

SSRF

SERVER SIDE REQUEST FORGERY NEDİR?

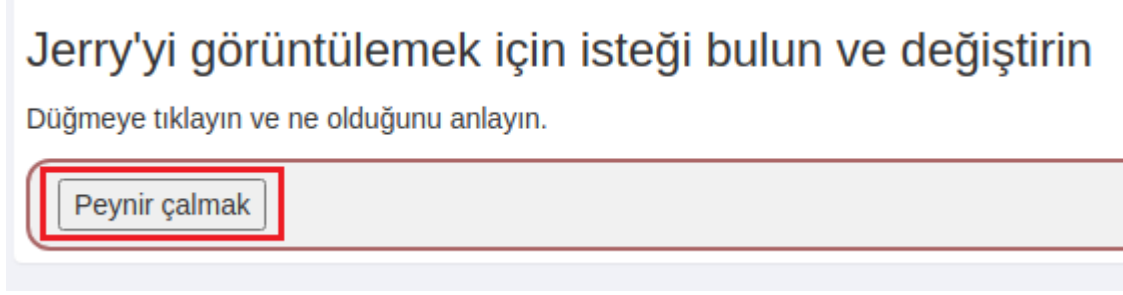
Server Side Request Forgery (SSRF) Zafiyeti Nedir? SSRF Zafiyetinden Nasıl Korunulur?

SSRF (Server Side Request Forgery/Sunucu Tarafı İstek Sahteciliği), bir saldırganın veya saldırganların sunucu tarafında uygulamayı istenmeyen bir konuma istekte bulunmasına neden olmasına izin veren web güvenlik açığıdır.

Sıradan SSRF saldırısında, saldırgan veya saldırganlar sunucunun kuruluşun altyapısındaki sadece dahili hizmetlere bağlantı kurmasına neden olabilmektedir. Öteki durumlardaysa sunucuyu rastgele harici sistemlere bağlanmaya zorlayabilir ve yetkilendirme kimlik bilgileri gibi hassas verileri potansiyel olarak sızdırabilmektedir. Üstelik harici üçüncü taraf sistemlere bağlantılara neden olan bir SSRF istismarı, güvenlik açığı bulunan uygulamayı barındıran kuruluşun geliyormuş gibi görünen kötü niyetli saldırılara neden olabilmektedir.

SSRF Zafiyeti ieren rnek bir uygulama:

OWASP'ın WebGoat adlı laboratuvar ortamının SSRF zafiyeti anlatımında ‘‘Peynir almak’’ butonuna basılmıřtır.



Butona basılmasıyla ařaėıdaki grselde yer aldıėı zere ‘‘Tom ve Jerry Show’’ adlı izgi filmdeki ‘‘Tom’’ karakteri ekrana gelmiřtir.




Sayfanın kaynak kodlarına saė tıklayarak ‘‘İncele’’ye basarak incelemelere bařlanmıřtır.

Jerry'yi görüntülemek için isteği bulun ve

Düğmeye tıklayın ve ne olduğunu anlayın.

Peynir çalmak

Peyniri çalmayı başaramadın!



- Geri Alt+Sol Ok
- İleri Alt+Sağ Ok
- Yeniden Yükle Ctrl+R
- Farklı kaydet... Ctrl+S
- Yazdır... Ctrl+P
- Yayınla...
- Google Lens ile görsel ara
- Bu sayfa için QR kodu oluştur
- Türkçe Diline Çevir
- Sayfa kaynağını görüntüle Ctrl+U
- İncele**

İncelemelere kaynak kodundan devam edilirken şüpheli bir şeye rastlanmamıştır. Resmin tamamına yeni sekmede bakılmıştır ki derinlemesine incelenebilirliği adına URL adresindeki dosya uzantılarına vb. noktalara mutlaka bakılmalıdır.



- Resmi yeni sekmede aç**
- Resmi farklı kaydet...
- Resmi kopyala
- Resim bağlantısını kopyala
- Resmi e-posta ile gönder...
- Erişilebilirlik özelliklerini denetle

localhost:8080/WebGoat/images/tom.png



Resmin "images" adlı dosya yolundan çekildiği tespit edilmiştir. Notunu alırken "Tom ve Jerry Show" adlı çizgi filmin "Tom" karakterinin yanı sıra "Jerry" karakterinin de "images" dizini altında yer alabileceği düşüncesiyle yeni sekmede değiştirilmiştir.

localhost:8080/WebGoat/images/jerry.png



Değişikliğin yapılmasıyla yukardaki görselde yer aldığı üzere başarılı olunduğu saptanmıştır. Erişilebilirliğin söz konusu olduğundan yola çıkılarak butonda bize görüntülenen karakterin resmini SSRF zafiyeti vasıtasıyla değiştirebiliriz.

