Intro to SSRF

Task 1 What is an SSRF? (Görev 1 SSRF nedir?)

Oda Özeti

Bu odada SSRF'nin ne olduğunu, ne tür etkileri olabileceğini öğrenecek, bazı örnek SSRF saldırılarını görecek, SSRF güvenlik açıklarını nasıl keşfedebileceğinizi, giriş kurallarını nasıl atlatabileceğinizi öğrenecek ve ardından yeni keşfettiğiniz becerilerinizi denemeniz için bir alıştırma yapacağız.

SSRF nedir?

SSRF, Sunucu Tarafı İstek Sahteciliği anlamına gelir. Kötü niyetli bir kullanıcının web sunucusunun saldırganın seçtiği kaynağa ek veya düzenlenmiş bir HTTP isteği yapmasına neden olan bir güvenlik açığıdır.

SSRF Türleri

İki tür SSRF güvenlik açığı vardır; birincisi, verilerin saldırganın ekranına döndürüldüğü normal bir SSRF'dir. İkincisi ise bir SSRF'nin meydana geldiği ancak saldırganın ekranına hiçbir bilginin geri dönmediği Kör SSRF güvenlik açığıdır.

Etkisi ne olacak?

Başarılı bir SSRF saldırısı aşağıdakilerden herhangi biriyle sonuçlanabilir:

- İzinsiz alanlara erişim.
- Müşteri/kurumsal verilere erişim.
- Dahili ağlara ölçeklendirme yeteneği.
- Kimlik doğrulama belirteçlerini/kimlik bilgilerini gösterin.

Sorular

Soru ⇒ SSRF ne anlama geliyor?

Cevap ⇒ Server-Side Request Forgery

Soru ⇒ Normal bir SSRF'nin aksine, diğer tür nedir?

Cevap ⇒ Blind

Task 2 SSRF Examples (Görev 2 SSRF Örnekleri)

Siteyi Görüntüle düğmesine tıkladığınızda, bazı yaygın SSRF örnekleri, bunlardan nasıl yararlanılacağı ve hatta öğrendiklerinizi kullanarak bir SSRF güvenlik açığından yararlanıp yararlanamayacağınızı görmek için bir simülasyon göreceksiniz.

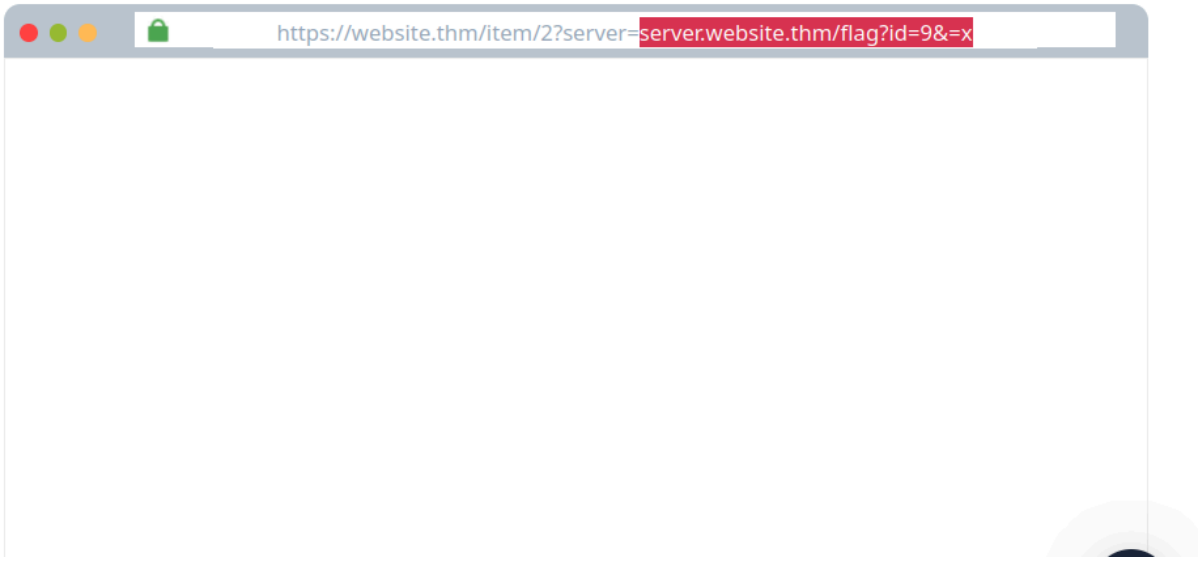
Soru ⇒ SSRF Örnekleri sitesindeki bayrak nedir(İpucu ⇒URL'nin geri kalanını yok saymak için sonuna &x= ekleyin.)?

