Web Uygulama Güvenliğine Hibrid Yaklaşım

Onur Yılmaz, Şubat 2010, WGT E-Dergi 4. Sayı

"Web" kavramının doğmasıyla beraber, hem farklı sektörlerde konumlanmış çeşitli ölçekteki kurumların web sitelerinin sayısında hem de farklı amaçlarla hazırlanmış kişisel kullanıma hitap eden web sitelerinin sayısında gün geçtikçe hızlı bir artış yaşanmaya başlamıştır.

Bilgi Teknolojilerinin hızla gelişmesi, ihtiyaçlar doğrultusunda web teknolojilerini de yanında sürüklemiş ve 2010 yılı itibariyle, Web 2.0 – 3.0 kavramları doğmuş, yayılmış, web uygulamaları kritik bilgileri saklar, son kullanıcıya erişim yetkileri doğrultusunda sunar ve tamamen kullanıcı etkileşimli bir yapıya bürünür hale gelmiştir.

Uygulamaların bu denli önemli bir yer edinmesi ve bilgiye erişimin çok kolay olması nedeniyle, uygulamalarımızı nasıl güvenli kılarız noktasında yeni bir yaklaşım olan "Hibrid Yaklaşım" konusunu ele alacağız.

Uygulamaların Güvensiz Olması ve Sonuçları

Programlama dili / çatısı seçiminin projenin kapsamının ihtiyaçlarını tam olarak gidememesi, yetersiz güvenlik farkındalığı, geliştirme döngüsünün izlenmemesi yada bu döngünün herhangi bir yerine güvenli yazılım geliştirme süreçlerinin dahil edilmemesi ve en önemlisi güvenlik testlerinin yapılmaması veya yetersiz kalması nedeniyle uygulamalarda birçok güvenlik sorunu doğmaktadır.

Bu güvenlik sorunlarının sonucunda uygulamalar çeşitli saldırılara maruz kalmakta ve;

- Kritik uygulamalarının hizmet verememesi
- Kritik bilgilerin dışarıya sızdırılması / ele geçirilmesi
- Uyumluluk kaybı
- Kazanç kaybı
- Yasal mesuliyet
- Telif hakları
- Müşteri inancının zedelenmesi
- Prestij kaybı
- vb...

gibi riskler ve sonuçlar doğurmaktadır.

Güvenlik uzmanları ve yazılımcılar arasındaki iletişim sorunu, web teknolojileri ve uygulama çeşitliliğindeki artış, hangi uygulamaların kritik bilgi taşıyıp taşımadığı konusundaki belirsizlikler, istenen fonksiyonellik ile güvenliğin ön planda tutulması ile ortaya çıkan fonksiyonellik arasındaki farklar, uygulamalardaki bu sorunları tetiklemekte ve güvenlik sorunlarının çözümü için yeni yaklaşımların oluşmasını sağlamaktadır.

Uygulama erişilebilir halde iken yada yayında iken güvenlik testleri gerçekleştirilerek olası bütün güvenlik zafiyetlerinin tespit edileceği varsayımı. Bu varsayımın eksik yanları şunlar;

- Testler uygulama çalışır durumdayken, uygulamalardaki güvenlik sorunlarına neden olabilecek hataları black-box yöntemiyle bulmaya motive olan uygulama güvenlik tarayıcıları ile gerçekleştirildiğinden dolayı;
    - Uygulamada yavaşlama ya da farklı hataların oluşmasına neden olabilir.
    - Uygulamada yavaşlığın oluşması işlem yapmak isteyen kullanıcıları hiç memnun etmeyecektir.
    - Uygulamanın test süreci esnasında oluşabilecek hataların başka kullanıcılar tarafından da görülebilme ihtimali olmasından dolayı, kötü niyetli bir kullanıcının bu bilgiye erişmesi potansiyel bir risk doğuracaktır.
- Güvenlik testlerinin kapsamı, uygulamanın hepsini kapsamayabilir. (authorization – out of band attacks vb.)
- Uygulama Güvenliği Tarayıcısının özelliklerinin yetersiz olması
- Ve son olarak en önemli sorun: **Bu yöntemin tek başına yeterli olmaması !**

Statik Kaynak Kodu Analizi

White-Box Testing kapsamında yer alan statik kaynak kodu analizi, geliştirilen uygulamanın kaynak kodlarının güvenlik uzmanı tarafından, güvenlik sorunlarına karşı incelenmesine dayanmaktadır.

Geliştirilen uygulamanın çok kompleks bir yapı üzerine oturtulması, sektördeki güvenlik hizmeti veren firmalarda bu konuda uzmanlaşmış nitelikli eleman azlığı ve farklı ve çok sayıda uygulama geliştirme platformlarının olması nedeniyle kaynak kod analiz işlemini otomatize olarak yapan araçların kullanımı yaygınlaşmıştır.

İhtiyaçların sonsuz olduğu günümüzde, bu sonsuz ihtiyaçları giderebilmek amacıyla farklı mantaliteye sahip yazılımcıların elinden çıkan çok farklı web teknolojisi altyapısına sahip uygulamalar ve bu uygulamalarda çok farklı kod blokları bulunabilmektedir.

Bu noktaya istinaden;kaynak kod analizinin manuel olarak yapılması çok uzun süreceğinden ve bu uzun süreç içerisinde kaynak kod analizi yapan kişinin motivasyonunda azalmalar olacağından dolayı pek tercih edilmemektedir.

Statik kaynak kod analizinin otomatize olarak yapılması, süreci azaltacak olsa bile yukarıda saydığımız nedenlerden dolayı false-positiveler ve false-negativeler de olacaktır. Ayrıca otomatize olarak bulunamayacak kodlama hatalarınında olma ihtimali, statik kaynak kod analizini de tek başına yeterli kılmayacaktır.

Hibrid Yaklaşım

Uygulama güvenlik tarayıcıları ile yapılan testler ve kaynak kod analizinin her biri tek başına yeterli olmayacağından dolayı Hibrid Yaklaşım modelinde Kaynak Kod Analizi ve Uygulama Güvenlik Testleri birleştirilerek, daha efektif sonuçların alınması ve bu sonuçlar sonucunda da gerekli görülen implementasyonların yapılması amaçlanmıştır.

Böylelikle uygulama güvenlik testlerinde kullanılan araçlar, uygulamaya son kullanıcı/saldırgan gibi davranarak test edecek, statik kaynak kod analizi araçları da kodlama hatalarına yoğunlaşarak black-box testlerin yetersiz kaldığı durumları gidermeye çalışacaktır. Hem statik kaynak kod analizi hem de uygulama güvenlik testlerinin yapılması, uygulamada karşılaşılabilecek güvenlik hatalarını en az seviyeye indirecektir.

Hibrid Yaklaşım Yeterli mi?

Hibrid yaklaşım metodunun izlenmesi ve bu yaklaşım sonucunda elde edilen veriler doğrultusundaki uygulamaya yapılan implementasyonlar güvenlik sorunları riskini en az düzeye indirse de tam manasıyla yeterli olmayacaktır.

Statik kaynak kod analizi ile yada otomatize olarak uygulama güvenlik testleri yapan araçlarla, uygulamadaki dizayn hatalarından kaynaklı mantıksal açıkların yakalanması mümkün değildir, en azından şimdilik.

Bu nedenle “hacker” gibi düşünme yetisi olan ve uygulamalara karşı bu şekilde yaklaşabilen kurum ya da kişilerden, özellikle mantıksal açıklıkların yakalanması konusunda “penetrasyon testi” hizmeti almanız, ultra hibrid bir yaklaşım yakalamanızı sağlayacaktır.