**WEB MİMARİ YAPISI**

**Frontend**

Web uygulamalarında kullanıcıların görebildiği ve doğrudan etkileşimde olduğu kısımdır. Web arayüzü bu kısıma bir örnektir.

Diller: Html, CSS, Javascript, JQuery

**İstemci(Client):** Sunuculardan ağ üzerinden hizmet alan uygulamalardır.

**Backend**

Web uygulamalarında kullanıcıların göremediği bölüm olup istemcinin isteklerinin gerçekleştirildiği kısımdır.

Diller: Php, Java, Phyton, SQL

**Sunucu(Server):** İstemcinin bilgi ve verilere ulaşmasını sağlayan sistemdir. **Veritabanı(Database):** Bilgilerin ve verilerin depolandığı alandır. Sunucudan bağımsız kullanılabilirler.



**HTTP (Hyper Text Transfer Protocol)**

HTTP’nin açılımı Hypertext Transfer Protocol(Hiper Metin Transfer Protokolü) dür. Web sayfalarının görüntülenmesini ve veri iletimini sağlayan protokoldür. Bir web tarayıcısına istekte bulunduğumuzda işlemler http protokolüne(kuralına) göre gerçekleştirilir. Http 80 portunu kullanır.

**HTTP METOTLARI:**

**GET:**Belirtilen bir kaynaktan veriyi ister. Yalnızca veri almak için kullanılır herhangi bir değişim yapılmaz. Önbelleğe alınabilir, tarayıcı geçmişinde kalır, verilerin uzunlukları kısıtlıdır. Veriler URL’de herkes tarafından görüntülenebilir. Hassas verilerde kullanılmamalıdır.

**POST:**Belirtilen bir kaynağa veri göndermek veya güncellemek için kullanılır. Önbelleğe alınmaz, tarayıcı geçmişinde kalmaz, verilerin uzunluğu kısıtlı değildir. Veriler URL’de görüntülenmez. Hassas verilerde kullanılır.

**HEAD:** GET ile neredeyse aynıdır fakat sadece head’deki verileri döndürür.

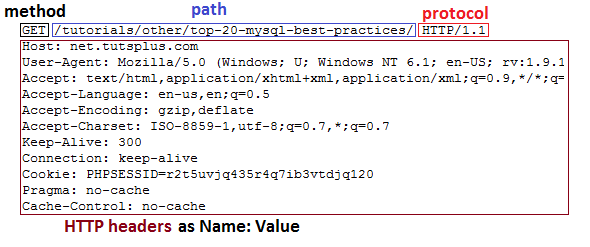
**PUT:** Veri güncellemek için kullanılır.

**TRACE:**Sunucuya yapılan isteğin aynı şekilde iletilip iletilmediğinin kontrolü için kullanılır.

**DELETE:**Belirtilen kaynaktaki veriyi siler.

**OPTIONS:**Hedef kaynak için iletişim seçeneklerini tanımlar.

**HTTP REQUEST**



**Host:** Sunucuda istek yapılan site

**User-Agent:** Tarayıcı hakkındaki bilgileri gösterir

**Accept:** İstemcinin kabul edebileceği nesne türleri

**Accept-Language:** Sitenin kullandığı dil

**Accept-Encoding:** Kodlama algoritması(genellikle sıkıştırma algoritması) hakkında bilgi verir

**Accept-Charset:** Kabul edilebilir karakter setleri

**Keep-Alive:** Kalıcı bir bağlantının ne kadar süre açık kalacağı bilgisi

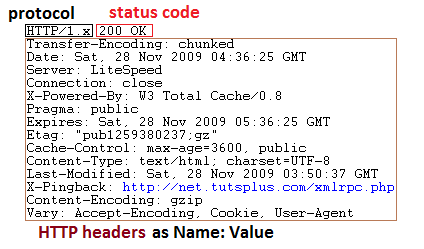
**Connection:** İşlem bittikten sonra bağlantının açık kalıp kalmayacağını denetler

