Testing for SSI Injection (SSI Enjeksiyonu için Test)

Summary (Özet)

Web sunucuları genellikle geliştiricilere, tam teşekküllü sunucu tarafı veya istemci tarafı dilleriyle uğraşmak zorunda kalmadan statik HTML sayfalarına küçük dinamik kod parçaları ekleme olanağı sunar. Bu özellik Server-Side Includes (SSI) tarafından sağlanmaktadır.

Server-Side Includes, web sunucusunun sayfayı kullanıcıya sunmadan önce kaldırdığı direktiflerdir. Sadece çok basit görevleri yerine getirmeniz gerektiğinde, sunucu tarafı komut dosyası dillerini
kullanarak CGI programlarını veya yerleştirme kodunun kullanılmasına bir alternatifi temsil ederler. Ortak SGK uygulamaları, harici dosyaları içermeleri, web sunucusu CGI ortam değişkenlerini ayarlamak ve yazdırmak veya harici CGI komutlarını uygulamak için direktifler (komutalar) sağlar.

SSI, Uzaktan Komuta İnfazı'na (RCE) yol açabilir, ancak çoğu web sunucusuna sahiptir. `exec` Varsayılan olarak direktif devre dışı.

Bu, klasik bir komut dosyası enjeksiyonu zafiyetine çok benzer bir güvenlik açığıdır. Bir hafifletme, web sunucusunun SSI'ya izin verecek şekilde yapılandırılması gerektiğidir. Öte yandan, SSI enjeksiyon güvenlik açıklarının kullanılması genellikle daha basittir, çünkü SSI direktifleri anlaşılması kolaydır ve aynı zamanda oldukça güçlüdür, örneğin, dosyaların içeriğini çıkarabilir ve sistem komutlarını yürütebilirler.

Test Objectives (Test Hedefleri)

- SSI enjeksiyon noktalarını belirleyin.
- Enjeksiyonun ciddiyetini değerlendirin.

How To Test (Nasıl Test Edilir)

Kullanılabilir SSI için test etmek için, SSI direktiflerini kullanıcı girişi olarak enjekte edin. SSI etkinse ve kullanıcı girişi doğrulaması düzgün bir şekilde

uygulanmadıysa, sunucu direktifi yürütür. Bu, klasik bir komut dosyası dil enjeksiyonu savunmasızlığına çok benzer, çünkü kullanıcı girdisi uygun şekilde doğrulanmadığında ve sanitize edilmediğinde ortaya çıkar.

İlk olarak web sunucusunun SSI direktiflerini destekleyip desteklemediğini belirleyin. Çoğu zaman, cevap evet, çünkü SSI desteği oldukça yaygındır. SSI direktiflerinin desteklenip desteklenmediğini belirlemek için, bilgi toplama tekniklerini kullanarak hedefin çalıştığı web sunucusu türünü keşfedin (bkz. Parmak İzi Web Sunucusu). Koda erişiminiz varsa, belirli anahtar kelimeler için web Sunucusu yapılandırma dosyalarında arama yaparak SSI direktiflerinin kullanılıp kullanılmadığını belirleyin.

SSI direktiflerinin etkinleştirildiğini doğrulamanın bir başka yolu da sayfalarla kontrol etmektir. `.shtml` SSI direktifleri ile ilişkilendirilen uzatma. Kullanımının kullanımı `.shtml` Uzatma zorunlu değildir, bu yüzden herhangi bir şey bulmamak `.shtml` Dosyalar mutlaka hedefin SSI enjeksiyon saldırılarına karşı savunmasız olmadığı anlamına gelmez.

Bir sonraki adım, SSI enjeksiyonunun istismar edilebilir olup olmadığını görmek için tüm olası kullanıcı girdi vektörlerini belirlemek ve test etmektir.

Öncelikle kullanıcı girişine izin verilen tüm sayfaları bulun. Muhtemel girdi vektörleri başlıklar ve kurabiyeler de içerebilir. Girişin nasıl depolandığını ve kullanıldığını, yani girişin bir hata mesajı veya sayfa öğesi olarak döndürülüp değiştirilmediğini ve bir şekilde değiştirilip değiştirilmediğini belirleyin. Kaynak koduna erişim, giriş vektörlerinin nerede olduğunu ve girişin nasıl ele alındığını daha kolay belirlemenize yardımcı olabilir.

Potansiyel enjeksiyon noktalarının bir listesine sahip olduğunuzda, girişin doğru bir şekilde doğru onaylanıp doğrulanmadığını belirleyebilirsiniz. SSI direktiflerinde kullanılan karakterleri enjekte etmenin mümkün olduğundan emin olun `<!#=/.">` ve `[a-zA-Z0-9]`

Aşağıdaki örnek değişkenin değerini döndürür. references Referanslar bölümü, belirli bir sistemi daha iyi değerlendirmenize yardımcı olmak için sunucuya özgü belgelerle yararlı bağlantılara sahiptir.

```
<!--#echo var="VAR" -->
```

Kullanırken `include` Yönerge, tedarik edilen dosya bir CGI senaryosu ise, bu yönerge CGI senaryosunun çıktısını içerecektir. Bu yönerge, bir dosyanın içeriğini veya liste dosyalarını bir dizine dahil etmek için de kullanılabilir:

```
<!--#include virtual="FILENAME" -->
```

Bir sistem komutunun çıktısını iade etmek için:

```
<!--#exec cmd="OS_COMMAND" -->
```

Uygulama savunmasızsa, direktif enjekte edilir ve bir dahaki sefere sayfa servis edildiğinde sunucu tarafından yorumlanır.

Web uygulaması dinamik olarak oluşturulmuş bir sayfa oluşturmak için bu verileri kullanıyorsa, HTTP başlıklarına da SSI direktifleri enjekte edilebilir:

```
GET / HTTP/1.1
Host: www.example.com
Referer: <!--#exec cmd="/bin/ps ax"-->
User-Agent: <!--#include virtual="/proc/version"-->
```

Tools (Araçlar)

- Web Proxy Burp Suite
- OWASP ZAP
- String searcher: grep

References (Referanslar)

- Nginx SSI module
- Apache: Module mod_include
- IIS: Server Side Includes directives
- Apache Tutorial: Introduction to Server Side Includes
- Apache: Security Tips for Server Configuration
- SSI Injection instead of JavaScript Malware
- IIS: Notes on Server-Side Includes (SSI) syntax
- Header Based Exploitation