Introduction to Business Logic (İş Mantığına Giriş)

Introduction to Business Logic (İş Logic'e Giriş)

Çok fonksiyonlu dinamik bir web uygulamasında iş mantığı kusurları için test, alışılmadık yöntemlerle düşünmeyi gerektirir. Bir uygulamanın kimlik doğrulama mekanizması, bir kullanıcıyı doğrulamak için bu özel sırayla 1, 2, 3 adımlarını gerçekleştirmek amacıyla geliştirilirse. Kullanıcı 1 adımdan 3'e kadar giderse ne olur? Bu basit örnekte, uygulama açık ve başarısız olarak erişim sağlıyor mu; erişimi mi yoksa 500 mesajla sadece hata mı?

Yapılabilecek birçok örnek vardır, ancak tek sürekli ders “konvansiyonel bilgeliğin dışında düşünmek” dir. Bu tür bir güvenlik açığı tarayıcısı tarafından tespit edilemez ve penetrasyon test
cihazının becerilerine ve yaratıcılığına dayanır. Buna ek olarak, bu tür bir güvenlik açığı genellikle tespit edilmesi en zor olanlardan biridir ve genellikle uygulamaya özgüdür, ancak aynı zamanda, istismar edilirse, genellikle uygulamaya en zararlı olanlardan biridir.

İş mantığı kusurlarının sınıflandırılması az çalışılmamıştır; iş kusurlarının sömürülmesi gerçek dünya sistemlerinde sıklıkla gerçekleşse de ve birçok uygulamalı güvenlik açığı araştırmacısı bunları araştırmaktadır. En büyük odak noktası web uygulamalarındadır. Topluluk içinde, bu sorunların özellikle yeni kavramları temsil edip etmediği veya iyi bilinen ilkelerin varyasyonları olup olmadığı konusunda tartışmalar vardır.

İş mantığı kusurlarının test edilmesi, mantıksal veya sonlu durum testlerine odaklanan fonksiyonel testçiler tarafından kullanılan test türlerine benzer. Bu tür testler, güvenlik uzmanlarının biraz farklı
düşünmelerini, kötüye kullanılan ve kötüye kullanım vakaları geliştirmelerini ve fonksiyonel testçiler tarafından benimsenen test tekniklerinin çoğunu kullanmalarını gerektirir. İş mantığı kötüye
kullanımı vakalarının otomasyonu mümkün değildir ve test cihazının becerilerine ve bunların tam iş süreci ve kuralları hakkındaki bilgilerine dayanan manuel bir sanat olmaya devam etmektedir.

Business Limits and Restrictions (İş Sınırları ve Kısıtlamaları)

Uygulama tarafından sağlanan iş fonksiyonuna ilişkin kuralları göz önünde bulundurun. İnsanların davranışlarında herhangi bir sınır veya kısıtlama var mı? Ardından, başvurunun bu kuralları uygulayıp uygulamadığını düşünün. İşletmeye aşinaysanız uygulamayı doğrulamak için test ve analiz vakalarını tanımlamak genellikle oldukça kolaydır. Üçüncü taraf bir testçiyseniz, sağduyunuzu kullanmanız ve işletmeye uygulama tarafından farklı operasyonlara izin verilip verilmeyeceğini
sormanız gerekecektir.

Bazen, çok karmaşık uygulamalarda, test cihazı başlangıçta uygulamanın her yönünü tam olarak anlamayacaktır. Bu durumlarda, müşterinin test cihazını uygulama üzerinden yürümesi en iyisidir,
böylece gerçek test başlamadan önce uygulamanın sınırlarını ve amaçlanan işlevselliğini daha iyi anlayabilirler. Ek olarak, test sırasında geliştiricilere doğrudan bir çizgiye sahip olmak (mümkünse), uygulamanın işlevselliği ile ilgili herhangi bir soru ortaya çıkarsa büyük ölçüde yardımcı olacaktır.

Challenges of Logic Testing (Mantık Testinin Zorlukları)

Otomatik araçlar bağlamı anlamakta zorlanır, bu nedenle bu tür testleri yapmak bir kişiye bağlıdır. Aşağıdaki iki örnek, uygulamanın işlevselliğini, geliştiricinin niyetlerini ve bazı yaratıcı “kutu dışı”
düşüncenin uygulamanın mantığını nasıl kırabileceğini gösterecektir. İlk örnek basit bir parametre manipülasyonu ile başlarken, ikincisi uygulamayı tamamen altüst etmeye yol açan çok adımlı bir sürecin gerçek bir dünya örneğidir.

Example 1: (**Örnek 1** :)

Bir e-ticaret sitesinin, kullanıcıların satın almak için öğeleri seçmelerine, özet bir sayfayı görüntülemelerine ve ardından satışı ihale etmelerini sağladığını varsayalım. Ya bir saldırgan özet sayfasına geri dönebildiyse, aynı geçerli oturumlarını sürdürür ve bir öğe için daha düşük bir maliyet enjekte edip işlemi tamamlayıp sonra kontrol edebilirdi?