**Cookie:** Önceden Set-Cookie ile gönderilen http tanımlama bilgisi

**Pragma:** HTTP/1.0 kullanıcıları için (cache-control bulunmadığından) önbellek kontrolü yapar.

**Cache Control:** Önbellek kontrolü

**HTTP RESPONSE**



**Transfer-Encoding:** Kaynağı kullanıcıya güvenli bir şekilde ulaştırmak için kullanılan kodlama biçimi

**Date:** Sunucu tarih-zaman bilgisi

**Server:** Sunucu bilgisi

**Connection:** İşlem bittikten sonra bağlantının açık kalıp kalmayacağını denetler

**X-Powered-By:** Sitenin yazıldığı dil

**Pragma:**

**Expires:** Yanıtın geçerli olduğu tarih-zaman bilgisi

**Cache-Control:** Önbellek kontrolü

**Content-Type:** Sayfanın uzantı bilgisi ve dil formatı

**Last-Modified:** İçeriğinsunucuda son değiştirilme tarih-zaman bilgisi

**X-Pingback:**

**Content-Encoding:**

**Vary:**

**Http Durum Kodları:**

**1xx: Bilgi Mesajları**

100: Continue(Devam)

**2xx: Başarılı**

200: OK(İstek Başarılı)

202: Accepted(Kabul Edildi) – Talep alındı ancak işleme konulmadı

**3xx: Yönlendirme**

300: Multiple Choices(Çoklu Seçenek) – Kullanıcı bir link seçebilir

301: Moved Permanently(Kalıcı olarak taşındı) – Kaynağın URL değişti

**4xx: İstemci Hatası**

400: Yanlış Yapılandırılmış URL İsteği

403: Forbidden(Yasaklandı)

404: Not Found(Bulunamadı)

**5xx: Sunucu Hatası**

500: Internal Server Error(Dahili Sunucu Hatası)

502: Bad Gateway – Sunucular arası iletişim zayıflığı

503: Server Unavailable(Sunucuya Erişilemiyor)

504: Gateway Timeout – Sunucu başka bir sunucudan zamanında yanıt

alamadı

**HTTP ve HTTPS Arasındaki Fark Nedir?**

Https'nin açılımı Hypertext Transfer Protocol Secure(Güvenli Hiper Metin Transferi) dir. Http ile belli bir oranda güvenli veri aktarımı sağlanmaktadır. Https ile http’ye şifreleme (RSA şifrelenmesi) eklenir ve daha güvenli hale gelir. Https bağlantısı için Google tarafından sağlanan SSL sertifikası alınması gereklidir. Https 443 portunu kullanır.

**COOKİES(ÇEREZLER)**

Bir web sitesi ziyaret edildiğinde depolanan metin dosyalarıdır. İstemci ile sunucu arasındaki etkileşimi daha güvenli hale getirirler.

SetCookie: <cookiename>=<cookievalue>; Domain=<domainvalue>; Path=<path-value> Expires=<date>; Max-Age=<non-zero-digit>; Secure; HttpOnly

**Cookie Başlıkları:**

**Name:** Çereze verilen isimdir. Opsiyonel olmayan tek çerez özelliğidir. Çerez oluşturulduğunda mutlaka bu özellik için bir değer verilmelidir.

**Domain:** Çerezin etki alanını gösterir.

**Path**: Çerezin hangi yollara yapılan isteklerde gönderileceğini belirlemek için kullanılır.

**Secure:** Çerezin yalnızca https bağlantılarında kullanılması istendiğinde bu özellik kullanılır.

**HttpOnly:** Javascript kodunile çereze erişimi engellemek için kullanılan özelliktir.

**Expires:** Çerezin tarayıcıda tutulacağı süredir. Bir çerezi silme istediğimizde bu seçeneği değiştererek geçmiş bir tarih olarak güncelleyip silebiliriz.

**Max-Age:** Expires özelliğiyle neredeyse aynıdır. Aralarındaki fark ise max-age özelliğinde saniye de yazılabilmesidir.

Expires ve Max-Age özelliklerinin her ikisi de bir çerezde kullanılırsa max-age baz alınır.

**EN ÇOK GÖRÜLEN SALDIRI YÖNTEMLERİ**

**SQL İnjection Attack:** Veritabanını yönetmek için kullanılan SQL dilinden kaynaklanan açıklardan yararlanılarak yapılan ataklardır. Genellikle bu dile ait meta-karakterler kullanılarak atak gerçekleştirilir. Saldırgan veritabanı sucusunu istediği gibi yönetebilir, veri çalabilir.

**Cross-site Scripting (XSS) Attack:**

Kullanıcılardan alınan verilerin kontrol edilmemesi sonucu oluşan zaafiyettir.

