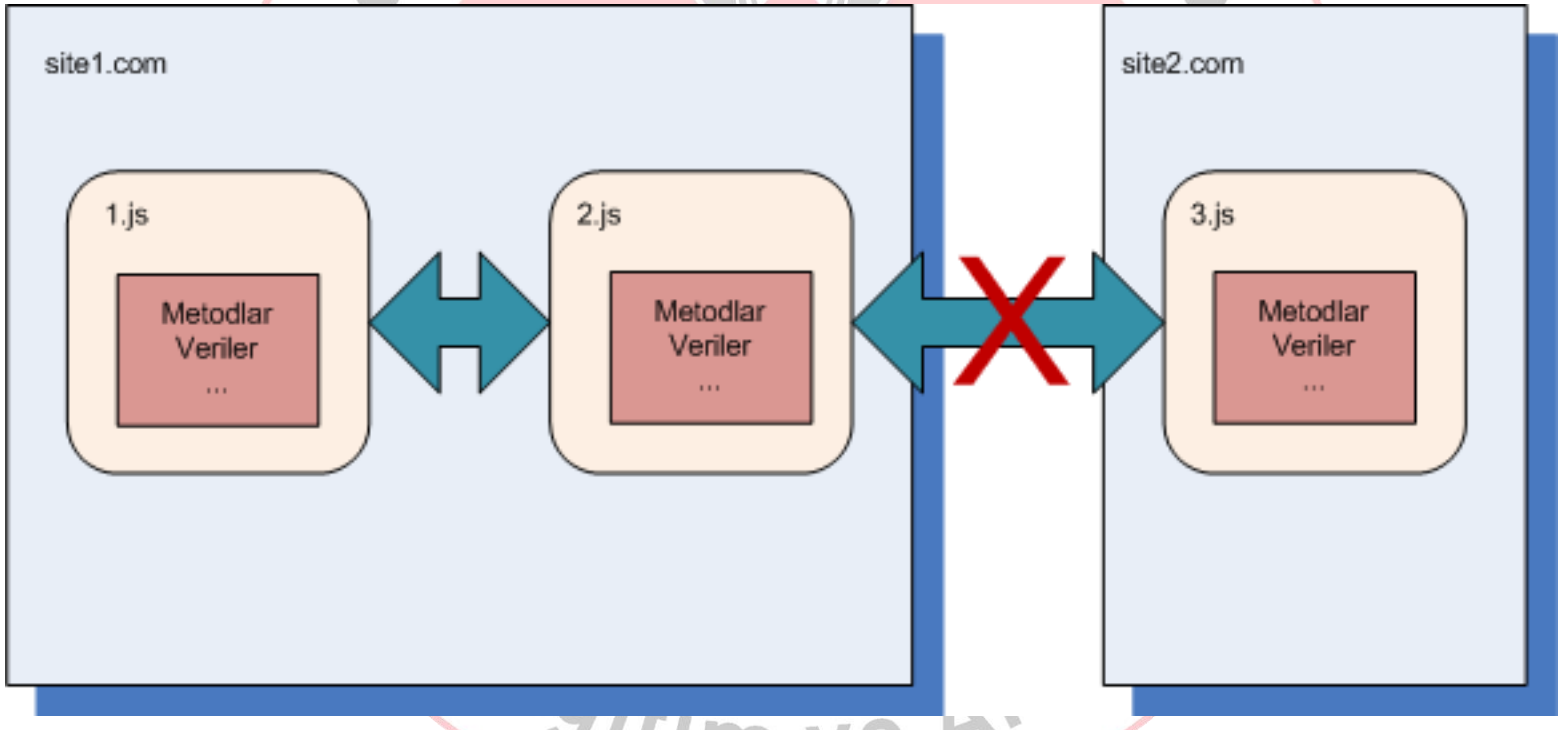
```
http://www.example.com/dir/page.html

http://www.example.com/dir2/other.html
http://www.example.com:81/dir/other.html
https://www.example.com/dir/other.html
http://en.example.com/dir/other.html
http://example.com/dir/other.html
http://v2.www.example.com/dir/other.html
```

Kritik Kavramlar

SOP (Same Origin Policy)

- Örnek



Kritik Kavramlar

HTML5

- Yeni etiketler
  - Audio, canvas, source, video
- Yeni metodlar
  - onbeforeonload, onmousemove, onerror, onpagehide, onpageshow, onforminput, oninvalid, autofocus+onfocus
- Base64 kodlama desteği
- History.pushState

Kritik Kavramlar

HTML5

- İstemci tarafı veri depolama ve FileSystem API
  - Verilerin istemci taraflı saklanması ve gizlilik
  - Çevrimdışı e-posta ve takvim kullanımı
- WebSockets, WebWorkers
  - Uygulama kodu üzerinden socket bağlantıları
  - Veri denetimi

Kritik Kavramlar

HTML5

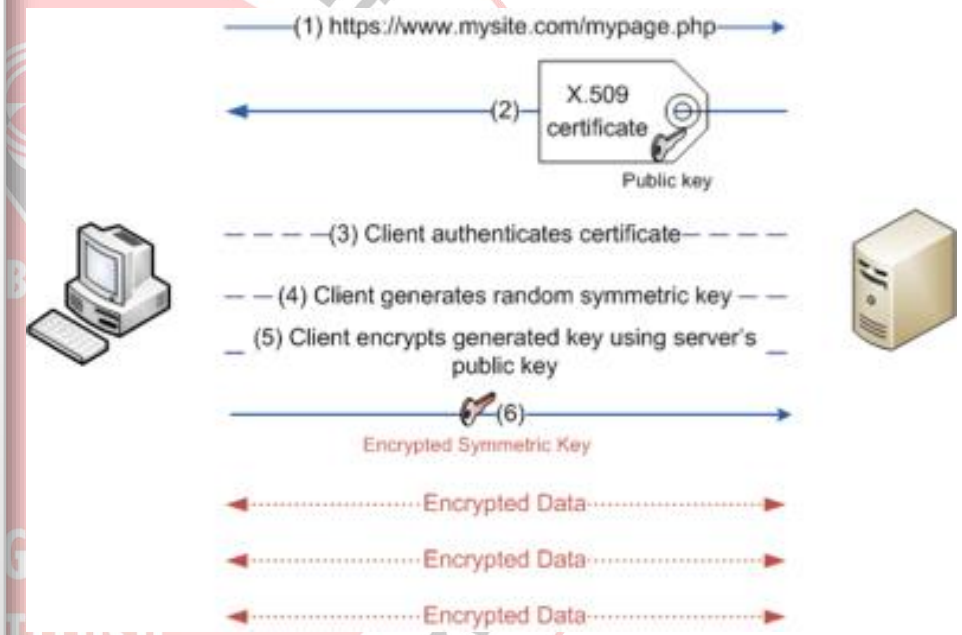
- Cross-document messaging, CORS
  - Iframe, sekme ve pencereler arası etkileşim
- WebGL, WebCL
  - JavaScript ile 3D, OpenGL, Canvas etiketi, GLSL ile GPU kullanımı

Kritik Kavramlar

Bağlantı Güvenliği

- SSL/TLS
  - Güvenli bir bağlantı oluşturulur
  - Mesajlar bu güvenli bağlantı üzerinden gönderilir.
  - SSL v3.0 ve üzeri kullanılmalıdır.

SSL



Genel Bilgiler

Web Servis Güvenliği

- SOAP 1.1 ve 1.2
  - Güvenlik Fonksiyonları
    - WS-I Basic Profile 1.1, 2.0 (WSDL), Interoperability
    - WS-Security
- SAML 1.1 ve SAML 2.0
  - Bütünleşik Oturum (Single Sign-On)
  - Federation

SAML + SOAP + HTTP + WS-Security

Genel Bilgiler

Web Servis Güvenliği

- WS-I Basic Profile
  - Basit Kimlik Doğrulama (Basic Authentication)
  - Sertifika Tabanlı Kimlik Doğrulama (Certificate-Based Authentication)
  - Veri gizliliği ve bütünlüğü (Transport Layer Security - SSL)
- WS-Security
  - X.509, Kerberos, SAML, Kullanıcı adı/Parola, Kullanıcı tanımlı yöntemler
- WS-SecureConversation, WS-Addressing, WS-Notification

Bilgi Toplama
(Information Gathering)

Bilgi Toplama

Girdi Noktaları Bulma

- Form detayları (drop-down menüler, butonlar, checkbox...)
- Linkler
- Iframe kaynakları
- Image kaynakları
- Gizli girdi alanları

Bağlantı Keşfi

- Crawling-Spidering: Uygulamadaki tüm bağlantıların çıkartılması
  - Site Linki
  -
  -
  - <form id="search" action="search.php" method="get"></form>
  - <iframe src=http://www.bir-site.com/frame1>

  ...

Bilgi Toplama

Otomatize Araçlar

- Amaç: Zafiyetleri minimum efor ile tespit etmek
- Ana Fazları:
  - Girdi alanı ve bağlantı keşfi
  - Zafiyet tespiti
- w3af, AppScan, Acunetix, Netsparker, WebInspect, BurpSuite, Nessus, Qualys GurardWAS, Skipfish, Arachni, WebSecurify...
- Otomatize araçlar her durumda (her açıklık için) etkin değildir.

Bilgi Toplama

Yayınlanmış Zafiyetler

- Uygulama yazımında kullanılan anaçatılar
- Kurulu oldukları uygulama sunucuları, web portalleri, anaçatılar
- Uygulamalarda kullanılan üçüncü parti modüller, bileşenler

Tekniğin Uyglama Adımları

1. Hedef uygulama hakkında bilgi toplama
2. İlgili telknoloji ve vesiyonları için yayınlanmış zafiyetlerin aranması (exploit-db.com v.b.)
3. Bulunan açıklık istismar yöntemlerinin denenmesi

Girdi ve Çıktı Denetimi

Girdi Denetimi

Tanım

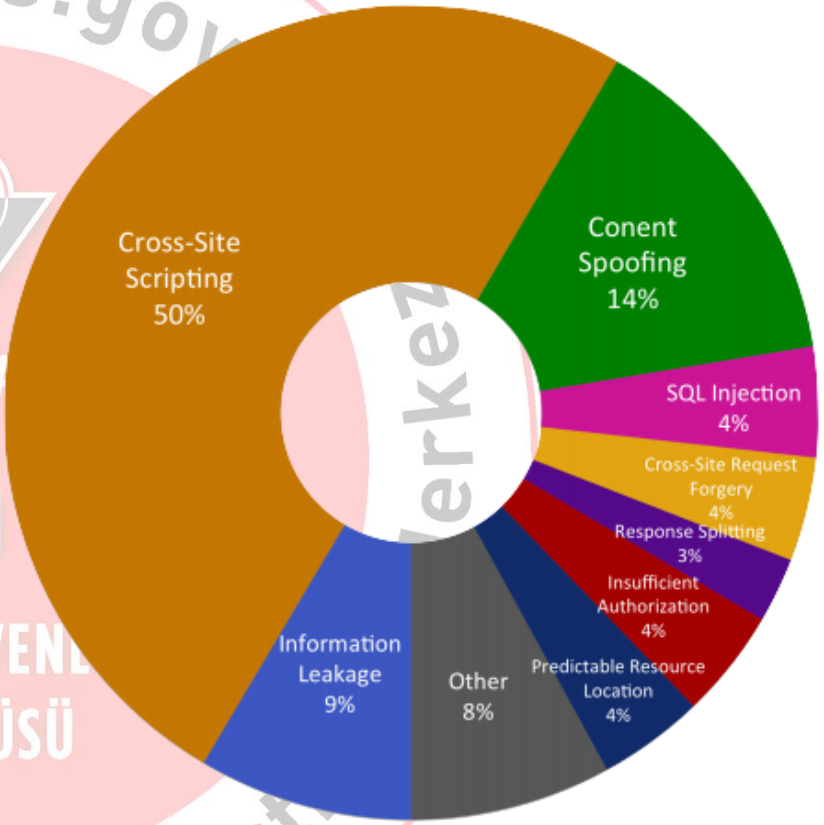
- Girdi %100 güvenilir değilse güvensizdir.
  - Kullanıcıdan alınan tüm veriler mutlaka denetlenmelidir.
- Negatif Girdi Denetimi
  - Kara Liste uygulanır.
  - İstenmeyen girdilere izin verilmez
- Pozitif Girdi Denetimi
  - Beyaz Liste uygulanır.
  - Sadece istenilen girdilere izin verilir.