Denetçi Konsol Hata ayıklayıcı Ağ Stil editörü Performans Bellek Depolama

URL'leri filtrele

Durum	Yöntem	Alan	Dosya
200	GET	localhost:8080	menuItemView.js
200	GET	localhost:8080	MenuModel.js
200	GET	localhost:8080	MenuView.js
200	GET	localhost:8080	PaginationControlView.js
200	GET	localhost:8080	paging_controls.html
200	GET	localhost:8080	polyglot.min.js
200	GET	www.youtube.com	remote.js
200	GET	localhost:8080	require.min.js
200	GET	localhost:8080	SSRF.lesson.lesson
200	GET	localhost:8080	start.mvc
200	POST	localhost:8080	task1

Üst bilgiler Çerezler İstek Yanıt Zamanlamalar Yiğın izi

İstek parametrelerini filtrele

Form verileri Ham

url: "images tom.png"

Yukardaki görselde yer aldığı üzere "İncele" sekmesinin "Ağ" özelliği aracılığıyla resmin görüntülendiği yere gidilir ve değer "tom"dan "jerry" değerine çekilir.

The screenshot shows the network tab of a web browser. The top bar includes icons for Denetçi, Konsol, Hata ayıklayıcı, Ağ (highlighted in red), Stil editörü, and Performans. Below the icons is a search bar for URL'leri filtrele. A table lists several requests with columns for Durum, Yöntem, Alan, and Dosya. The last request is a POST to localhost:8080/task1. Below the table is a 'Yeni istek' section with 'Vazgeç' and 'Gönder' buttons. The 'Yöntem' is set to POST and the 'URL' is http://localhost:8080/WebGoat/SSRF/task1. The 'İstek üst bilgisi:' section shows headers like Host, User-Agent, Accept, etc. The 'İstek gövdesi:' section shows the request body: url=images%2Fjerry.png (highlighted in red).

Değerin değiştirilmesiyle “Gönder” butonuna basılır ve bizlere farklı dönüş sağlandığı aşağıdaki resimde görüldüğü üzere verilmiştir.

The screenshot shows the response tab of the browser's developer tools. The top bar includes icons for Üst bilgiler, Çerezler, İstek, Yanıt (highlighted in blue), Zamanlamalar, and Yiğün izi. Below the icons is a search bar for Özellikleri filtrele. The response is in JSON format. The fields are: lessonCompleted: true (highlighted in red), feedback: "You rocked the SSRFI" (highlighted in red), output: "", assignment: "SSRFTask1", and attemptWasMade: true.

Böylelikle başarılı olunmuştur.

SSRF ZAFİYETİNE KARŞI ALINABİLECEK ÖNLEMLER

SSRF zafiyetine karşı sistem tarafında alınabilecek bazı önlemler vardır.

- Uygulamanızın erişmesi gereken ana bilgisayar adını (DNS adı) veya IP adresini beyaz listeye (whitelist) almalısınız. Beyaz liste (whitelist) yaklaşımı size uymuyorsa ve bir kara listeye (blacklist) güvenmeniz gerekiyorsa, kullanıcı girişini doğru şekilde doğrulamak önemlidir.
- Kullanıcı girişine uygulanan basit kara listeler (blacklist) ve normal ifadeler, SSRF zafiyetini azaltmak için kötü bir yaklaşımdır ve sıklıkla tercih edilir. Bypasslanabilirlik (atlatılabilirlik) durumları söz konusu olduğundan saldırgan veya saldırganlar her zaman farklı metotlara başvuracaktır. Bu ve benzeri söz konusu durumları en aza indirgeyebilmeniz için wildcard DNS çözümlerinin IP bloklandırılmasını yapmalısınız.
- Yanıt verilerinin saldırgana sızmasını önlemek için alınan yanıtın beklendiği gibi olup olmadığından emin olmalısınız. Hiçbir koşulda sunucu tarafından gönderilen istekten gelen ham yanıt gövdesi istemciye teslim edilmemelidir.
- Web adresinizin isteklerinde sadece HTTP veya HTTPS kullanılıyorsa, yalnızca HTTP veya HTTPS URL yapılarına izinlendirme tanımlamalısınız. Kullanılmayan URL şemalarını devre dışı bırakırsanız, saldırgan veya saldırganlar “file:///”, “ftp://” vb. potansiyel tehlikeli yapıları kullanarak isteklerini yapabilmeleri için web uygulamasını kullanamaz hale gelecektir.
- Varsayılan haliyle gelen Memcached, Redis, Elasticsearch ve MongoDB vb. hizmetler kimlik doğrulama gerektirmemektedir. Saldırgan veya saldırganlar, bu ve benzeri hizmetlerden bazılarında herhangi bir kimlik doğrulaması olmadan erişebilmek için SSRF güvenlik açıklığını kullanabilir. Dolayısıyla, hassas bilgilerinizi korumak için yerel ağdaki hizmetler için bile mümkün mertebe her yerde kimlik doğrulamasını etkinleştirmeniz gerekmektedir.