**RFI Nedir?**

Remote File Inclusion (RFI) veya Türkçe adıyla Uzak Dosya Dahil Etme açığı, web uygulamalarının kullanıcıdan aldığı girdiyi yeterince kontrol etmemesi sonucu ortaya çıkan kritik bir güvenlik açığıdır. Bu açıklık sayesinde saldırganlar, kendi sunucularında bulunan kötü amaçlı dosyaları hedef sunucuya dahil ettirip çalıştırabilir.

Genellikle PHP gibi sunucu tarafı dillerde dosya çağırmak için kullanılan include(), require() gibi fonksiyonlar, kullanıcı girdisi ile kontrolsüz şekilde kullanılırsa RFI açığı oluşur.

RFI zafiyeti, web uygulamasının sunucusuna dışarıdan kod enjekte edilmesine ve çalıştırılmasına yol açar. Bu durum, sistemin ele geçirilmesi, veri sızıntısı veya servis dışı bırakma saldırıları gibi ciddi güvenlik risklerine sebep olur.

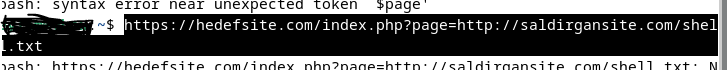
## RFI’nin Çalışma Prensibi

Remote File Inclusion (RFI) açığının temel çalışma mantığı, web uygulamasının kullanıcıdan aldığı bir parametreyi **dosya dahil etme amacıyla** doğrudan kullanmasıdır. Eğer bu parametre herhangi bir filtreleme ya da sınırlama olmadan include, require, include\_once gibi PHP fonksiyonlarıyla kullanılıyorsa, saldırgan bu durumu istismar edebilir.

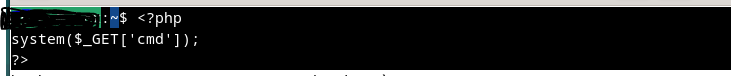
Bu açıklık sayesinde saldırgan, kendi kontrolündeki uzak bir sunucudan bir dosya çağırarak hedef sisteme dahil ettirebilir. Eğer bu dosya içerisinde zararlı PHP kodu varsa, bu kod hedef sunucu üzerinde çalıştırılır

**Örnek PHP Kodu (Zafiyetli): **

-Bu örnekte, page parametresine gelen değer doğrudan include() fonksiyonunda kullanılmıştır. Bu durum ciddi bir güvenlik açığıdır.

**Saldırganın Kullanacağı Örnek URL: **

**-**Bu istek sonucunda, hedef site saldırganın belirttiği uzak adresteki dosyayı indirip çalıştırmaya çalışır.

**Uzak Dosyanın İçeriği (shell.txt): **

**-**Saldırgan bu dosya sayesinde sunucu üzerinde komut çalıştırabilir:

****

**-**Bu örnekte id komutu çalıştırılmış ve sunucunun kullanıcı bilgisi elde edilmiştir.

## RFI’nin Riskleri ve Sonuçları

Remote File Inclusion (RFI) açığı, basit gibi görünse de bir web uygulamasında gerçekleştiğinde **çok ciddi güvenlik ihlallerine yol açabilir**. RFI, saldırganlara yalnızca dışarıdan dosya dahil etme imkânı sunmakla kalmaz, aynı zamanda sistemin tamamen ele geçirilmesine de neden olabilir.

### Başlıca Riskler:

1. **Uzaktan Kod Çalıştırma (RCE)**  
   RFI sayesinde saldırgan, sunucuda sistem komutları çalıştırabilir. Bu durum, sunucunun tamamen kontrol altına alınmasına neden olabilir.
2. **Sunucu Ele Geçirme**  
   Zararlı dosya içerisinde backdoor (arka kapı) varsa, saldırgan sunucuya kalıcı erişim elde edebilir.
3. **Veri Sızıntısı**  
   Saldırganlar, sunucuda çalışan kullanıcıya ait dosyalara, veritabanlarına ya da yapılandırma bilgilerine erişebilir.
4. **Web Sitesinin Tahrip Edilmesi (Defacement)**  
   Saldırganlar web sitesinin içeriğini değiştirebilir veya kötü amaçlı yazılım yayabilir.
5. **Ağ İçi Yayılma**  
   Sunucu ele geçirildikten sonra aynı ağdaki diğer sistemlere de saldırılar düzenlenebilir.
6. **Hukuki ve İtibar Kaybı**  
   RFI saldırıları sonucu kullanıcı verileri sızdırılırsa, kurum ciddi cezalarla karşılaşabilir ve itibar kaybı yaşayabilir.

### Gerçek Hayatta RFI'nin Kullanıldığı Bazı Saldırılar

* Geçmişte birçok açık kaynak CMS (İçerik Yönetim Sistemi) yazılımı, eklenti veya tema yüzünden RFI açıklarına maruz kalmıştır.
* WordPress ve Joomla gibi sistemlerde yüklenen zararlı temalar, saldırganlara uzaktan erişim sağlamıştır.