Girdi denetimi yetersiz ise ne olur?

Girdi Denetimi

Cross Site Scripting - XSS

- Siteler Arası Betik Çalıştırma
- JavaScript'in keşfi ile başlar (1995)
- Bir tarayıcıda izinsiz olarak kod çalıştırmak.
- 3 Tipi Vardır:
  - Yansıtılan XSS (Reflected XSS)
  - Depolanan XSS (Stored XSS)
  - DOM Tabanlı XSS (DOM Based XSS)



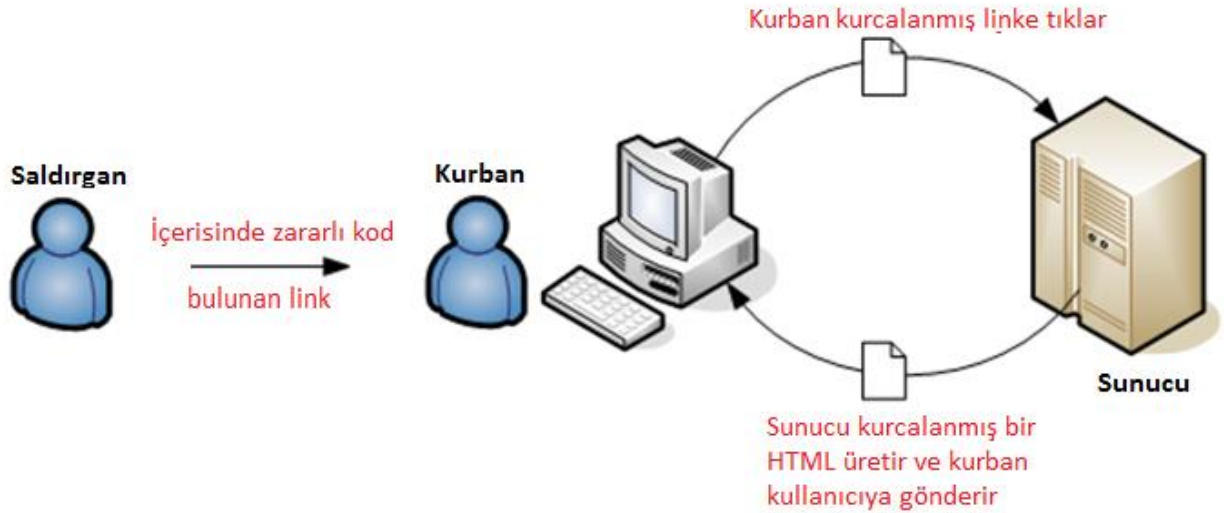
WhiteHat Security Rapor 12 (2012)*

* https://www.whitehatsec.com/assets/WPstats_summer12_12th.pdf

Cross Site Scripting - XSS

Reflected XSS (Yansıtılan XSS)

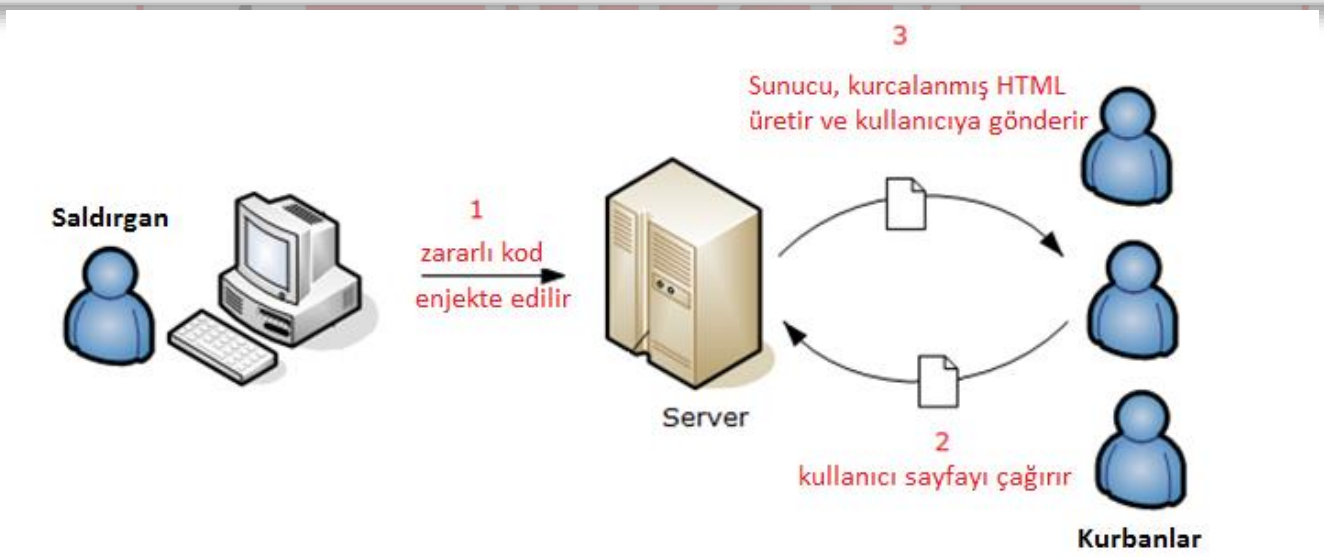
- Saldırgan, zararlı bir link hazırlar.
- Kurban kullanıcıyı bu linke tıklamaya ikna eder.
- Kurban kullanıcının tarayıcısından zararlı bir talep gönderilir.
- Linkteki zararlı HTML, web sayfası içerisine eklenerek kullanıcıya geri gönderilir.
- Kurban kullanıcının tarayıcısı zararlı HTML'i çalıştırır.



Cross Site Scripting - XSS

Stored XSS (Depolanan XSS)

- Saldırgan, zararlı kodunu veritabanı veya sunucuya enjekte eder.
- Sayfayı ziyaret eden tüm kullanıcıların tarayıcıları bu zararlı kodu çalıştırır.
- Reflected XSS'e göre daha tehlikelidir.



Cross Site Scripting – XSS

Gerçek Hayattan Bir Örnek

- Samy Worm (MySpace Worm-2005)

1. Samy profiline bir kod ekler
2. Samy'nin sayasını ziyaret eden herkes;
   i. Samy'yi arkadaş olarak ekler.
   ii. Bu kodu kendi profiline ekler

Cross Site Scripting – XSS

Gerçek Hayattan Bir Örnek

Kod eklenmeden önce: 73 arkadaş

1 saat sonra: 73 arkadaş - 1 arkadaşlık isteği

7 saat sonra: 74 arkadaş - 221 arkadaşlık isteği

1 saat sonra: 74 arkadaş - 480 arkadaşlık isteği

1 saat sonra: 518 arkadaş - 561 arkadaşlık isteği

3 saat sonra: 2503 arkadaş – 6.373 arkadaşlık isteği

5 saaat sonra: 2503 arkadaş – 917.084 istek

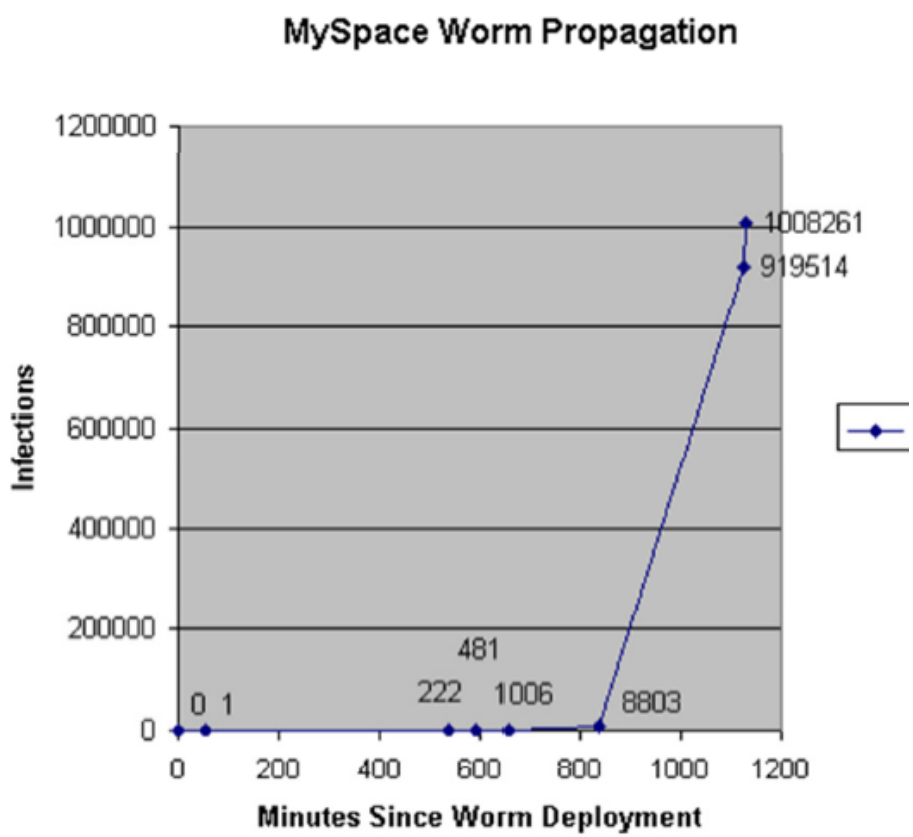
3 saniye sonra: 2503 arkadaş – 918.268 istek

3 saniye sonra: 2503 arkadaş – 919.664 istek

Birkaç dakika sonra: 2503 ark. – **1.005.831** istek

Birkaç saniye sonra:

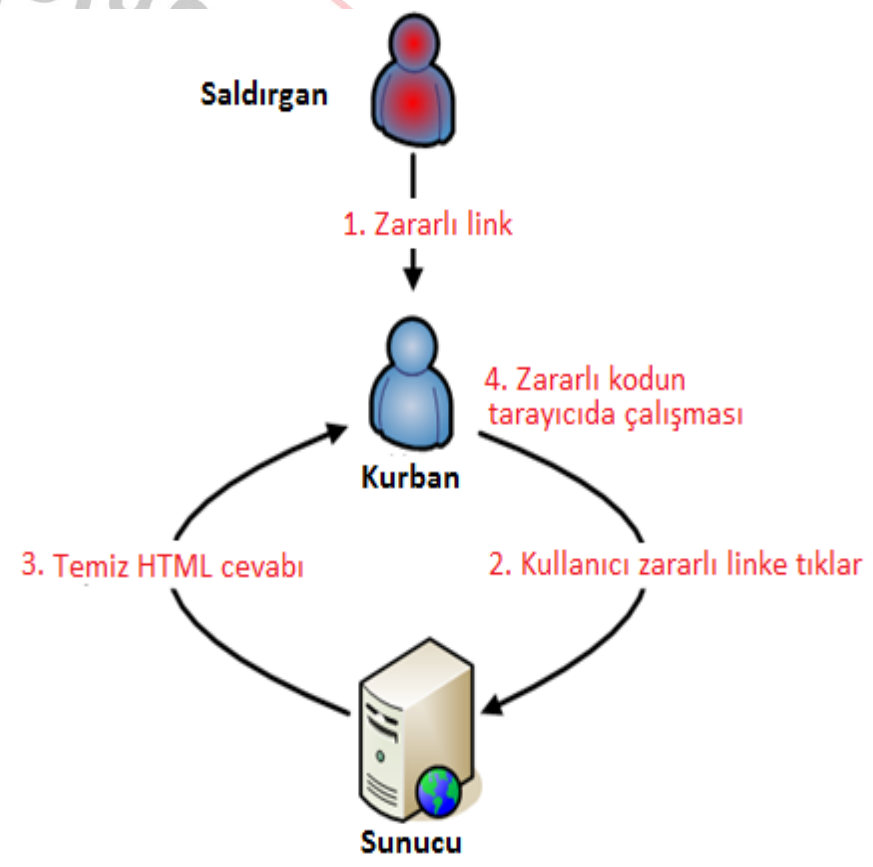
Your Profile is Down for Maintenance



Cross Site Scripting - XSS

DOM Based XSS (DOM Tabanlı XSS)

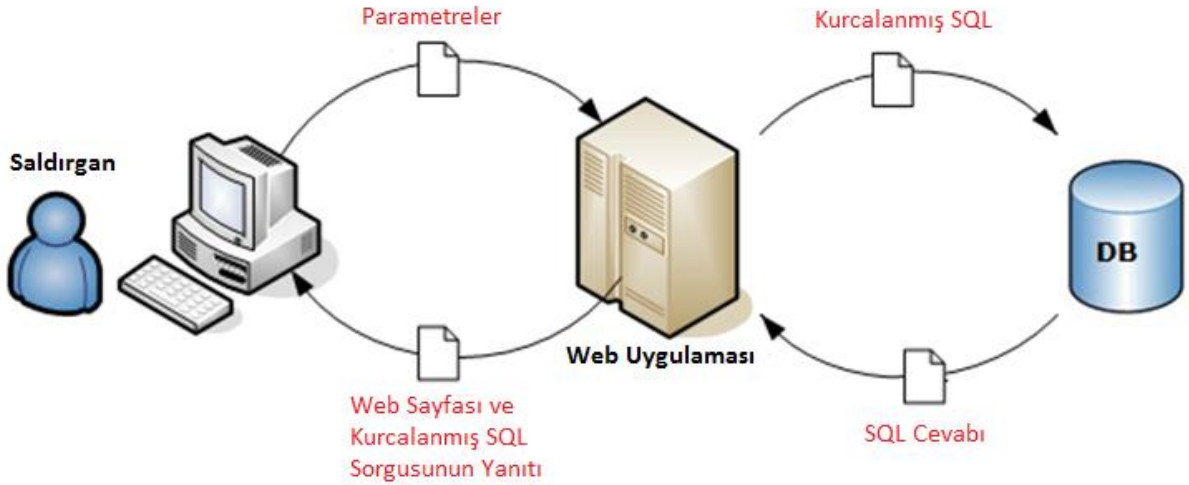
- Yansıtılan XSS'e benzer.
- Fark: Kullanıcıya gönderilen HTML zararlı bir kod içermez.
- Tarayıcı DOM objesini çağırır.
- DOM objesi zararlı kodu çalıştırır.



SQL Enjeksiyonu (SQLi)

SQL Injection

- **Tanım:** Veritabanında izinsiz SQL sorgusu çalıştırmak
- **Problem:** Kullanıcıdan alınan girdinin SQL sorgusu olarak işleme alınması



SQL Enjeksiyonu

Dinamik SQL Sorgusu

```
$sQuery = “SELECT name FROM products WHERE id=‘” + $id
+“’;”
```

Normal Değişken Değeri

```
SELECT name FROM products WHERE id=‘100’;
```

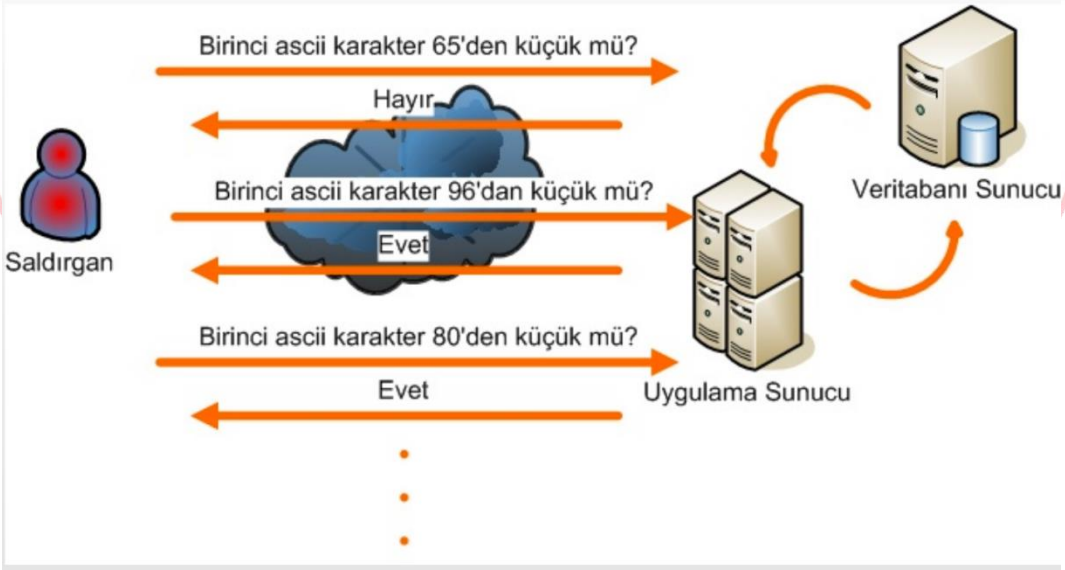
Anormal Değişken Değeri = SQL Enjeksiyonu

```
SELECT name FROM products WHERE id=‘100’ UNION SELECT
password FROM users WHERE userid=‘1’
```

Kör SQL Enjeksiyonu (Blind SQLi)

Blind SQL Injection

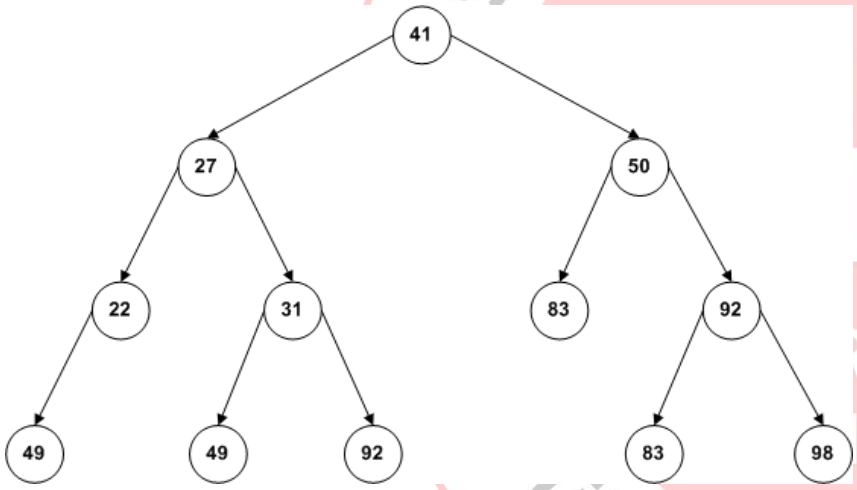
- Sunucu her deneme için aynı hata mesajını üretiyorsa
- Uygulama SQL sorgusunun sonucunda herhangi bir bilgi göstermiyorsa
- Kısaca, enjekte edilen sorgunun işleme alınıp alınmadığı bilinmiyorsa

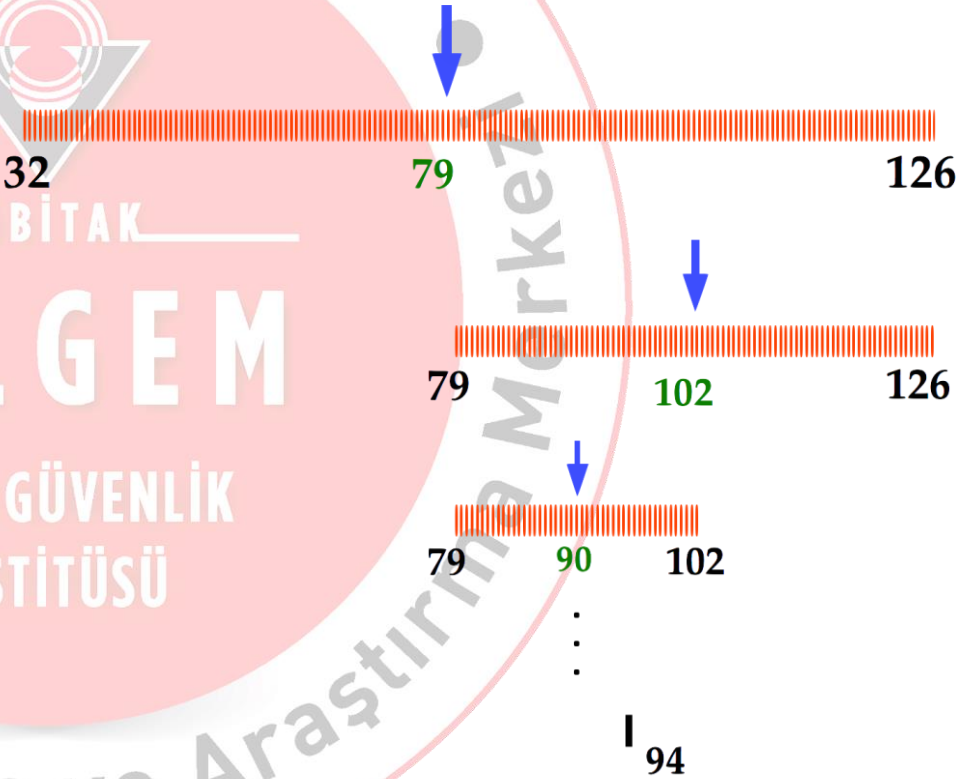


Kör SQL Enjeksiyonu (Blind SQLi)