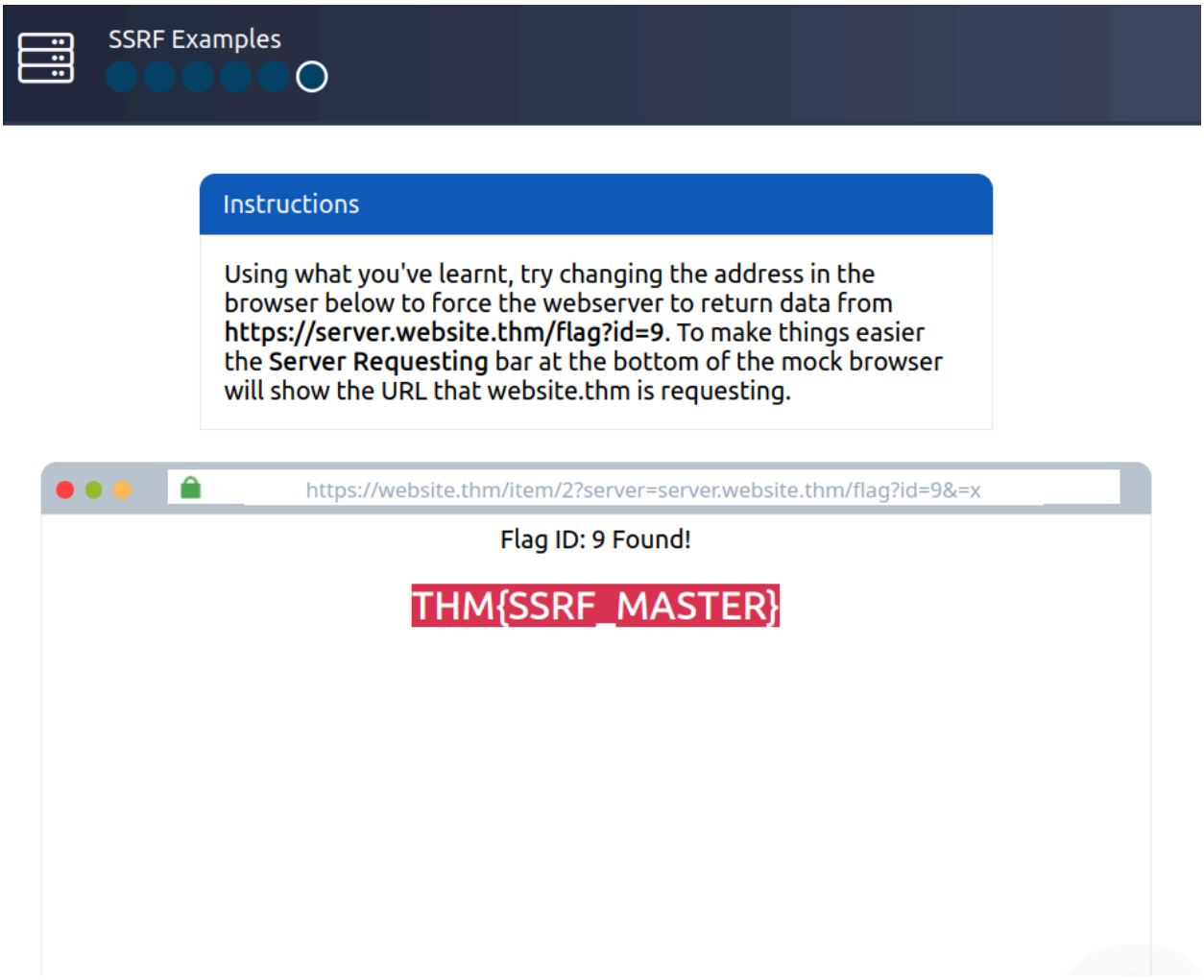
Cevap ⇒ THM{SSRF_MASTER}

SSRF Examples

Instructions

Using what you've learnt, try changing the address in the browser below to force the webserver to return data from **https://server.website.thm/flag?id=9**. To make things easier the **Server Requesting** bar at the bottom of the mock browser will show the URL that website.thm is requesting.





Task 3 Finding an SSRF (Görev 3 Bir SSRF Bulma)

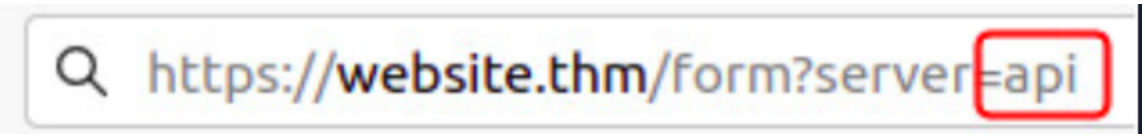
Potansiyel SSRF güvenlik açıkları web uygulamalarında birçok farklı şekilde tespit edilebilir. İşte bakılması gereken dört yaygın yere bir örnek:

Adres çubuğundaki bir parametrede tam bir URL kullanıldığında:

Formdaki gizli bir alan:

```
<form method="post" action="/form">
    <input type="hidden" name="server" value="http://server.website.thm/store">
    <div>Your Name:</div>
    <div><input name="client_name"></div>
    <div>Your Email:</div>
    <div><input name="client_email"></div>
    <div>Your Message:</div>
    <div><textarea name="client_message"></textarea></div>
</form>
```

Yalnızca ana bilgisayar adı gibi kısmi bir URL:



Ya da belki sadece URL'nin yolunu:

