

API Security

The background of the slide is a deep purple space scene. It features a large, dark purple planet with a thin, glowing blue ring in the upper left. The sky is filled with numerous small, bright white stars and soft, ethereal purple nebulae, creating a cosmic and mysterious atmosphere.

İçerik

00 Whoami

01 Geçmişten Günümüze

02 API Nedir ?

03 TOP 10 API Zafiyeti

İçerik

04 Uygulama

05 Hayattan Örnekleri


06 Kapanış

The background is a deep space scene. It features a large, glowing purple nebula on the right side, with wispy, ethereal clouds of gas. In the upper right, a large, dark planet with a thin, glowing purple ring is visible. The entire scene is set against a dark, star-filled sky with numerous small, bright white stars scattered throughout.

00

Whoami


Whoami




Web App. Penetration
Tester



DevSecOps



Code Review / Code
Security

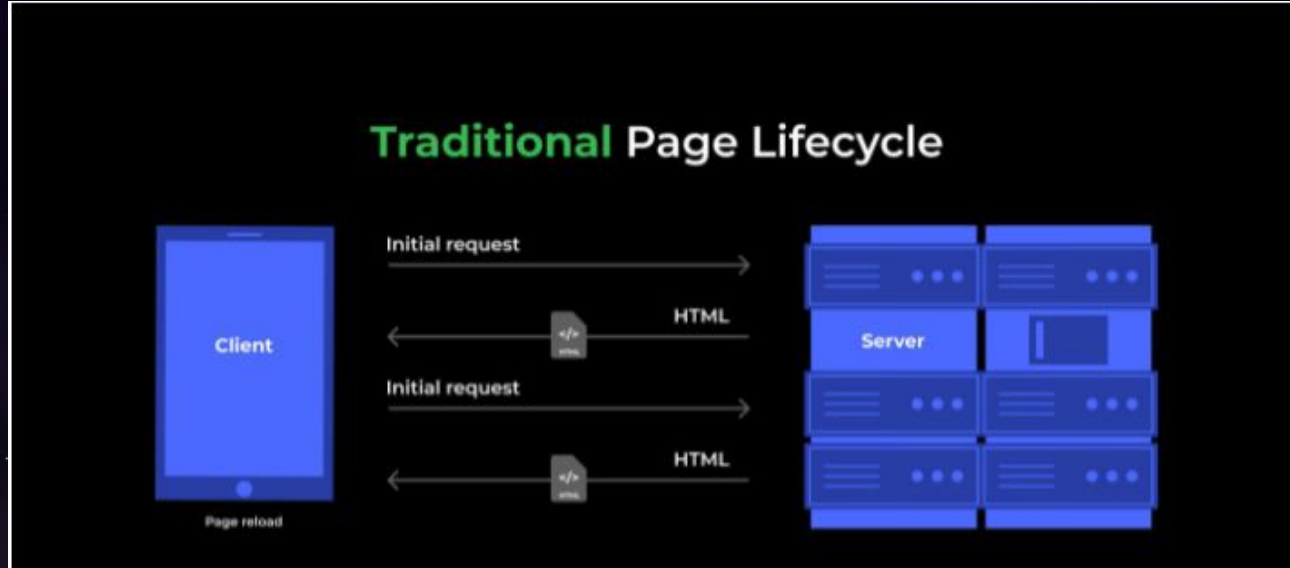


Developer (Part-time)

01 Geçmişten Günümüze

Multiple Page Application (Traditional)

Pekçok işlemin backend tarafında yapıldığı uygulamalardır. Kullanıcı bir sayfa erişmek istediğinde bunun için backend'e istek atar ve belirtilen sayfa cevap içerisinde getirilir. Getirilen sayfa kullanıcı tarayıcısında render edilir. Client tarafındaki herhangi bir işlem için bir istek tetiklenir ve backend'e iletilerek döngü devam eder.