Blind SQL Injection

- Binary Search Tree





Kör SQL Enjeksiyonu (Blind SQLi)

Boolean Based Blind SQL Injection

```
SELECT name FROM products WHERE id='1' OR 'a'='a'--'
SELECT name FROM products WHERE id='1' AND 'a'='a'--'
SELECT name FROM products WHERE id='1' AND 'a'='b'--'
SELECT name FROM products WHERE id='1 AND substring(@@version 1,1)>'5'
SELECT name FROM products
WHERE id='1 AND ascii(substring(database(),1,1))<'104'
```

Time Based Blind SQL Injection

```
SELECT name FROM products WHERE id='1' AND IF
((substring(@@version,1,1))>5,SLEEP(3),1) ='1'
SELECT name FROM products WHERE id='1' AND IF
(ASCII(lower(substring((DATABASE()),1,1)))>104,SLEEP(1),1) ='1'
```

SQL Enjeksiyonu (SQLi)

SQL Injection

- Gerçek hayata uyarlanan bir SQL Enjeksiyonu

Diğer Enjeksiyonlar

Command Injection

- Girdinin işletim sisteminda çalıştırılacak komuta parametre olarak verildiği durumda
- Girdinin denetimsiz bir şekilde işletim sisteminde çalıştırılacak kodun içerisinde yer alması

```
MacBookProuz:basics ouz$ ping webegitim.bsg.egitim
PING webegitim.bsg.egitim (192.168.1.11): 56 data bytes
64 bytes from 192.168.1.11: icmp_seq=0 ttl=64 time=0.504 ms
64 bytes from 192.168.1.11: icmp_seq=1 ttl=64 time=0.784 ms
64 bytes from 192.168.1.11: icmp_seq=2 ttl=64 time=0.784 ms
64 bytes from 192.168.1.11: icmp_seq=3 ttl=64 time=0.696 ms
64 bytes from 192.168.1.11: icmp_seq=4 ttl=64 time=0.550 ms
64 bytes from 192.168.1.11: icmp_seq=5 ttl=64 time=0.588 ms
```

Girdi Denetimi

Girdi Denetim Stratejileri

- Negatif Girdi Denetimi

```
<?php

$searchparam = $_POST['searchparam'];
$key = array('<script>', '</script>');
$replace = array(' ', ' ');
$searchparam = str_replace ($key, $replace,
$searchparam);

echo searchparam;
?>
```

Girdi Denetimi

Girdi Denetim Stratejileri

- Pozitif Girdi Denetimi

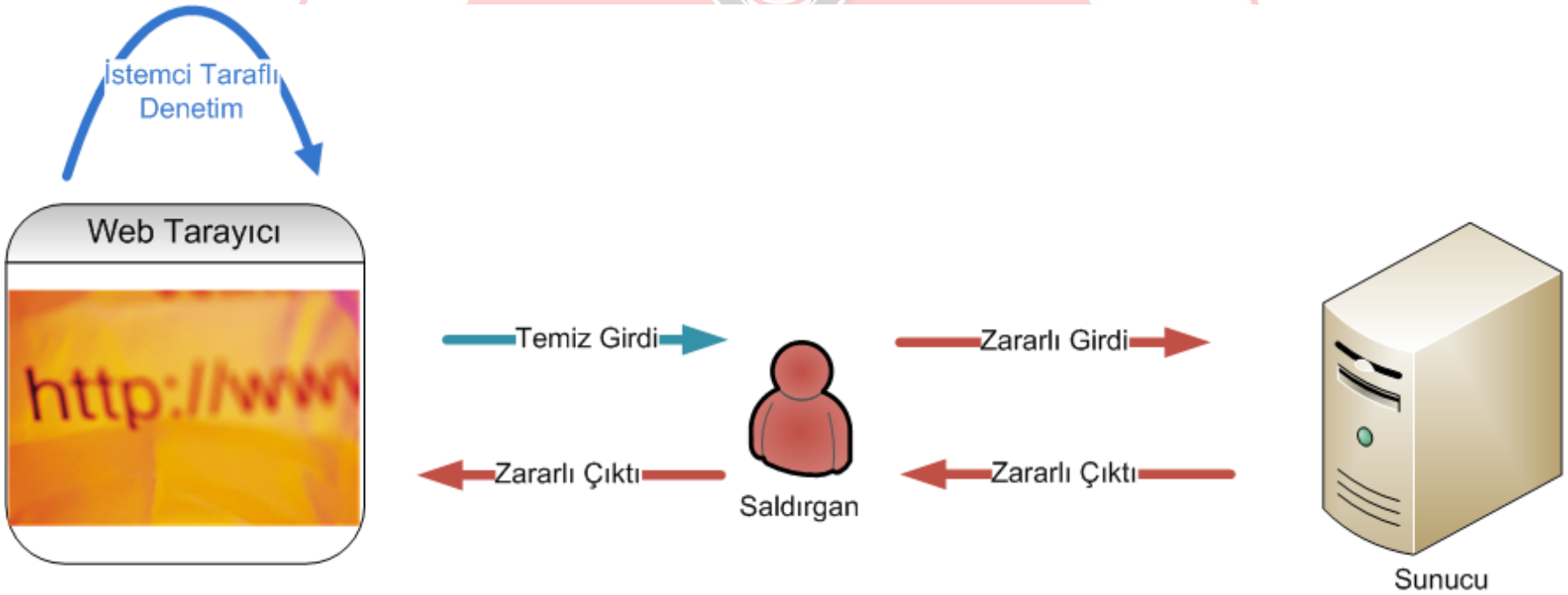
```
<?php

$searchparam = $_POST[searchparam];
if(!preg_match('[-_ 0-9A-Za-z] ', $searchparam))
{
header( 'Location: error.php' );
}
else{
echo searchparam;
?>
```

Girdi Denetimi

Girdi Denetim Stratejileri

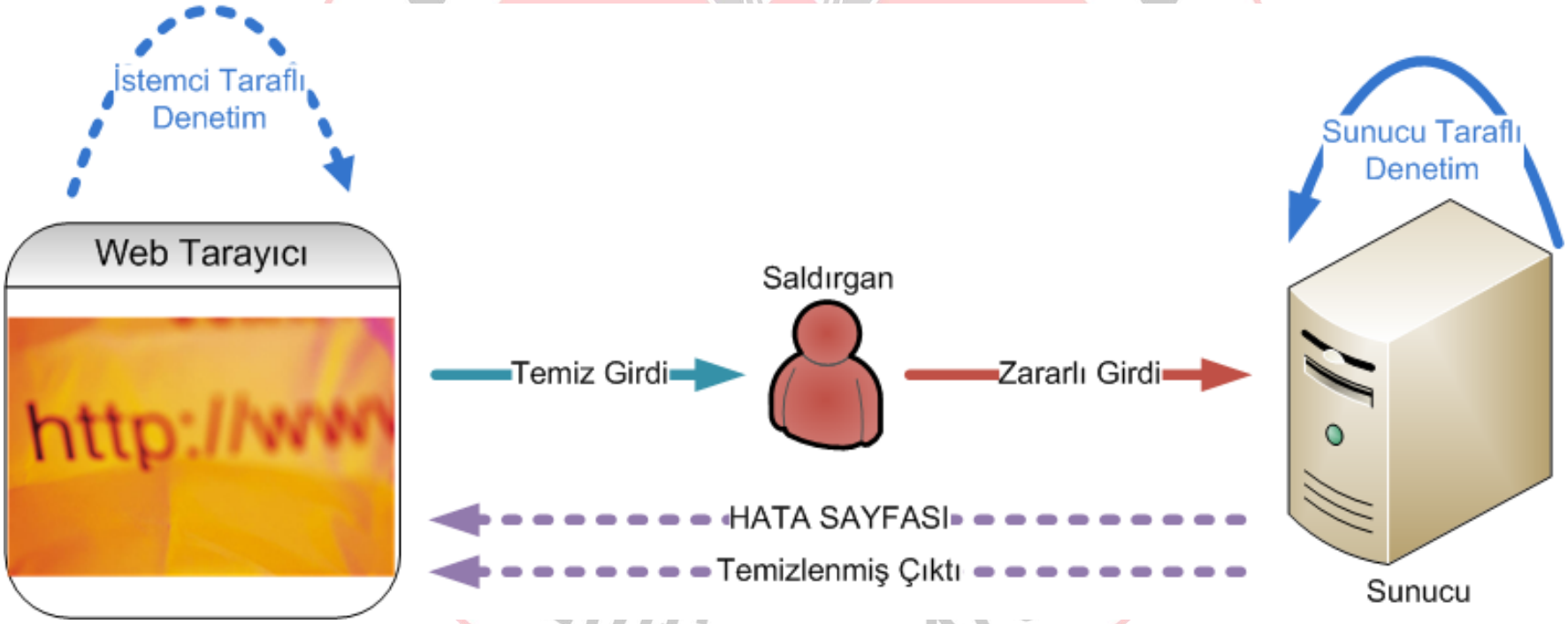
- İstemci Taraflı Denetim



Girdi Denetimi

Girdi Denetim Stratejileri

- Sunucu Taraflı Denetim



Girdi Denetimi

Sunucu Taraflı – İstemci Taraflı

- ASP.NET örneği

İstemci taraflı aspx dosyasında

```
<asp:RegularExpressionValidator runat="server"
ID="searchValidator"
ControlToValidate="TxTSearch" ErrorMessage="Special characters
are not allowed" ValidationExpression"^[a-zA-Z0-9]*$"/>
```

Sunucu taraflı C# dosyasında

```
using System.Text.RegularExpressions;

Regex regex = new Regex("^[a-zA-Z0-9]*$");
if (regex.Match(queryString).Success)
{
//Do something!
}
else{
//Do something!
}
```

Girdi Denetimi

Parametrik Sorgular (Parameterized Queries)

- SQL Enjeksiyonuna karşı en etkili önlem

String Ekleme

```
String username = Session["username"];
String password = Session[“password"];
String selectCommand = "SELECT * FROM usersWHERE username='" + username + "' AND password='" +
password + "'";
SqlCommand myCommand = new SqlCommand(selectCommand, DataConnection);
SqlDataReader dr = myCommand.ExecuteReader();
```

Parametrik Sorgu

```
String selectCommand = "SELECT * FROM users WHERE username=@username
AND password=@password;";
myCommand.Parameters.AddWithValue("username", Session["username"].ToString());
myCommand.Parameters.AddWithValue("password", Session["password"].ToString());
SqlDataReader dr = myCommand.ExecuteReader();
```

Çıktı Denetimi

Özel Karakterler Kullanılması Gerekiyorsa?