Bu örneklerden bazılarını istismar etmek diğerlerine göre daha kolaydır ve bu noktada çalışan bir yük bulmak için çok fazla deneme yanılma yapılması gerekecektir.

Size hiçbir çıktının yansıtılmadığı kör bir SSRF ile çalışıyorsanız, istekleri izlemek için requestbin.com, kendi HTTP sunucunuz veya Burp Suite'in Collaborator istemcisi gibi harici bir HTTP günlük kaydı aracı kullanmanız gerekir.

Soru ⇒ Basit bir gözleme dayanarak, aşağıdaki URL'lerden hangisinin SSRF'ye karşı savunmasız olma olasılığı daha yüksektir?

1. `https://website.thm/index.php`
2. `https://website.thm/list-products.php?categoryId=5325`
3. `https://website.thm/fetch-file.php?fname=242533.pdf&srv=filestorage.cloud.thm&port=8001`
4. `https://website.thm/buy-item.php?itemId=213&price=100&q=2`

Cevap ⇒ 3

Task 4 Defeating Common SSRF Defenses (Görev 4 Yaygın SSRF Savunmalarının Etkisiz Hale Getirilmesi)

SSRF güvenlik açıklarının risklerinin farkında olan daha güvenlik meraklısı geliştiriciler, talep edilen kaynağın belirli kuralları karşıladığından emin olmak için uygulamalarında kontroller uygulayabilir. Bunun için genellikle iki yaklaşım vardır: bir reddetme listesi ya da bir izin listesi.