REFLECTED XSS: Bu zafiyeti kullanarak kullanıcının girdilerini ekranda çıkartabilmesi durumudur.

DOM(Document Object Model) XSS:

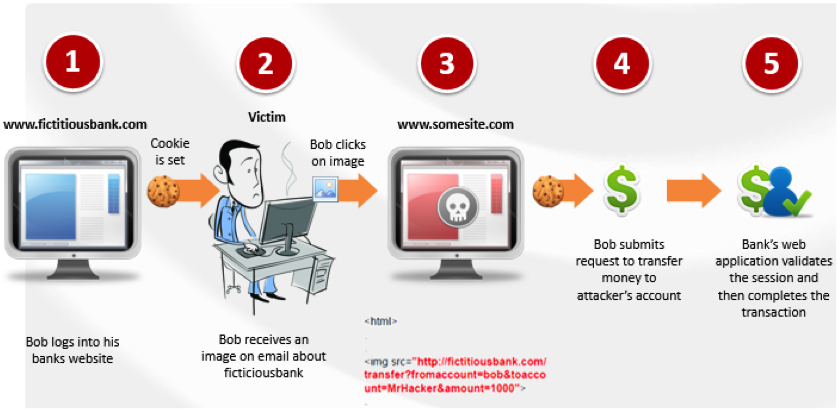
Document internet sayfası, Object sayfa içindekiler(nesneler)

Bu zafiyet kullanılarak sitedeki nesnelerde değişiklik yapılabilir veya yeni nesneler ekleyebiliriz.

STORED XSS: Saldırgan tarafında kod siteye enjekte edilir ve sonrasında veritabanına kaydedilir. Bu kod herhangi bir kullanıcı siteyi kullanırken yürütülür ve böylece kullanıcı bilgileri çalınabilir. Genellikle forum ve yorum alanları ile gerçekleştirilir.

Bu saldırıdan korunmak için kullanıcılardan alınan verileri veritabanına kaydetmeden script kodu olup olmadığının kontrol edilmesi gerekir.

**Cross-site Request Forgery(CSRF):** Bir siteye kullanıcının saldırganın kullanıcı üzerinden siteye erişmesini sağlayan link, resim vb. açmasıyla gerçekleştirilen ataklardır.

****

Resim:<https://cdn1.marklogic.com/wp-content/uploads/2017/08/csrf.png>

**Broken Authentication:** Giriş bölümünün çeşitli yöntemlerle atlatılarak sisteme giriş yapılmasıdır.

Credential stuffing: Herhangi bir siteden ele geçirilen kullanıcı bilgilerinin başka bir sisteme giriş için kullanılmasıdır.

Automated attacks: Rastege kullanıcı bilgileri oluşturularak sisteme giriş için denenmesidir.

Default passwords: Bazı sistemler için varsayılan kullanıcı bilgileri kullanılarak sisteme giriş yaparken kullanılmasıdır. (ör: admin-admin, root-toor)

Çoklu doğrulama, karmaşık şifre kullanma, giriş için şifre denemesini sınırlama gibi yöntemlerle bu saldırı yöntemine karşı güvenliğimizi arttırabiliriz.

**Broken Access Controls:** Sunucu yapılandırmadaki hatalardan dolayı sınırlı yetkisi olan bir roldeki kullanıcının admin yetkisi kazanarak veya erişim denetimindeki zayıflıklardan yararlanarak istediği kullanıcının bilgilerine ulaşmasıdır.

Sunucuyu test etmek, varsayılan erişime izin vermemek ve rol tabanlı kimlik denetimi yapmak bu atağa karşı alınabilecek önlemlerdendir.

**Information Leakage:** Sunucunun verdiği bilgilere göre sistemdeki hassas bilgilerin elde edilmesiyle gerçekleştirilen ataklardır.

**Referanslar**

[1] <https://youtu.be/AYHE2m651dY>

[2] <https://erhankilic.org/post/http-dersleri-sozluk/>

[3] <https://www.netsparker.com.tr/blog/web-guvenligi/http-isleyisi-ve-guvenligi-acisindan-cookie-ve-session-yonetimi/>

[4]<https://www.youtube.com/watch?v=rWHvp7rUka8&list=PLyqga7AXMtPPuibxp1N0TdyDrKwP9H_jD>