- Kullanıcıya en yakın noktada
- Output Escaping / Encoding
  - URL Encoding (URL kodlama)
    - **ASP/ASP.NET:** Server.URLEncode()
    - **PHP:** urlencode()
  - HTML Encoding (HTML kodlama)
    - **ASP/ASP.NET:** Server.HTMLEncode()
    - **PHP:** htmlspecialchars()

```
& --> &amp;
< --> &lt;
> --> &gt;
" --> &quot;
' --> &#x27;
/ --> &#x2F;
```

Sunum İçeriği

- Genel Bakış
- Bilgi Toplama
- Girdi & Çıktı Denetimi
- Oturum Yönetimi
- Kimlik Doğrulama & Yetkilendirme
- İş Mantığı Problemleri
- Ayar Yönetimi

Neler Öğrendik?

- HTTP Mesajları
- HTTP Trafiğinde Araya Girmek ve Mesajları Manipüle Etmek
- SSL Nedir? Nasıl Çalışır?
- Siteler Arası Betik Çalıştırma (XSS)
- SQL Enjeksiyonu
- Poizitif & Negatif Girdi Denetimi
- İstemci Taraflı & Sunucu Taraflı Denetim
- Çıktı Denetimi

Oturum Yönetimi

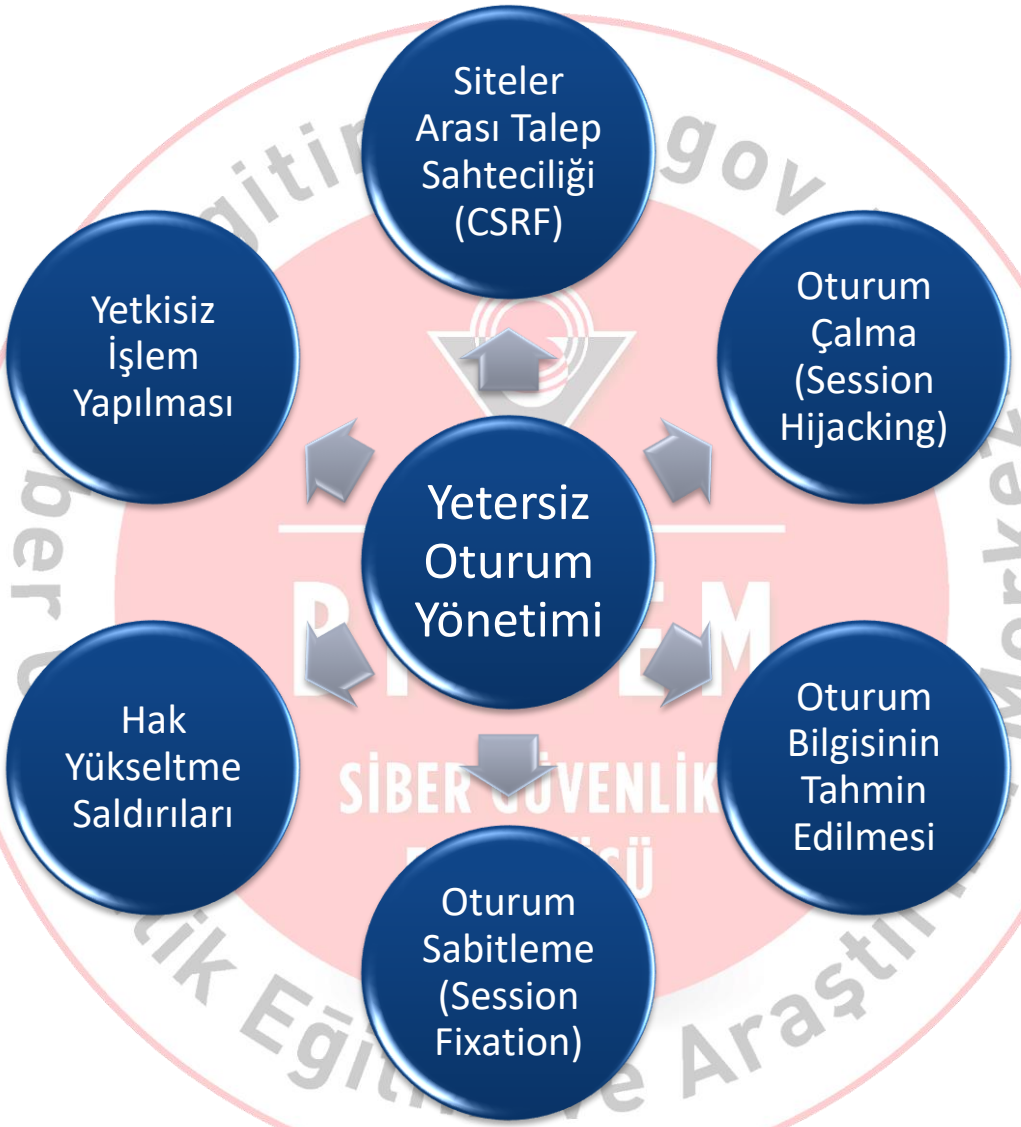
Oturum Yönetimi

Oturum

- HTTP, durum bilgisis (stateless) bir protokoldür.
  - Durum / Oturum bilgisine ihtiyacımız var.
  - Oturum bilgisinin her talepte gönderilmesi gerekli.
- En popüler oturum yönetim metodu Çerez (cookie)'dir.

```
GET /etkinlikler/index.php HTTP/1.1
Host: www.bilgiguvenligi.gov.tr
User-Agent: Mozilla/5.0 (Windows NT 6.1; rv:12.0) Gecko/20100101 Firefox/12.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-us,en;q=0.5
Accept-Encoding: gzip, deflate
Proxy-Connection: keep-alive
Cookie: e23ea89e9ae24e569be891f1e6d862c2=-
```

Oturum Yönetimi



Oturum Sabitleme

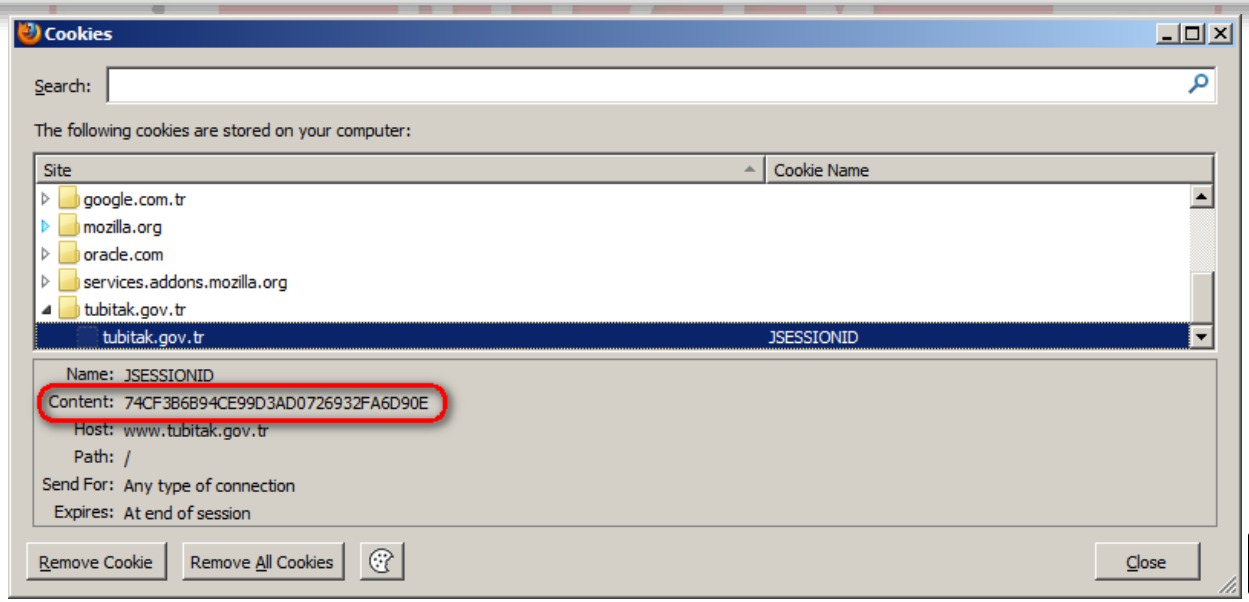
Session Fixation

- Problem: Kullanıcı girişi sonrası oturum bilgisinin değiştirilmemesi

Siteler Arası Talep Sahteciliği

Cross Site Request Forgery

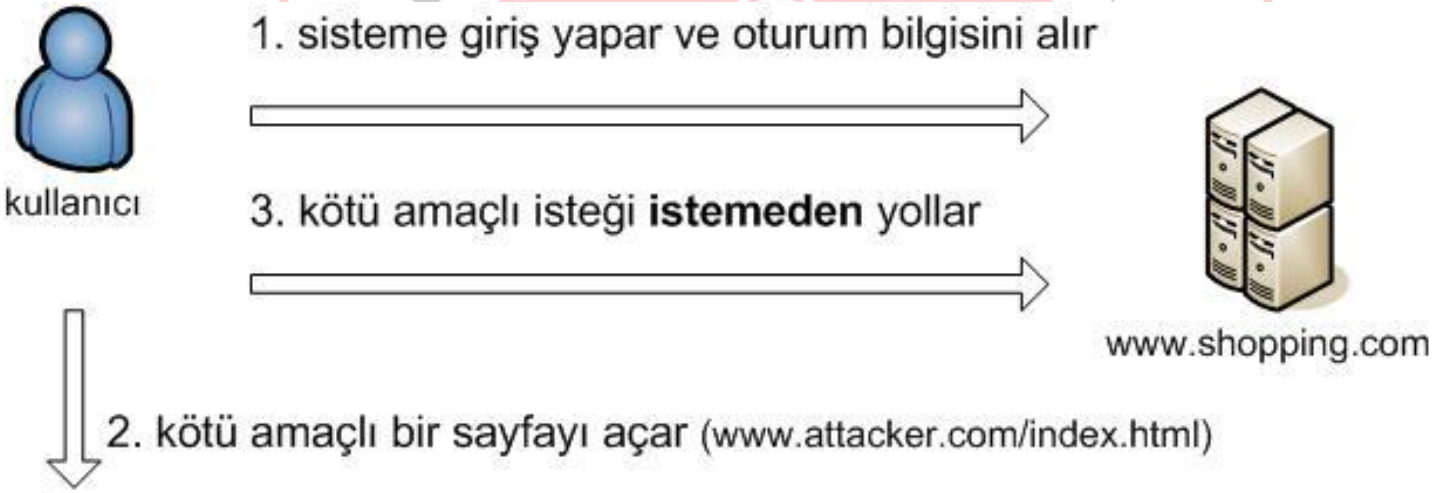
- Kullanıcıyı, giriş yapmış olduğu bir uygulamada, istenmeyen işlemleri gerçekleştirmeye zorlayan saldırı.
- Otomatik açıklık tarayıcılar (vulnerability scanner) ile tespiti zordur.