Example 2: (**Örnek 2** :)

Kaynakların tutulması / kilitlemesi ve bu öğeleri çevrimiçi satın almalardan alıkoymak, saldırganların daha düşük bir fiyata ürün satın almasına neden olabilir.

Bu sorunun karşı önlemi, yalnızca doğru fiyatın tahsil edilebilmesini sağlamak için zaman aşımları ve mekanizmaları uygulamaktır.

Example 3: **(Örnek 3** :)

Ya bir kullanıcı kulüp / sadıklık hesabına bağlı bir işlem başlatabilirse ve daha sonra hesaplarına puan eklendikten sonra işlem dışı iptal edildiyse? Puan/krediler hala hesaplarına uygulanacak mı?

Tools (Araçlar)

İş süreçlerinin geçerli durumlarda doğru şekilde çalıştığını test etmek ve doğrulamak için araçlar olsa da, bu araçlar mantıksal güvenlik açıklarını tespit edemez. Örneğin, bir kullanıcının iş sürecini
düzenleme parametreleri yoluyla akmasını atlatıp atamayacağını, sınırlı kaynaklara erişmek için kaynak isimlerini tahmin edip etme veya insan testçilerinin bu durumdan şüphelenmesine yardımcı olacak herhangi bir mekanizmaya sahip olup olmadığını tespit etme imkanı yoktur.

Aşağıdakiler, iş mantığı sorunlarını belirlemede yararlı olabilecek bazı yaygın araç türleridir.

Ekonsları yüklerken, talep ettikleri izinleri ve tarayıcı kullanım alışkanlıklarınızı göz önünde bulundururken her zaman gayretli olmalısınız.

Intercepting Proxy (Proxy'yi Yakalamak)

HTTP Trafiğinin İsteği ve Yanıt Bloklarını Gözlemlemek

- OWASP Zed Attack Proxy
- Burp Proxy

Web Browser Plug-ins (Web Browser Plug-ins)

HTTP/HTTPS başlıklarını görüntülemek ve değiştirmek, parametreleri yayınlamak ve Gverser'ın DOM'unu gözlemlemek

- Tamper Data for FF Quantum
- Tamper Chrome (for Google Chrome)

Miscellaneous Test Tools (Çeşitli Test Araçları)

- Web Developer toolbar
    - Web Geliştirici uzantısı, çeşitli web geliştirici araçlarına sahip tarayıcıya bir araç çubuğu düğmesi ekler. Bu, Firefox için Web Geliştirici uzantısının resmi portudur.
- HTTP Request Maker for Chrome
- HTTP Request Maker for Firefox
    - Request Maker penetrasyon testi için bir araçtır. Bununla birlikte, web sayfaları tarafından yapılan istekleri kolayca yakalayabilir, URL'yi, başlıkları ve POST verilerini kurcalayabilir ve elbette yeni isteklerde bulunabilirsiniz.
- Cookie Editor for Chrome
- Cookie Editor for Firefox
    - Çerez editörü bir çerez yöneticisidir. Çerezleri ekleyebilir, silebilir, düzenleyebilir, arayabilir, koruyabilir ve engelleyebilirsiniz

References (Referanslar)

Whitepapers (Beyaz Kağıtlar)

- The Common Misuse Scoring System (CMSS): Metrics for Software Feature Misuse Vulnerabilities - NISTIR 7864
- Finite State testing of Graphical User Interfaces, Fevzi Belli
- Principles and Methods of Testing Finite State Machines - A Survey, David Lee, Mihalis Yannakakis
- Security Issues in Online Games, Jianxin Jeff Yan and Hyun-Jin Choi
- Securing Virtual Worlds Against Real Attack, Dr. Igor Muttik, McAfee
- Seven Business Logic Flaws That Put Your Website At Risk – Jeremiah Grossman Founder and CTO, WhiteHat Security
- Toward Automated Detection of Logic Vulnerabilities in Web Applications - Viktoria Felmetsger Ludovico Cavedon Christopher Kruegel Giovanni Vigna

OWASP Related (OWASP ile İlgili)

- How to Prevent Business Flaws Vulnerabilities in Web Applications, Marco Morana

Useful Web Sites (Yararlı Web Siteleri)

- Abuse of Functionality
- Business logic
- Business Logic Flaws and Yahoo Games
- CWE-840: Business Logic Errors
- Defying Logic: Theory, Design, and Implementation of Complex Systems for Testing Application Logic
- Software Testing Lifecycle

Books (Kitaplar)

- The Decision Model: A Business Logic Framework Linking Business and Technology, By
  Barbara Von Halle, Larry Goldberg, Published by CRC Press, ISBN1420082817 (2010)