Deny List

Reddetme Listesi, bir listede belirtilen veya belirli bir kalıpla eşleşen kaynaklar dışında tüm isteklerin kabul edildiği bir listedir. Bir Web Uygulaması, diğer konumlara erişime izin verirken hassas uç noktaları, IP adreslerini veya etki alanlarını genel erişimden korumak için bir reddetme listesi kullanabilir. Erişimi kısıtlamak için belirli bir uç nokta, sunucu performans verilerini veya daha hassas bilgileri içerebilen localhost'tur, bu nedenle localhost ve 127.0.0.1 gibi alan adları bir reddetme listesinde görünecektir. Saldırganlar, 0, 0.0.0.0, 0000, 127.1, 127...*, 2130706433, 017700000001 gibi alternatif localhost referanslarını veya 127.0.0.1.nip.io gibi 127.0.0.1 IP Adresine çözümlenen bir DNS kaydına sahip alt etki alanlarını kullanarak bir Reddetme Listesini atlayabilir.

Ayrıca, bir bulut ortamında, muhtemelen hassas bilgiler de dahil olmak üzere dağıtılan bulut sunucusu için meta veriler içeren 169.254.169.254 IP adresine erişimi engellemek faydalı olacaktır. Bir saldırgan, 169.254.169.254 IP Adresini işaret eden bir DNS kaydıyla kendi etki alanında bir alt etki alanı kaydederek bunu aşabilir.

Allow List

İzin listesi, bir listede görünmedikleri veya belirli bir kalıpla eşleşmedikleri sürece tüm isteklerin reddedildiği yerdir, örneğin bir parametrede kullanılan bir URL'nin https://website.thm ile başlaması gerektiği kuralı gibi. Bir saldırgan, saldırganın alan adında https://website.thm.attackers-domain.thm gibi bir alt alan adı oluşturarak bu kuralı hızlı bir şekilde atlatabilir. Uygulama mantığı artık bu girdiye izin verecek ve bir saldırganın dahili HTTP isteğini kontrol etmesine izin verecektir.

Open Redirect

Yukarıdaki baypaslar işe yaramazsa, saldırganın kolunda bir numara daha vardır, açık yönlendirme. Açık yönlendirme, web sitesi ziyaretçisinin otomatik olarak

başka bir web sitesi adresine yönlendirildiği sunucu üzerindeki bir uç noktadır. Örneğin, https://website.thm/link?url=https://tryhackme.com bağlantısını ele alalım. Bu uç nokta, ziyaretçilerin reklam/pazarlama amacıyla bu bağlantıya kaç kez tıkladığını kaydetmek için oluşturulmuştur. Ancak, yalnızca https://website.thm/ ile başlayan URL'lere izin veren katı kurallara sahip potansiyel bir SSRF güvenlik açığı olduğunu düşünün. Bir saldırgan, dahili HTTP isteğini saldırganın seçtiği bir etki alanına yönlendirmek için yukarıdaki özelliği kullanabilir.

Sorular

Soru ⇒ Katı kuralları atlamak için hangi yöntem kullanılabilir?

Cevap ⇒ Open Redirect

Soru ⇒ Bir bulut ortamında hangi IP adresi hassas veriler içerebilir?

Cevap ⇒ 169.254.169.254

Soru ⇒ Yalnızca belirli girdilere izin vermek için ne tür bir liste kullanılır?

Cevap ⇒ Allow List

Soru ⇒ Belirli bir girişi durdurmak için ne tür bir liste kullanılır?

Cevap ⇒ Deny List

Task 5 SSRF Practical (Görev 5 SSRF Uygulaması)

SSRF hakkında öğrendiklerimizi kurgusal bir senaryoda test edelim.