Asynchronous JavaScript and XML (AJAX)

2000'lerin başında MPA'nın performans problemlerini daha uygun hale getirmek için üretildi. AJAX client side tarafta sayfanın tamamının yerine belirli bir kısmınının tekrar yüklenmesini sağlayan bir geliştirmedir. Bu geliştirme backend'e bir istek gönderip cevabın ne zaman geleceğini umursamadan çalışır. Cevap geldiğinde sayfanın ilgili kısmı değişir.



Single Page Application (SPA)

2010'ların başında MPA ve AJAX'ın harmanlanmış hali olarak kurgulandı. SPA de pek çok mantıksal işlemi tarayıcı tarafında gerçekleştirmektedir. Bir ihtiyaç olduğunda Web API aracılığıyla backend'le konuşur. SPA'nın çalışması için gerekli olan kaynaklar (HTML , CSS , Javascript) tarayıcıya bir kerede yüklenir ve çalışma döngüsü boyunca client'ın sayfayı tekrar yükleme ihtiyacı olmaz. İçerikler dynamic olarak değişir. SPA tarafındaki çalışmalar zamanla gelişerek frameworkler üretildi(Angular , Vue, React vs.).



02 API Nedir

API

- Application Programing Interface'in kısaltılmışıdır. Basit bir ifadeyle API, geliştiricilerin bir uygulamanın, işletim sisteminin veya diğer hizmetlerin belirli özelliklerine veya verilerine erişmesine izin veren bir dizi işleve sahip bir yapıdır.
- Başka bir deyişle geliştiricilerin karmaşık işlevleri daha kolay oluşturmalarına olanak sağlamak için programlama dillerinde kullanıma sunulan arayüzdür.
 - Evlerde kullanılan prizler API'ya örnek olarak verilebilir. Bir elektrik ihtiyacı olduğunda kablo evin şaltellerine bağlanmak yerine ev içerisindeki prizler kullanılır.

Web Service / Web API

- Genelde HTTP protokolü kullanılarak client ile backend'in haberleştiği konsept'lerdir.
 - Twitter'ın REST API'leri, twitter'ın yeteneklerini kendi uygulamamıza entegre edebileceğimiz verileri okumak ve yazmak için programlı erişim sağlar.

SOAP

- Simple Object Access Protocol olarak da bilinir. Bir protokol olarak dizayn edilmiştir. Farklı programların veya uygulamaların verilere kolay yoldan ulaşabilmeleri/işlemeleri için üretilmiştir.
- Bazı durumlarda HTTP dışında TCP protokolü üzerinden de hizmet verir.
- Veri alışverişinde XML formatını destekler.
- Oluşturulan istekleri bir araya daha düzenli görmek için WSDL (Web Service Description Language) kullanır.

REST

- Representational State Transfer olarak da bilinir. SOAP alternatif client / server arasındaki iletişimi kolaylaştırmak için geliştirilmiştir.
- REST sadece HTTP/HTTPS protokollerine üzerinden çalışmaktadır.
- REST mimarisine göre tanımlanmış herhangi bir web servis, RESTful olarak adlandırılır. (Restful olabilmesi için GET , POST , PUT , PATCH , DELETE metodların endpoint'ler üzerinde tanımlanması gerekir.)
- Veri alışverişinde XML , HTML ve JSON formatı kullanılabilir.
- Oluşturulan istekleri bir araya daha düzenli görmek için WADL(Web Application Description Language) kullanır

03 TOP API Zafiyeti

API Security Top 10

- A1: Broken Object Level Authorization
- A2: Broken User Authentication
- A3: Excessive Data Exposure
- A4: Lack of Resources & Rate Limiting
- A5: Broken Function Level Authorization
- A6: Mass Assignment
- A7: Security Misconfiguration
- A8: Injection
- A9: Improper Assets Management
- A10: Insufficient Logging & Monitoring

A1 - Broken Object Level Authorization

Saldırganlar, API çağrısı için gönderdikleri istek içerisinde var olan ID / GUID vb. değerini değiştirerek sahip olmadığı bir veriyi elde etmesi / üzerinde işlem yapmasıdır. Bu saldırı aynı zamanda IDOR olarak da bilinir.

