Testing for Session Hijacking (Oturum Kaçırma için test)

Summary (Özet)

Kullanıcı oturumu çerezlerine erişen bir saldırgan, bu tür çerezleri sunarak onları taklit edebilir. Bu saldırı oturum kaçırma olarak bilinir. İnsan saldırganlarını, yani kurbanın kullandığı ağı kontrol
eden saldırganları düşünürken, oturum çerezleri HTTP üzerinden saldırgana gereksiz yere maruz kalabilir. Bunu önlemek için, oturum çerezleri ile işaretlenmelidir
`Secure` Bağlılık, sadece HTTPS üzerinden iletilmeleri için.

Not et the the (hada) `Secure` Netlik, web uygulaması tamamen HTTPS üzerinde konuşlandırıldığında da kullanılmalıdır, aksi takdirde aşağıdaki çerez hırsızlığı saldırısı
mümkündür. Bunu varsayalım
`example.com` HTTPS üzerinden tamamen dağıtılır, ancak oturum çerezlerini işaretlemez. `Secure` . . Aşağıdaki saldırı adımları mümkündür:

1. Mağdur bir talep gönderir `http://another-site.com` . .
2. Saldırgan, ilgili yanıtı bozar, böylece bir talebi tetikler. `http://example.com` . .
3. Tarayıcı artık erişmeye çalışıyor `http://example.com` . .
4. İstek başarısız olsa da, oturum çerezleri HTTP üzerinden açıkta sızdırılır.

Alternatif olarak, HSTS kullanarak HTTP'nin kullanımını yasaklayarak oturum kaçırması
önlenebilir. Burada çerez kepçeleme ile ilgili bir incelik olduğunu unutmayın. Özellikle, oturum çerezleri yayınlandığında tam HSTS evlat edinme gereklidir.
`Domain` Hibe seti.

Tam HSTS evlat edinme, Stefano Calzavara, Alvise Rabitti, Alessio Ragazzo ve Michele Bugliesi tarafından *Web Oturumlarında Dürüstlük Kusurları Testi adlı* bir makalede açıklanmıştır. Tam HSTS evlat edinme, bir ev sahibi HSTS'yi kendisi ve tüm alt kıyafetleri için etkinleştirdiğinde ortaya

çıkar. Kısmi HSTS evlat edinme, bir ev sahibinin HSTS'yi sadece kendisi için etkinleştirmesidir.

The ile birlikte `Domain` Nitelik seti, oturum çerezleri alt-domanlar arasında paylaşılabilir. HTTP
üzerinden gönderilen şifresiz çerezlerin açıklanmasını önlemek için alt-domanlarla HTTP kullanımından kaçınılmalıdır. Bu güvenlik kusurunu örneklemek için, web sitesinin
`example.com` HSTS'yi etkisiz hale getirir `includeSubDomains` Seçenek. Web sitesi oturum çerezlerini ile yayınlar `Domain` atfedilen atfed `example.com` . . Aşağıdaki saldırı mümkündür:

1. Mağdur bir talep gönderir `http://another-site.com` . .
2. Saldırgan, ilgili yanıtı bozar, böylece bir talebi tetikler. `http://fake.example.com` . .
3. Tarayıcı artık erişmeye çalışıyor `http://fake.example.com` HSTS konfigürasyonu tarafından izin verilir.
4. Talep bir alt mayına gönderildiği için `example.com` The ile birlikte `Domain` Nitelik seti, HTTP üzerinden açıkta sızdırılan oturum çerezlerini içerir.

Bu saldırıyı önlemek için apex etki alanında tam HSTS etkinleştirilmelidir.

Test Objectives (Test Hedefleri)

- Savunmasız oturum çerezlerini belirleyin.
- Savunmasız çerezleri kaçırın ve risk seviyesini değerlendirin.

How to Test (Nasıl Test Edilir)

Test stratejisi ağ saldırganlarını hedef alır, bu nedenle yalnızca tam HSTS benimsemesi olmadan sitelere uygulanması gerekir (tamamlı HSTS benimseyen siteler güvenlidir, çünkü çerezleri HTTP üzerinden iletilmez). Test altında web sitesinde iki test hesabı olduğunu varsayıyoruz, biri kurban olarak hareket etmek ve biri saldırgan olarak hareket etmek. Saldırganın HTTP üzerinden ifşaya karşı korunmayan tüm çerezleri çaldığı ve mağdurun hesabına erişmek için web sitesine sunduğu
bir senaryoyu simüle ediyoruz. Bu çerezler kurbanın adına hareket etmek için yeterliyse, oturum kaçırma mümkündür.

İşte bu testi yürütme adımları:

1. Mağdur olarak web sitesine giriş yapın ve kimlik doğrulaması gerektiren güvenli bir işlev sunan herhangi bir sayfaya ulaşın.
2. Aşağıdaki koşullardan herhangi birini karşılayan tüm kurabiye kavanozlarından silin.
    - HSTS'nin evlat edinilmesi söz konusu değilse: `Secure` Hibe belirlenmiştir.
    - Kısmi HSTS benimsenmesi durumunda: `Secure` Nitelik ayarlanır ya da `Domain` Hibe ayarlanmamıştır.
3. Kurabiye kavanozunun bir anlık görüntüsünü kaydedin.
4. 1. adımda tanımlanan güvenli işlevi tetikleyin.
5. 4. adımdaki operasyonun başarılı bir şekilde yerine getirilip getirilmediğini gözlemleyin. Eğer öyleyse, saldırı başarılı oldu.
6. Kurabiye kavanozuyu temizleyin, saldırgan olarak giriş yapın ve 1. adımda sayfaya ulaşın.
7. Kurabiye kavanozunda, tek tek, 3 adımda tasarruf edilen kurabiyeler yazın.
8. 1. adımda tanımlanan güvenli işlevi tekrar tetikleyin.
9. Kurabiye kavanozunu temizleyin ve kurban olarak tekrar giriş yapın.
10. 8 adımdaki operasyonun kurbanın hesabında başarılı bir şekilde gerçekleştirilip gerçekleştirilmediğini gözlemleyin. Eğer öyleyse,saldırı başarılı oldu; aksi takdirde, site oturum kaçırmaya karşı güvenlidir.

Kurban ve saldırgan için iki farklı makine veya tarayıcı kullanmanızı öneririz. Bu, web uygulaması belirli bir çerezden etkinleştirilen erişimi doğrulamak için parmak izi alırsa yanlış pozitiflerin sayısını
azaltmanıza olanak tanır. Test stratejisinin daha kısa ama daha az kesin bir varyantı sadece bir test hesabı gerektirir. Aynı modeli takip eder, ancak 5. adımda durur (bunun 3. adımı işe yaramaz hale getirdiğini unutmayın).

Tools (Araçlar)

- OWASP ZAP
- JHejack - sayısal bir oturum kaçırma aracı