Siteler Arası Talep Sahteciliği

Cross Site Request Forgery

- Şaşkın Vekil (Confused Deputy) Problemi: Tarayıcı, talebin yetkili olup olmadığını bilemez.




```
<html>
 ...
 <img src="www.shopping.com/admin?cmd=deleteAll" width=0 height=0>
 ...
</html>
```

Siteler Arası Talep Sahteciliği

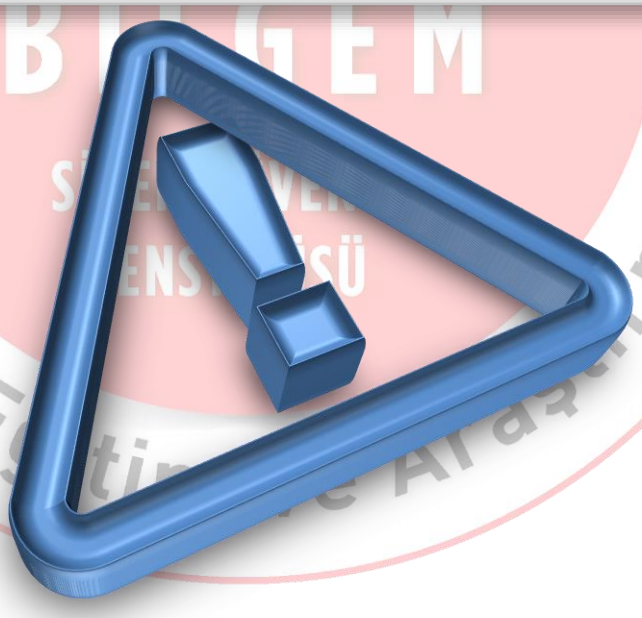
Cross Site Request Forgery

- Önlemler

Siteler Arası Talep Sahteciliği

Cross Site Request Forgery

- URL içinde tekil rastgele tanımlayıcılar
  - /servlet/f81d4fae-7dec/ManageUser
- URL parametreleri içinde saklanan tekil tanımlayıcılar
  - /servlet/ManageUser?csrfid= f81d4fae-7dec



Siteler Arası Talep Sahteciliği

Cross Site Request Forgery

- Gizli Rastgele Parametre

```
<form method=post action="/changeinfo.php">
        <input type=text name=username value=""></td>
        <input type=text name=location value=""></td>
        <input type=text name=newpassword value=""></td>
        <input type=text name=re-password value=""></td>
</form>
```

```
<form method=post action="/changeinfo.php">
        <input type=text name=username value=""></td>
        <input type=text name=location value=""></td>
        <input type=text name=newpassword value=""></td>
        <input type=text name=re-password value=""></td>
        <input type="hidden"
        name="sharedsecret"value="8dcb5e56904d9b7d4bbf333afdd154ca">
</form>
```

Oturum Yönetim Stratejileri

Kimlik doğrulama sonrası oturum bilgisinin yenilenmesi

Çıkış (logout) sonrası oturum bilgisinin sonlandırılması

Çıkış (logout) sonrası oturum bilgisinin geçersiz hale getirilmesi

Belirli bir süre aktif olunmaması durumunda oturumun otomatik sonlandırılması

Çerez içeriğinde HttpOnly ve Secure parametrelerinin kullanılması

Kimlik Doğrulama

Kimlik Doğrulama

Authentication

- Tanım: Kullanıcının kimliğini saptamak
- Yetersiz kimlik doğrulama sonuçları:
  - Yetki artırımı ve yetkisiz işlem
  - Kullanıcı sahteciliği
  - Servis dışı bırakma

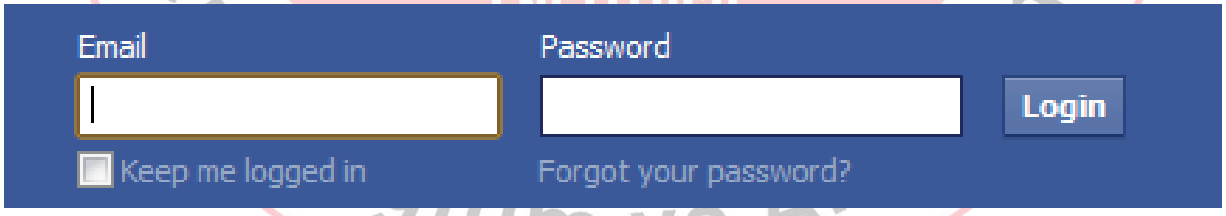
Kimlik Doğrulama: 3 Faktör-3 Metod

3 Faktör

- Kullanıcının Bildiği
- Kullanıcının Sahip Olduğu
- Kullanıcının Kendisi

3 Metod

- Tek Etmenli
- İki Etmenli
- Birçok Etmenli



Kimlik Doğrulama Çeşitleri

TÜBİTAK

Basit (Basic)

- ✓ En fazla desteklenen metoddur (RFC 2617)
- ✓ Base64 kodlama kullanılır
- ✓ Esnek değildir
- ✓ Logout fonksiyonu yoktur

Özet (Digest)

- ✓ Tuzlanmış (salted) MD5 kriptografik özet (hash) kullanılır
- ✓ Parola sunucu tarafında (Apache-htpasswd) açık (clear-text) tutulur
- ✓ Proxy ve güvenlik duvarları ile uyumludur.

Bütünleşik (Integrated) Windows

- ✓ Sadece Windows sunucu ve istemcileri
- ✓ Parola sunucu tarafında açık tutulmaz
- ✓ NTLM veya Kerberos
- ✓ Proxy ve güvenlik duvarları ile uyumlu değildir

Sertifika Tabanlı (Certificate Based)

- ✓ İki taraflı kimlik doğrulama mümkündür
- ✓ Geçerli sertifikaya ihtiyaç vardır.
- ✓ Sertifikanın korunması gerekmektedir.

Form Tabanlı (Form Based)

- ✓ Özel yapım çözüm
- ✓ Esnek yapılıdır
- ✓ Proxy ve güvenlik duvarları ile uyumludur
- ✓ Çok kullanıcılı uygulamalarda tercih edilir

Kimlik Doğrulama Stratejileri

- İstemci – Sunucu taraflı kimlik doğrulama
- Varsayılan kullanıcı değerleri
- Kullanıcı adı – parola politikaları
- SSL ile koruma
- Şifre kurtarma operasyonları
- Kimlik doğrulamada kayıt tutma
- İstemcinin sertifika ile doğrulanması

Kimlik Doğrulama

Form Tabanlı Kimlik Doğrulama

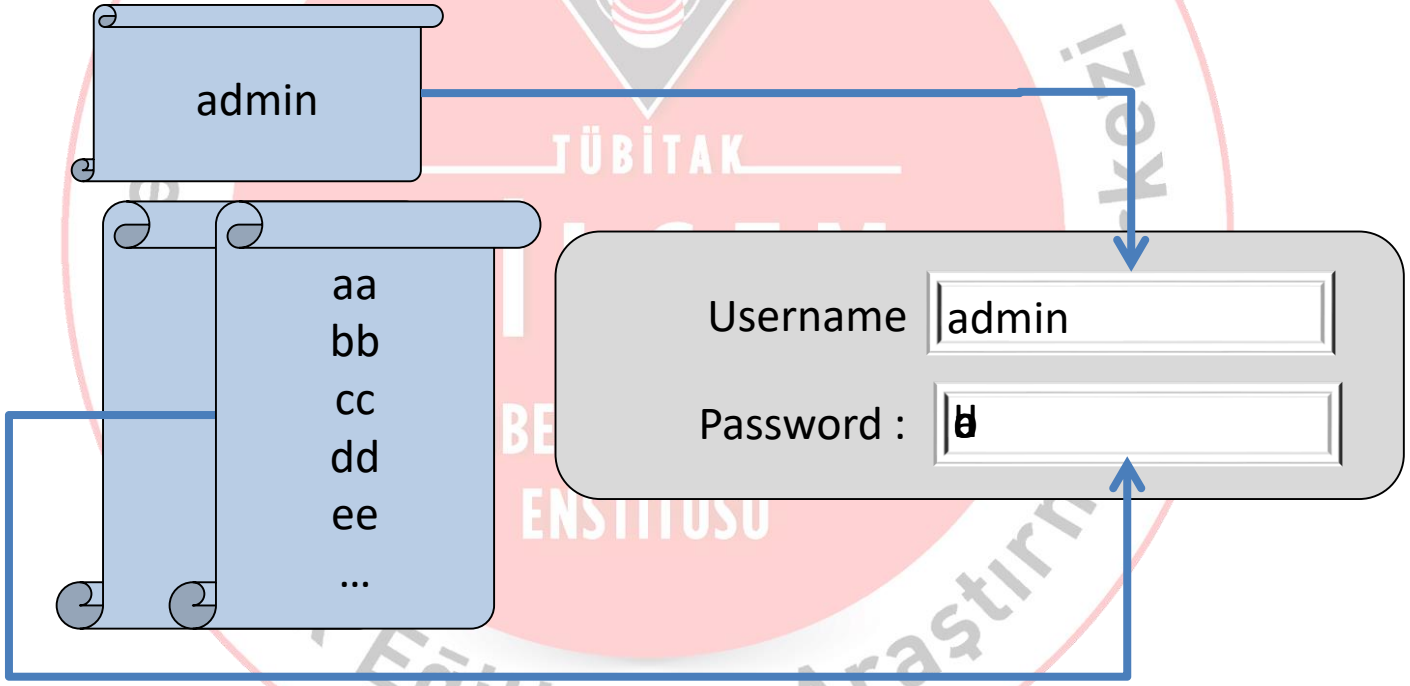
- Kimlik doğrulama sonrası yanıt
  - 200 OK ✗
  - 302 Found ✓
- Hassas bilgiler ve istemci depoları (cache)


```
Expires: -1
Cache-Control: no-cache, no-store
Pragma: no-cache
```

```
<META HTTP-EQUIV="Cache-Control" CONTENT="no-cache,no-store">
<META HTTP-EQUIV="Pragma" CONTENT="no-cache">
<META HTTP-EQUIV="Expires" CONTENT="-1">
```