Senaryo

- /api/v1/user/profile/13 ile kullanıcı kendi profilini görürken buradaki sayısal değeri değiştirip farklı kullanıcı profillerini görüntülemesi

A2 - Broken User Authentication

API üzerinde kötü uygulanan kimlik doğrulaması sebebiyle, saldırganların kullanıcıların oturum bilgilerini elde etmesidir.

Senaryo

- Bir IP üzerinden uygulama login paneline birden fazla istek devamlı olarak gelmesi
 - Toplu olarak username password içerikli isteklerin gönderilmesi
 - Credentials Stuffing
- Hassas bilgilerin URL üzerinden taşınması (API Token , Password)
- Server tarafında üretilen tokenlarda validasyon eksikliği (JWT vs.)

A3 - Excessive Data Exposure

API cevaplarında filtrelemeyi client'ın yapacağını düşünerek, müşterinin yasal olarak ihtiyaç duyduğundan çok daha fazla veriyi göndermektir.

Senaryo

- Kullanıcı kendi profilini görüntülemek için istek gönderdiğinde gelen cevap içerisinde ad ,soyad ,yaş alanı dışında kullanılmayan diğer bilgilerinde gelmesi
- Kredi kartı kaydetme ve kaydedilen kredi kartlarının çağırılması

A4 - Lack of Resources & Rate Limiting

API'leri aşırı miktarda isteğe veya yük durumlarına karşı korumasız olmaları durumudur. Saldırganlar bu durumu Denial of Service veya Authentication üzerinde brute force saldırıları gerçekleştirmek için kullanmasıdır.

Senaryo

- Ürün listeleme isteği içerisinde gönderilen “size” değerinin olduğundan daha fazla girilmesi backend ve DB'i olumsuz olarak etkilemesi
- Yüksek miktarda dosya kaydetme isteğinin tekrar tekrar gönderilmesi

A5 - Broken Function Level Authorization

Uygulama içerisinde yetkilere göre ayrılmış API isteklerini tespit etmek ve yetki sahibi olunmayan fonksiyonu veya API'leri kullanmak.

Senaryo

- Frontend tarafında kullanıcıya gösterilmeyen path'leri tespit etmek ve kötüye kullanma
- Görüntüleme yetkisine sahip olduğun bir API'nin HTTP Verb'ini değiştirmek.(Delete , Put)
- Bilinen path üzerinde değişiklik yapmak
 - /api/users/v1/user/myinfo
 - /api/users/v1/user/alluser

A6 - Mass Assignment

API isteği içerisinde beklenen alanlar içerisine ek olarak farklı bir değer daha ekleyip uygulamanın beklenen dışı çalışmasını sağlamaktır.

Senaryo

- Kullanıcı kayıt etme işlemi esnasında ad,soyad,kadi,kparola beklenirken saldırgan bunların yanında krole=admin ekleyip istek içerisinde göndermesi
- Uygulama üzerinde banka kartı bilgilerini güncellerken bakiye bilgisini istek içerisine eklenmesi

A7 - Security Misconfiguration

API sunucularının zayıf yapılandırmasını sonucunda kendisi veya üzerinde çalıştığı sistem hakkında saldırganlara detaylı bilgi vermesi / kendisini kullandırmasıdır.

Senaryo

- Eksik CORS yapılandırması
- Uygulamadan dönen detaylı hata mesajları
- Eksik veya hiç yapılandırılmamış HTTP başlıkları

A8 - Injection

Kullanıcıdan alınan değerlere herhangi bir kontrol gerçekleştirmeden doğrudan yapılan işlemler uygulamanın beklenenden farklı çalışmasına sebebiyet vermesi durumudur.

Senaryo

- Saldırganlar, uygulamaya giden istekler içerisine beklenmeyen <, >, ', ", \$, %, * vs gibi karakterler eklemesi
 - SQL / NoSQL
 - LDAP
 - OS Command

A9 - Improper Assets Management

Saldırganlar tarafından test yapılan ortamda farklı amaçlar için oluşturulmuş API versiyonlarını/dökümantasyonlarını tespit edip, yeni saldırılar başlatmak için kullanılmasıdır.