## Laboratuvar Ortamında RFI Denemesi

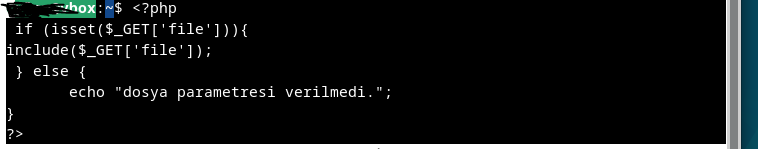
Bu bölümde, RFI zafiyetinin gerçek bir sistem üzerinde nasıl oluşturulacağını ve nasıl sömürüleceğini göstereceğiz. Deneme ortamı olarak Debian kullanıldı.

1: Apache ve PHP Kurulumu: 

Apache'yi başlat: 

#### Adım 2: Zafiyetli PHP Dosyasının Oluşturulması

/var/www/html dizinine gir: 

İçeriği: 

#### 3: Zararlı Dosyanın Hazırlanması

/var/www/html içinde shell.txt oluştur: 

İçeriği: 

#### 4: RFI Açığını Denemek

Tarayıcıda aşağıdaki gibi çağrılır: 

Burada:

* rfi.php: Zafiyetli dosya
* file=...: Uzak dosya çağrısı (bu örnekte yine localhost kullanıldı)
* cmd=ls: Komut çalıştırma isteği

### RFI (Remote File Inclusion) Zafiyetine Karşı Güvenlik Önerileri:

1. **Kullanıcıdan gelen verileri filtrele:**
   * Dosya adı, URL veya parametre olarak gelen tüm veriler mutlaka doğrulanmalı ve kontrol edilmelidir.
2. **Uzak dosya erişimini kapat:**
   * Web uygulamalarında uzak (remote) dosya dahil etmeye izin veren ayarlar devre dışı bırakılmalıdır (örneğin allow\_url\_include PHP'de kapatılır).
3. **Beyaz liste yöntemi kullan:**
   * Kullanıcının sadece izin verilen dosyalar arasından seçim yapmasına izin ver. Örneğin, "hakkimizda.php", "iletisim.php" gibi sabit bir liste oluştur.
4. **Yönlendirme (redirect) ve dahil etme işlemlerini sabit yap:**
   * Dosya isimleri ya da içerikler, kullanıcı tarafından doğrudan gönderilmek yerine sistemin içinde sabit olarak tanımlanmalıdır.
5. **Sunucu izinlerini sıkılaştır:**
   * Web sunucusu, yalnızca ihtiyaç duyulan dosya dizinlerine erişebilmeli. Gereksiz yetkiler azaltılmalı.
6. **Log kayıtları tutulmalı:**
   * Kullanıcı hareketleri loglanarak, şüpheli aktiviteler daha hızlı tespit edilebilir.
7. **Güncel yazılım kullan:**
   * Kullandığın CMS (örneğin WordPress), framework veya kütüphaneler güncel tutulmalı. Bilinen güvenlik açıkları hızla kapatılmalıdır.
8. **Hata mesajlarını sınırla:**
   * Uygulama, kullanıcıya teknik hata mesajları (örneğin "include failed") göstermemelidir. Bu tür bilgiler saldırganlar için ipucu olabilir.

**RFI Açığı İçin Kullanılan Araçlar**

**1. RFI Scanner**  
RFI Scanner, web uygulamalarındaki RFI zafiyetlerini otomatik olarak tespit etmek için kullanılan bir araçtır. Belirli URL parametrelerine kötü amaçlı dosya dahil edilip edilmediğini kontrol eder. Hızlı ve basit testler için tercih edilir.

**2. Fimap**  
Fimap, RFI ve LFI (Local File Inclusion) zafiyetlerini bulmak ve istismar etmek için geliştirilmiş açık kaynaklı bir araçtır. Komut satırı üzerinden çalışır ve geniş bir test senaryosu sunar. İstismar aşamasında dosya içeriklerini okumak veya komut çalıştırmak için kullanılır.

**3. Kadabra**  
Kadabra, RFI açıklarını bulmaya yönelik bir tarama aracıdır. Hem zafiyet tespiti hem de istismar için fonksiyonlar içerir. Python tabanlıdır ve modüler yapısıyla kolayca özelleştirilebilir.

**4. Liffy**  
Liffy, özellikle RFI ve LFI açıklarını tespit etmek için kullanılan basit ve hızlı bir komut satırı aracıdır. Geniş parametre testleri yaparak dosya dahil edilme açıklıklarını arar. Kullanımı kolaydır ancak daha gelişmiş özellikler için sınırlı kalabilir.