Kaba Kuvvet (Brute Force) Saldırıları

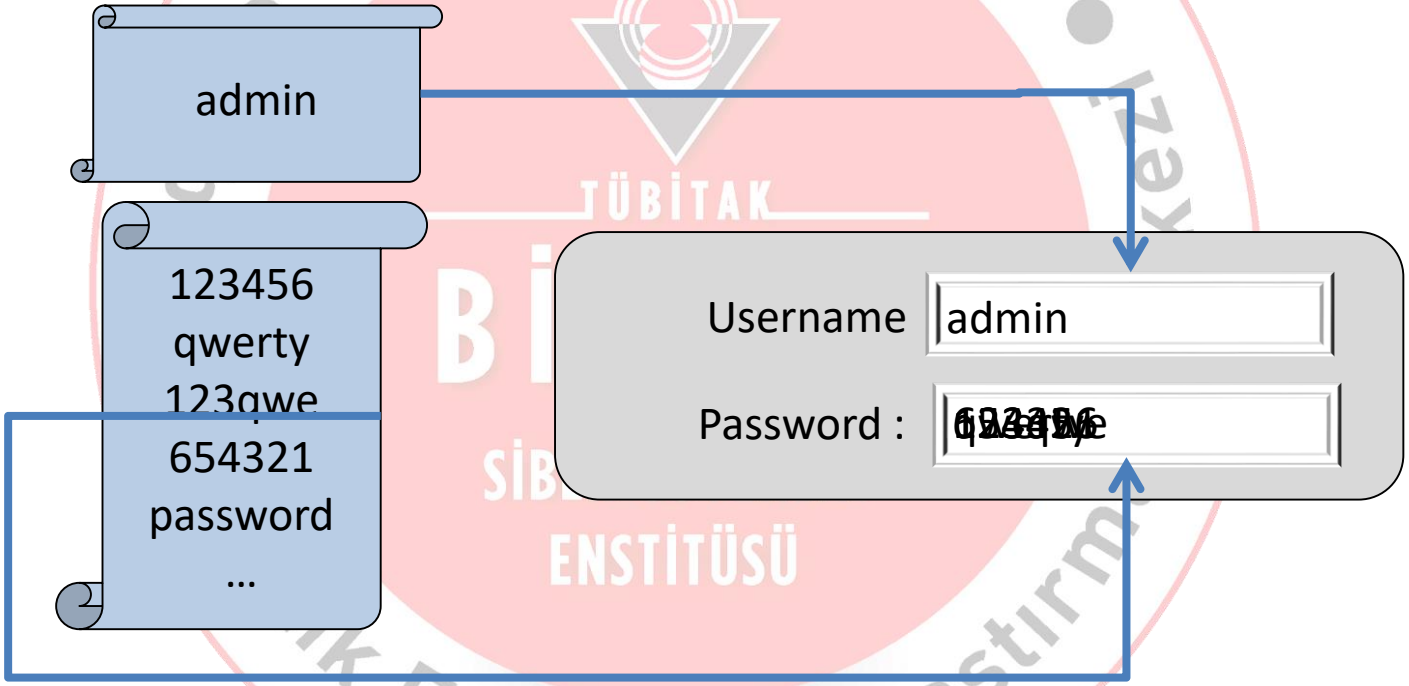
- **Tanım:** Mümkün olan tüm ihtimallerin denenmesi.



Kimlik Doğrulama

Sözlük (Dictionary) Saldırıları

- **Tanım:** Daha önce hazırlanmış sözlükteki anahtar kelimelerin denenmesi



Kimlik Doğrulama

CAPTCHA

- **C**ompletely **A**utomated **P**ublic **T**uring test to tell **C**omputers and **H**umans **A**part
- Bilgisayar ve insanın ayırt edilmesi
- Kaba Kuvvet ve Sözlük saldırılarına karşı etkili

Kimlik Doğrulama

OTP (One Time Password)

- Tek kullanımlık şifre üretimi
- Kimlik doğrulamayı bir faktör artırır.
- Kazıma kartları, SMS mesajları, Token'lar

Kimlik Doğrulama

Tümleşik Oturum (Single Sign-On)

- Alt etki alanları (subdomains)
  - sunucu1.sirket.com.tr, sunucu2.sirket.com.tr
- Farklı etki alanları
  - sirket-bir.com.tr, sirket-iki.com.tr
  - Federation
  - SAML, OpenID, WS-Trust, WS-Federation, OAuth
- Merkezi yetkilendirme
  - Dizin sunucusu, Veritabanı
- CAS, OpenSSO, Oracle IDM, CA
- Farklı protokoller ve bileşenler de dahil olabilir
  - Genelde *Kerberos* üzerinden
  - Bütünleşik Windows -> İşletim sistemi + uygulama
  - Radius + Kerberos + IEEE 802.1x + işletim sistemi + uygulama

Kimlik Doğrulama

Bileşenler Arası Kimlik Doğrulama

- Uygulama sunucular arası etkileşim
  - Yük dengeli çalışan sunucular arası kimlik doğrulama
  - HTTP sunucular (IIS, Apache v.s) ile uygulama sunucuları (BizTalk, WebLogic, Managed Services Engine (MSE), Reports Server) arası kimlik doğrulama
- Uygulama sunucu ve HTTP sunucu ile üçüncü katman sunucular (Dizin servisleri, Veritabanı) arası kimlik doğrulama
- Kullanıcı kimlik bilgilerinin bileşenler arası aktarılması
  - Delegation, Impersonation, Protocol Transition

Yetkilendirme

Yetkilendirme

Authorization

- Kimlik doğrulamış kullanıcıların erişim ve haklarının belirlenmesi
- Yetkilendirme çeşitleri:
  - IP veya sunucu adı tabanlı
  - URL tabanlı
  - Uygulama tabanlı
- Kontrollü gezinim. (Forced Browsing)
  - HTTP Referer başlığı
  - Tuzlanmış URL kriptografik özeti
- Yetkilendirme sunucu tarafında yapılmalıdır.

ActiveX, Flash, Silverlight, Java Applet

Yetkilendirme

Stratejiler (1)

- Platform tarafından sunulan hazır yetkilendirme yöntemlerinin kullanımı
  - JAAS, Apache Shiro
  - .NET Framework: role-based security
- Uygulama sunucularının güvenli kurulum değerlerine sahip olması
  - Kurulumda güvenli varsayılan değerler ve koruma
  - Minimum hak prensibi

İşletim Sistemi	Uygulama Sunucusu	Koşan İş	Koşan Kullanıcı
Windows	IIS/6/7 ve ASP.NET	W3wp.exe	NETWORK SERVICE / Service Principal
Unix	Apache ve PHP	Httpd /apache	apache
Windows	IBM Websphere	java.exe	Local System / Service Principal

Yetkilendirme

Stratejiler (2)

- Üçüncü katman sunucu bağlantıları
  - Veritabanına tek kullanıcı ya da çok kullanıcı ile bağlantı
    - Tüm katmanlardaki kullanıcıların merkezileştirilmesi
    - Oracle Proxy Authentication
  - Minimum hak prensibi
    - “sa”, “system”, “root” vb. kullanılmamalı
    - LDAP: anonim kullanıcılar
    - Veritabanı: sql depo prosedürleri

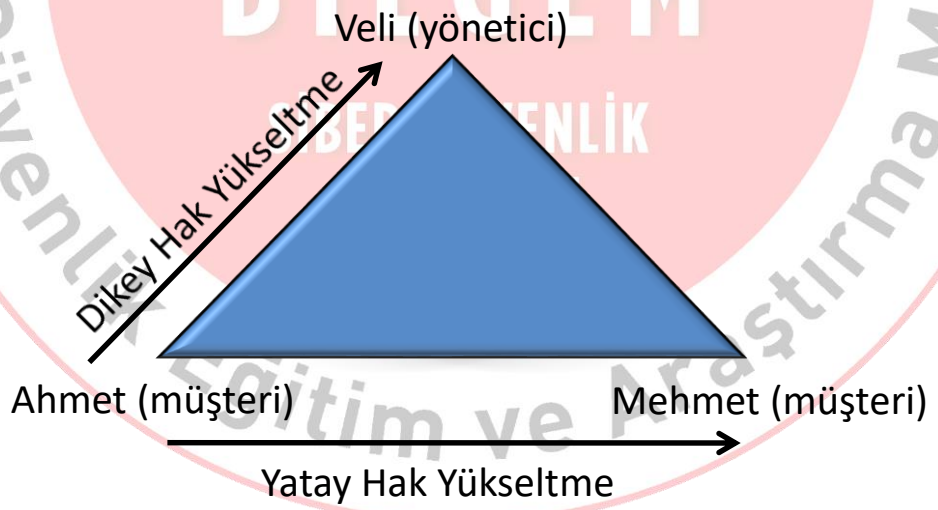
Stratejiler (3)

- Geliştirme Ortamları
  - Geliştirme ve üretimde minimum hak prensibi
- Otomatik web tarayıcıları
  - Dosya temizliği
  - robot.txt
- Kaba kuvvet saldırıları
  - Kaynak kontrolü metotları (anti-resource metering)
    - Cevaplama zamanı ayarlama, CAPTCHA

Yetkilendirme

Hak Yükseltme Saldırıları

- Kullanıcının yetkisi dışındaki haklara erişimi
- Türleri
  - Yatay hak yükseltme
  - Dikey hak yükseltme



Yetkilendirme

Yatay Hak Yükseltme & Dikey Hak Yükseltme

- Örnekler

http://message.mysite.com/index.cfm?fuseaction=read&messageID=500

```
<form method="POST">
action="http://site.com/mailing_list.pl">
    ...
    <input type="hidden" name="login_name" value="aUser">
    <input type="hidden" name="list"
value="FREQUENT_FLYER">
    ...
    <input type="hidden" name="list_admin" value="F">
    ...
</form>
```

Sunum İçeriği

- Genel Bakış
- Bilgi Toplama
- Girdi & Çıktı Denetimi
- Oturum Yönetimi
- Kimlik Doğrulama & Yetkilendirme
- İş Mantığı Problemleri
- Ayar Yönetimi

Neler Öğrendik?

- Neden Oturum Yönetimi?
- Oturum Sabitleme (Session Fixation)
- Siteler Arası Talep Sahteciliği (CSRF)
- Kimlik Doğrulama Metodları
- Kimlik Doğrulama Çeşitleri
- Kimlik Doğrulama Stratejileri
- Yetkilendirme Stratejileri
- Hak Yükseltme Saldırıları

İş Mantığı Problemleri

İş Mantığı Problemleri

Business Logic

- Web Application Vulnerability Scanner (Web Uygulama Açıklık Tarayıcıları) tespit edemez.
- IDS / IPS cihazları yakalayamaz.
- Web Application Firewalls (Uygulama Güvenlik Duvarları) tespit edemez.

```
action=update&id=1
```

```
action=delete&id=1
```

İş Mantığı Problemleri

Gerçek Hayattan Bir Örnek

- Bir web uygulaması
  - Sadece **Kullanıcı Bilgisi Gör** var; **Kullanıcı Bilgisi Güncelle** yok.
  - Kullanıcılar kendi kişisel bilgilerini değiştiremiyor!
  - Kullanıcı bilgisi güncelleme yetkisi sadece **admin** kullanıcılarda

```
<form name ="UpdateCustomerInfo" border="0" width="600"
method="POST" action="customerinfoend.aspx“>
```

```
<script>
        function init(){
        /*
        if (document.forms[0].txtCellPhone1){
           document.forms[0].txtCellPhone1.INTEGER=true;
           document.forms[0].txtCellPhone2.INTEGER=true;
           document.forms[0].txtCellPhone3.INTEGER=true;
```