Senaryo

- Var olan dışarıya açık olan API 'ların alternatifini geliştirmek eski olanlarını kaldırmamak (v1 -> v2 fakat v1'in hala canlı ortama açık olması)
- Canlı ortam testlerinde fuzz çalışması sonrasında API Dokümantasyon sayfasının tespit edilmesi

A10 - Insufficient Logging & Monitoring

Log kaydının ve uygulama monitoring'in yetersiz yapılması gelen / gelecek saldırıların veya saldırganların fark edilmemesine olanak tanır.

Senaryo

- Login sayfasına Credentials Stuffing için gelen birden fazla istek
- Uygulamanın herhangi bir kısmı içerisinde injection saldırısının yapılması
- Uygulamanın düzensiz olarak down tekrar up olması

04 Uygulama

Zafiyet Senaryoları

- Broken Object Level Authorization
- Broken Authentication
- Excessive Data Exposure
- Lack of Resources & Rate Limiting
- Mass assignment
- Injection

The background is a deep space scene with a dark blue and purple color palette. A large, partially visible planet with a blue and white horizon line is in the upper right. Numerous small white stars are scattered across the dark background.

05 Hayattan Örnekler

LazyPay - Broken Object Level Authorization

Ehraz Ahmed , Lazy Pay mobile uygulaması üzerinde authentication uygulanmayan bir API tespit etti. Tespit edilen API telefon numarası verildiğinde , telefon numarasına karşı gelen kullanıcı bilgilerinin döndüğünü gözlemleyip rastgele telefon numarası istekleri göndererek uygulama üzerindeki birden fazla kullanıcı bilgisini elde etmiştir.

Çözüm Önerisi

- Dışarıya açılan API'larda Authentication ve Authorization çalışması yapmak
- ID ile erişilecek bir obje oluşturulduğu bu değerin olabildiğinde karmaşık yapılmasını sağlamak
- Rate limiting

ThemeREX Addons Plugin - Injection

ThemeRex Addons, Wordpress üzerinde ThemeRex temalarına eşlik etmek ve onları yönetmek için kullanılan eklentidir. Bu plugin'ın dışarıya hizmet veren bir API kullanıcıdan alınan veriyi doğrudan kullanıldığı için PHP Injection zafiyeti tespit edilmiştir. Bu zafiyet kullanılarak işletim sistemi üzerinde komut çalıştırılabilinmiştir.

Çözüm Önerisi

- Input sanitize

06 Kapanış

Teşekkürler

Linkedin : <https://www.linkedin.com/in/erdemyildiz/>

Github : <https://github.com/Erdemstar>

Medium : <https://medium.com/@erdemstar08>

Email : erdem-yildiz@windowslive.com / erdem.yildiz@bgasecurity.com

Kaynaklar

- Lazypay IDOR :
<https://gadgets.ndtv.com/internet/news/lazypay-security-flaw-vulnerability-fix-payu-2465220>
- 1.4 million doctor data :
<https://apisecurity.io/issue-79-1-4-million-doctor-records-scraped-using-api/>
- Wordpress ThemeRex
<https://www.wordfence.com/blog/2020/02/zero-day-vulnerability-in-theme-rex-addons-plugin-exploited-in-the-wild/>
- https://owasp.org/www-pdf-archive/API_Security_Top_10_RC_-_Global_AppSec_AMS.pdf
- https://owasp.org/www-pdf-archive/OWASP_APIs_Security_Project_Kick_Off.pdf
- <https://apisecurity.io>
- <https://www.redhat.com/en/topics/integration/whats-the-difference-between-soap-rest>
- <https://smartbear.com/blog/soap-vs-rest-whats-the-difference/>
- <https://www.guru99.com/api-vs-web-service-difference.html>

Kaynaklar

- <https://salt.security/blog/owasp-api-security-top-10-explained>
- <https://owasp.org/www-project-api-security/>

