Testing WebSockets (WebSockets Testi)

Summary (Özet)

Geleneksel olarak, HTTP protokolü yalnızca TCP bağlantısı başına bir istek / yanıt verir. Asyankron JavaScript ve XML (AJAX), müşterilerin sunucuya (sayfa yenilemesi olmadan arka planda) veri göndermelerini ve almalarını sağlar, ancak AJAX istemcinin istekleri başlatmasını ve sunucu yanıtlarını beklemesini (yarı-duplex) gerektirir.

WebSockets, istemcinin veya sunucunun "tam-duplex" (iki yönlü) bir iletişim kanalı oluşturmasına izin vererek istemcinin ve sunucunun gerçekten asenkron olarak iletişim kurmasını sağlar. WebSoketleri ilk *yükseltme* el sıkışmalarını HTTP üzerinden gerçekleştirir ve o andan itibaren tüm
iletişim araçlarının kullanılmasıyla TCP kanalları üzerinden gerçekleştirilir. Daha fazlası için, WebSocket Protokolü'ne bakın.

Origin (Kökeni)

Doğrulamanın amacı sunucunun sorumluluğundadır `Origin` İlk HTTP WebSocket el sıkışma başlığı. Sunucu, ilk WebSocket el sıkışmasında orijinal başlığı doğrultmazsa, WebSocket sunucusu herhangi bir kökene ait bağlantıları kabul edebilir. Bu, saldırganların CSRF benzeri sorunlara izin veren WebSocket sunucusu çapraz-domain ile iletişim kurmasına izin verebilir. Ayrıca 102017 A5-Broken Erişim Kontrolüne de bakınız.

Confidentiality and Integrity (Gizlilik ve Dürüstlük)

WebSockets, şifrelenmemiş TCP veya şifrelenmiş TLS üzerinde kullanılabilir. Şifrelenmemiş WebSoketleri kullanmak `ws://` URI şeması kullanılır (varsayılan bağlantı noktası 80), şifreli (TLS) WebSockets'ı kullanmak için kullanılır `wss://` URI şeması kullanılır (varsayılan bağlantı noktası 443). Ayrıca 102017 A3-Antifiel Veri Maruziyetine de bakınız.

Input Sanitization (Girdi Sanitasyon)

Güvenilir olmayan kaynaklardan kaynaklanan herhangi bir veride olduğu gibi, veriler de düzgün bir şekilde dezanıtılması ve kodlanması gerekir. Ayrıca En İyi 102017 A1-Enjeksiyon ve En İyi 10 2017 A7-Cross-Site Scripting (XSS)

Test Objectives (Test Hedefleri)

- WebSoketlerin kullanımını belirleyin.
- Normal HTTP kanallarında aynı testleri kullanarak uygulanmasını değerlendirin.

How To Test (Nasıl Test Edilir)

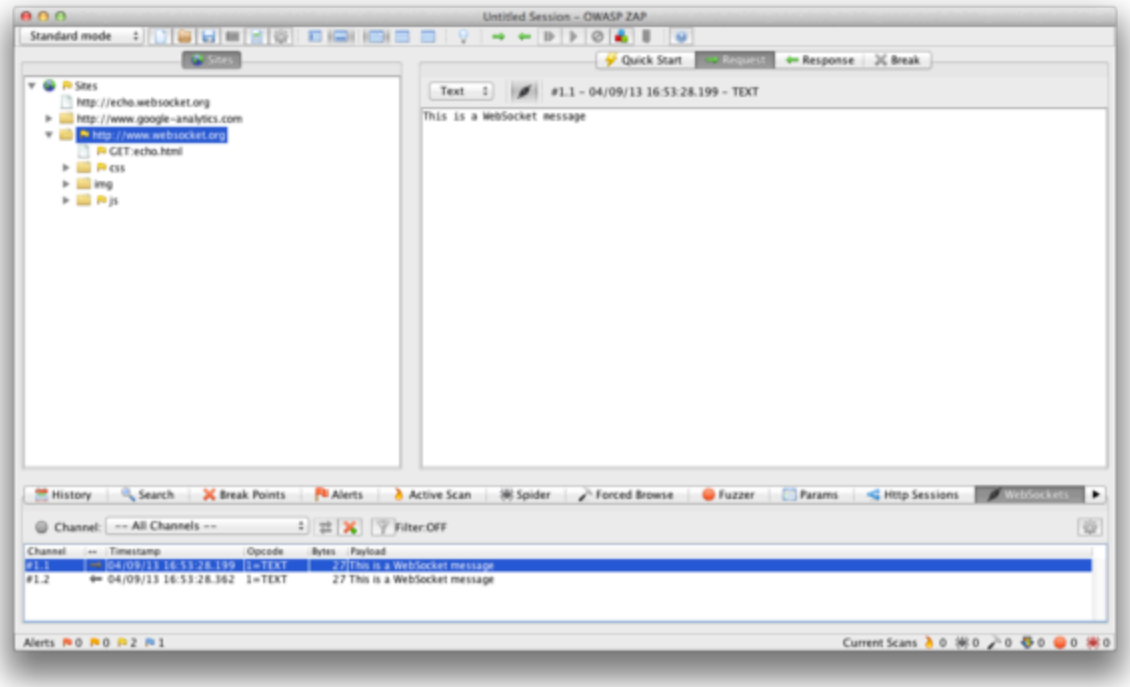
Black-Box Testing (Siyah-Kutu Testi)