İş Mantığı Problemleri

Gerçek Hayattan Bir Örnek

- Sahte Form alanlarının oluşturulması

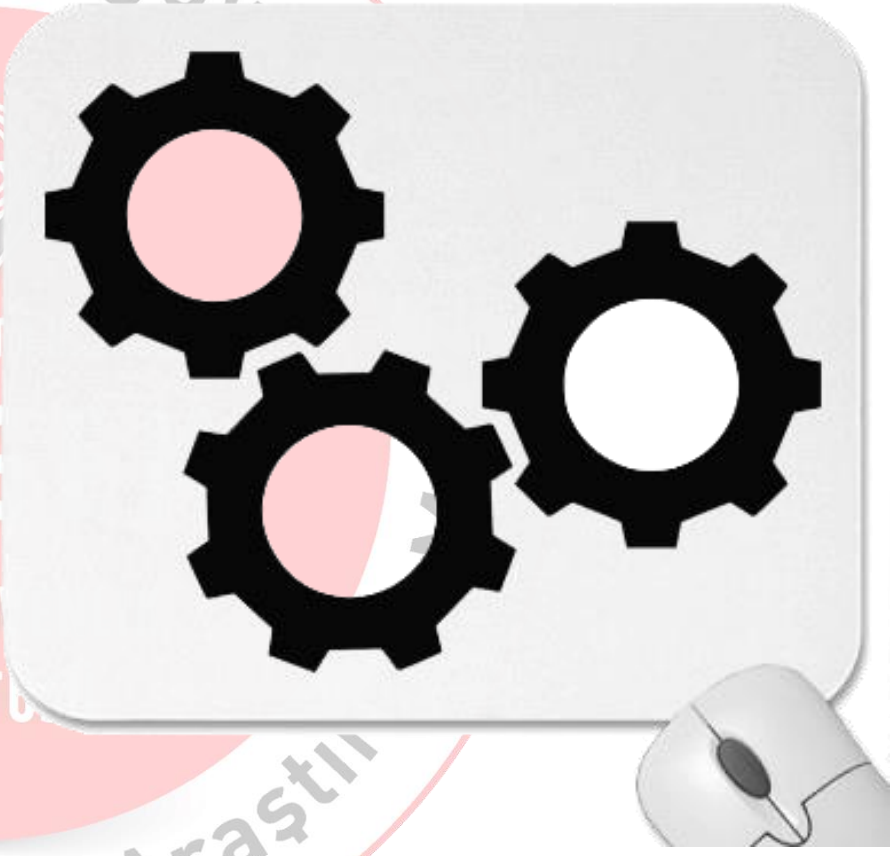
```
<html>
     <form name ="UpdateCustomerInfo" border="0" width="600"
method="POST" action="customerinfoend.apsx">
       <input type=text name="txtCellPhone1" value="90">
       <input type=text name="txtCellPhone2" value="444">
       <input type=text name="txtCellPhone3" value="4444444">
      </form>
      <script type="text/javascript">
       document.UpdateCustomerInfo.submit();
      </script>
</html>
```

Ayar Yönetimi

Ayar Yönetimi

İçerik

- Minimum Bilgi Prensibi
- Bağlantı Güvenliği: SSL / TLS
- HTTP Metodları Denetimi
- Test Ortamları



Minimum Bilgi Prensibi

Need To Know Principle

- Mesajlar, Hatalar, Dokümanlar, Sayfalar

Geçersiz parola, lütfen tekrar deneyiniz!

Kullanıcı adınızı kontrol ediniz!

Parolanız 6 karakter uzunluğunda olmalıdır

Geçersiz kullanıcı adı veya parola!

```
Content-Type = text/html;charset=ISO-8859-1
Keep-Alive = timeout=5, max=100
Content-Length = 693
Connection = Keep-Alive
Server = Apache/2.2.22 (Unix) mod_ssl/2.2.22 OpenSSL/0.9.8r DAV/2 PHP/5.4.4
Date = Mon, 13 May 2013 12:30:38 GMT
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 3.2 Final//EN">
<html>
 <head>
```

Bağlantı Güvenliği SSL/TLS

SSL Parametreleri

- Protokol Versiyonları:

  SSL 1.0, 2.0 ve 3.0, TLS 1.0 , 1.1 ve TLS 1.2

- Şifre Demetleri (Cipher Suites)

  Anahtar değişim (Key exchange)
  Kimlik doğrulama (Authentication)
  Simetrik şifreleme (Symmetric encryption)
  Bütünlük koruma (integrity protection)

SSL_RSA_WITH_RC4_128_MD5

Bağlantı Güvenliği SSL/TLS

Zayıf Kabul Edilen SSL Parametreleri

- Kullanılan Protokol
  - SSL v2 ve öncesi
- Anahtar Değişim Fazı
  - Anonymous Diffie-Hellman, EXPORT RSA
- Simetrik Şifreleme
  - NULL şifreleme, 56 bitten küçük şifreleme algoritmaları
- Bütünlük Koruma
  - MD2, MD4, MD5 (?)

Bağlantı Güvenliği SSL/TLS

Apache SSL Yapılandırması

- SSLCipherSuite direktifi kullanılarak yapılır.
- Zayıf Yapılandırma:

```
SSLCipherSuite RC4-MD5:DES-CBC3-SHA:EXP-RC4-MD5
```

- Kuvvetli Yapılandırma:

```
!ADH:HIGH:MEDIUM:!SSLv2
```

- En Kuvvetli Yapılandırma:

```
HIGH:!aNULL:!MD5
```

IIS SSL Yapılandırması

- Şifre demeti sıkılaştırmaları için:

```
HKLM\System\CurrentControlSet\Control\SecurityProviders\SCHANNEL\Ciphers
```

- Protokol versiyonu sıkılaştırmaları için:

```
HKLM\System\CurrentControlSet\Control\SecurityProviders\SCHANNEL\Protocols
```

Bağlantı Güvenliği SSL/TLS

İstemci Yapılandırmaları

- Internet Explorer
  - Registry ayarları (regedit)
- Firefox
  - About:config
- Opera
  - Preferences-Advanced-Security-Security Protocols
- Chrome
  - Options-Under the Hood-Security

Bağlantı Güvenliği SSL/TLS

SSL Güvenlik Kontrol Listesi

1	Sunucu İsmi	▪ Sertifikadaki “Issued to” alanında yazan alan adı ile sitenin alan adının aynı olması gereklidir
2	Sertifika Makamı	▪ Sertifikadaki “Issued by” alanında yazan makamın güvenilir olması gereklidir.
3	Geçerlilik Süresi	▪ Sertifikaların geçerlilik sürelerinin dolmadığını kontrol edilmelidir.
4	SSL/TLS Versiyon	▪ SSL/TLS protokollerinin en yeni sürümü kullanılmalıdır.
5	Şifre Demetleri	▪ Kuvvetli şifre demetleri kullanılmalıdır (kimlik doğrulama, anahtar değişimi, şifreleme, bütünlük)

Bağlantı Güvenliği

Giriş Sayfalarının Güvenliği

- Kullanıcı girişi sayfalarında mutlaka SSL/TLS kullanılmalı.
- SSL'li hizmet verilen uygulamalarda aynı zamanda SSL'siz hizmet verilmemeli.

https://

HTTP Metodları Denetimi

Diğer HTTP Metodları

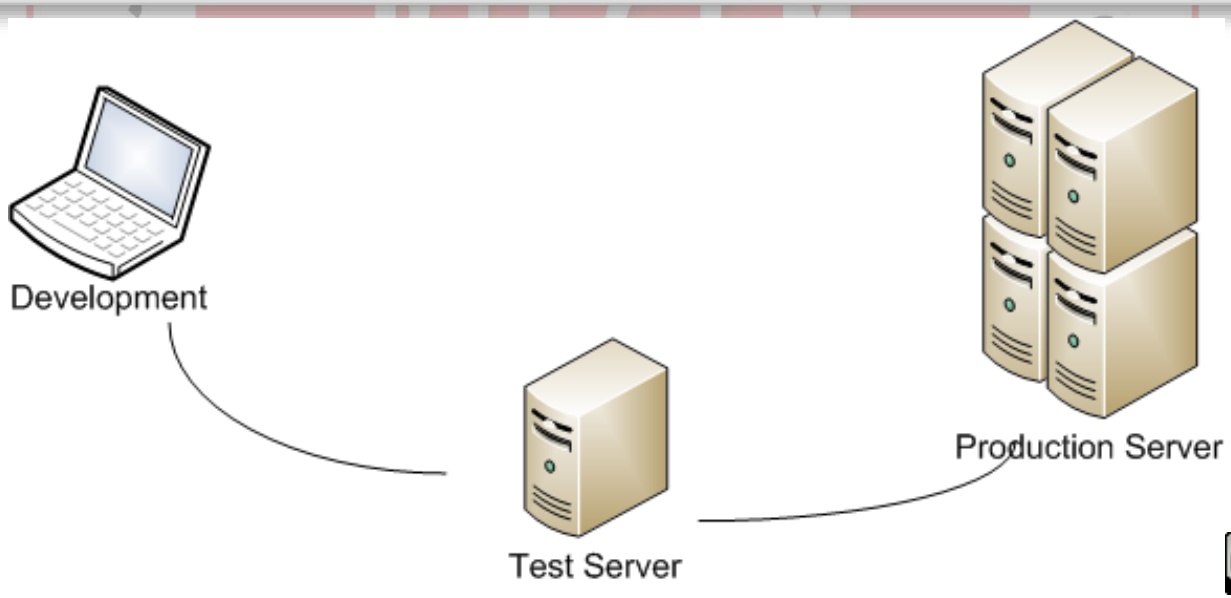
- HEAD: GET gibi çalışır fakat istek karşılığında gövde kısmını (body) döndürmez. Genelde cevap içinde bulunan meta bilgilerini okumaya yarar.
- PUT: Belirtilen kaynağı yükler.
- DELETE: Belirtilen kaynağı siler.
- OPTIONS: İzin verilen HTTP metotlarını listeler.
- CONNECT: Web sunucusunu bir proxy gibi kullanmaya yarar.
- TRACE: Hata ayıklamada kullanılır.

```
Server = Apache/2.2.22 (Unix) mod_ssl/2.2.22 OpenSSL/0.9.8r DAV/2 PHP/5.4.4
Content-Type = text/html; charset=iso-8859-1
Allow = GET,HEAD,POST,OPTIONS,TRACE
Date = Sun, 26 May 2013 12:27:00 GMT
Keep-Alive = timeout=5, max=100
Content-Length = 221
Connection = Keep-Alive
<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<html><head>
<title>405 Method Not Allowed</title>
</head><body>
<h1>Method Not Allowed</h1>
<p>The requested method PUT is not allowed for the URL /.</p>
</body></html>
```

Test Ortamları

Test Ortamı Güvenliği

- Veri gizliliğinin korunması
  - Data Masking
  - Ayar dosyalarında bulunan bağlantı bilgileri
  - Şifrelemede kullanılan anahtarların taşınması
- Test ve canlı ortam güvenlik ayarları senkronizasyonu
  - işletim sistemi, uygulama sunucusu, dizin sunucusu ve uygulama
- Güvenli Clone alma sürecinin dokümantasyonu



Web Güvenliği

TÜBİTAK

Teşekkürler