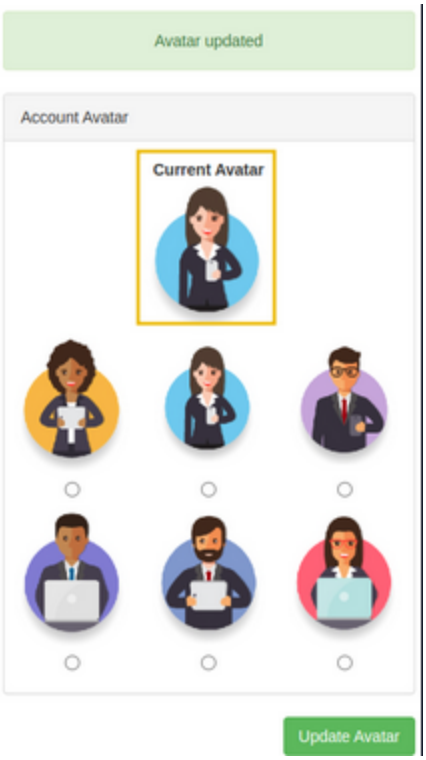
Acme IT Support web sitesine yönelik bir içerik keşif çalışması sırasında iki yeni uç noktayla karşılaştık. Bunlardan ilki /private, bize içeriğin IP adresimizden görüntülenemeyeceğini açıklayan bir hata mesajı veriyor. İkincisi ise /customers/new-account-page adresindeki müşteri hesap sayfasının yeni bir versiyonu ve müşterilerin hesapları için bir avatar seçmelerine olanak tanıyan yeni bir özellik.

Acme IT Support web sitesini başlatmak için Makineyi Başlat düğmesine tıklayarak başlayın. Çalıştırdıktan sonra, https://LAB_WEB_URL.p.thmlabs.com URL adresini ziyaret edin ve ardından bayrağı almak için aşağıdaki talimatları izleyin.

Öncelikle bir müşteri hesabı oluşturun ve oturum açın. Oturum açtıktan sonra, yeni avatar seçimi özelliğini görüntülemek için https://LAB_WEB_URL.p.thmlabs.com/customers/new-account-page adresini ziyaret edin. Avatar formunun sayfa kaynağını görüntülediğinizde, avatar form alanı değerinin resmin yolunu içerdiğini göreceksiniz. Background-image stili, aşağıdaki ekran görüntüsüne göre yukarıdaki DIV öğesinde bunu doğrulayabilir:

```
<div class="col-xs-4">
    <div class="avatar-image" style="background-image: url('/assets/avatars/1.png')"></div>
    <input type="radio" name="avatar" value="assets/avatars/1.png">
</div>
```

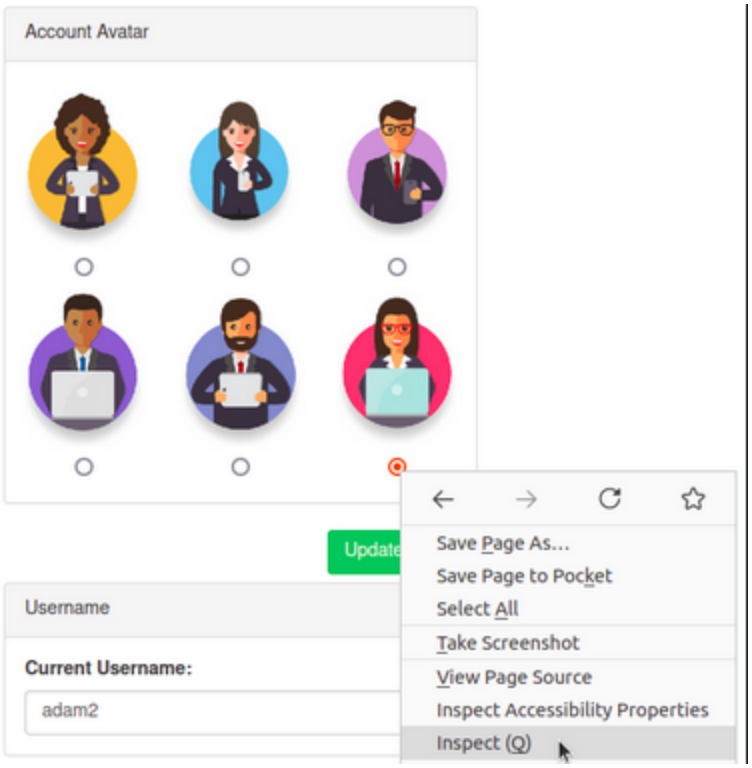
Avatarlardan birini seçip Avatarı Güncelle düğmesine tıklarsanız, formun değiştiğini ve üzerinde o anda seçili olan avatarınızın görüntülendiğini göreceksiniz.