1. Uygulamanın WebSockets kullandığını belirleyin.
    - istemci tarafı kaynak kodunu inceleyin `ws://` ya da `wss://` URI planı.
    - Network WebSocket iletişimini görüntülemek için Google Chrome'un Geliştirici Araçlarını kullanın.
    - ZAP'in WebSocket sekmesini kullanın.
2. Kökeni.
    - Bir WebSocket istemcisi (aşağıdaki Araçlar bölümünde bulunabilir) uzaktan WebSocket sunucusuna bağlanmaya çalışır. Bir bağlantı kurulursa, sunucu WebSocket el sıkışmanın orijinal başlığını kontrol etmeyebilir.
3. Gizlilik ve Dürüstlük.
    - WebSocke bağlantısının hassas bilgileri taşımak için SSL kullandığını kontrol edin `wss://` . .
    - Güvenlik sorunları için SSL Uygulamasını kontrol edin (Davad Sertifikası, BEAST, SUÇ, RC4, vb.) Bu kılavuzun Zayıf Taşıma Araç Takımı Güvenliği Bölümü için Test bölümüne bakın.
4. Kimlik doğrulaması.
    - WebSoketler kimlik doğrulamayı kaldırmıyor, normal black-box kimlik doğrulama testleri yapılmalı. Bu kılavuzun Kimlik Doğrulama Test bölümlerine bakın.

5. Yetkilendirme.
    - WebSockets yetkilendirmeyi kaldırmıyor, normal kara kutu yetkilendirme testleri yapılmalı. Bu kılavuzun Yetkilendirme Test bölümlerine bakın.
6. Girdi Sanitasyon.
    - WebSocket isteğini ve yanıtlarını tekrarlamak ve karıştırmak için ZAP'in WebSocket sekmesini kullanın. Bu kılavuzun Veri Doğrulama Test bölümlerine bakın.

Example 1 (Örnek 1)

Uygulamanın WebSockets kullandığını belirledikten sonra (yukarıda açıklandığı gibi) WebSocket talebini ve yanıtlarını engellemek için OWASP Zed Attack Proxy'yi (ZAP) kullanabiliriz. ZAP daha sonra WebSocket isteğini / yanıtlarını tekrar oynatmak ve karıştırmak için kullanılabilir.



Şekil 4.11.10-1: ZAP WebSoketleri

Example 2 (Örnek 2)

Bir WebSocket istemcisi (aşağıdaki Araçlar bölümünde bulunabilir) uzaktan WebSocket sunucusuna bağlanmaya çalışır. Bağlantıya izin verilirse, WebSocket sunucusu WebSocket el sıkışmanın orijinal başlığını kontrol etmeyebilir. Daha önce çapraz-domain WebSocket iletişiminin mümkün olduğunu doğrulamak için engellenen talepleri tekrarlama girişimi.

Şekil 4.11.10-2: WebSoket Müşterisi

Gray-Box Testing (Gri-Kutu Testi)

Gri kutu testi kara kutu testine benzer. Gri kutu testinde, kalem test cihazı uygulama hakkında kısmi bilgiye sahiptir. Buradaki tek fark, beklenen WebSocket talebini ve yanıtlarını içeren test edilen uygulama için API belgelerine sahip olmanızdır.

Tools (Araçlar)

- OWASP Zed Attack Proxy (ZAP)
- WebSocket Client

- Google Chrome Simple WebSocket Client

References (Referanslar)

- HTML5 Rocks - Introducing WebSockets: Bringing Sockets to the Web
- W3C - The WebSocket API
- IETF - The WebSocket Protocol
- Christian Schneider - Cross-Site WebSocket Hijacking (CSWSH)
- Jussi-Pekka Erkkilä - WebSocket Security Analysis (PDF)
- Robert Koch- On WebSockets in Penetration Testing
- DigiNinja - OWASP ZAP and Web Sockets