Sayfa kaynağı görüntülendiğinde, mevcut avatarınızın veri URI şeması kullanılarak görüntülendiği ve görüntü içeriğinin aşağıdaki ekran görüntüsüne göre base64 kodlu olduğu görülecektir.

```
<div class="col-xs-6 col-xs-offset-3">
    <div><strong>Current Avatar</strong></div>
    <div class="avatar-image" style="background-image: url(
</div>
```

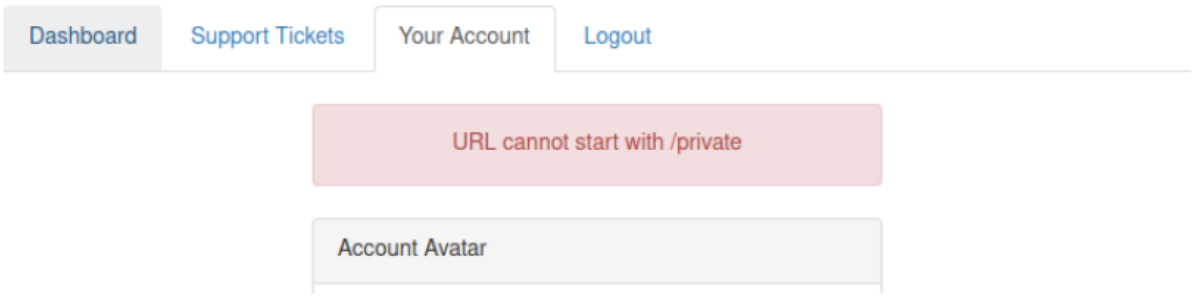
Şimdi isteği tekrar yapmayı deneyelim ancak sunucunun kaynağa erişmesi ve IP adresi engelini aşması umuduyla avatar değerini özel olarak değiştirelim. Bunu yapmak için, öncelikle avatar formundaki radyo düğmelerinden birine sağ tıklayın ve İncele'yi seçin:



Ardından radyo düğmesinin değerini özel olarak düzenleyin:

```
<div class="col-xs-4">
    <div class="avatar-image" style="background-image: url('/assets/avatars/6.png')"></div>
    <input type="radio" name="avatar" value="private">
</div>
```

Düzenlediğiniz avatarı seçtiğinizden emin olun ve ardından Avatarı Güncelle düğmesine tıklayın. Ne yazık ki, web uygulamasının bir reddetme listesi varmış ve /private uç noktasına erişimi engellemiş gibi görünüyor.



Hata mesajından da görebileceğiniz gibi, yol /private ile başlayamaz, ancak endişelenmeyin, bu kuralı atlamak için hala bir numaramız var. İstediğimiz uç noktaya ulaşmak için bir dizin geçişi hilesi kullanabiliriz. Avatar değerini x/../private olarak ayarlamayı deneyin

```
<div class="col-xs-4">
  <div class="avatar-image" style="background-image: url('/assets/avatars/6.png')"></div>
  <input type="radio" name="avatar" value="x/../private">
</div>
```

Şimdi kuralı atladığımızı ve kullanıcının avatarını güncellediğini göreceksiniz. Bu hile işe yarıyor çünkü web sunucusu x/../private isteğini aldığında, ../ dizesinin bir dizini yukarı taşımak anlamına geldiğini biliyor ve şimdi isteği sadece /private olarak çeviriyor.

Avatar formunun sayfa kaynağını görüntülediğinizde, şu anda ayarlanmış olan avatarın artık /private dizinindeki içeriği base64 kodlamasında içerdiğini göreceksiniz, bu içeriği çözün ve aşağıda girebileceğiniz bir bayrak ortaya çıkacaktır.

Soru ⇒ /private dizinindeki bayrak nedir?

Cevap ⇒ THM{YOU_WORKED_OUT_THE_SSRF}