

Burp Suite: Other Modules

Task 1 Introduction (Görev 1 Giriş)

Burp Suite Diğer Modüller odasına hoş geldiniz!

Yaygın olarak bilinen Repeater ve Intruder odalarına ek olarak, Burp Suite daha az bilinen birkaç modül içermektedir. Bunlar bu odanın keşfinin odak noktasını oluşturacaktır.

Dikkatler Kod Çözücü, Karşılaştırmacı, Sıralayıcı ve Düzenleyici araçları üzerinde olacak. Bu araçlar kodlanmış metinlerle işlemleri kolaylaştırır, veri kümelerinin karşılaştırılmasını sağlar, yakalanan belirteçler içindeki rastgeleliğin analizine olanak tanır ve daha sonra tekrar ziyaret etmek isteyebileceğiniz HTTP mesajlarının kopyalarını saklamanıza ve açıklama eklemenize yardımcı olur. Bu görevler basit görünse de, Burp Suite içinde gerçekleştirilmeleri önemli ölçüde zaman kazandırabilir, bu nedenle bu modülleri etkili bir şekilde kullanmayı öğrenmenin önemini vurgulamaktadır.

Sözü daha fazla uzatmadan ilk araç olan Dekoderi inceleyelim.

Yeşil renkli Makineyi Başlat düğmesine basarak göreve bağlı makineyi ve kendi makinenizi kullanmıyorsanız AttackBox'ı (sayfanın üst kısmındaki AttackBox'ı Başlat düğmesini kullanarak) dağıtın.

Cevap Gerekmemektedir.

Task 2 Decoder: Overview (Görev 2 Kod Çözücü: Genel Bakış)

Burp Suite'in Decoder modülü kullanıcıya veri manipülasyon yetenekleri sağlar. Adından da anlaşılacağı üzere, yalnızca bir saldırı sırasında ele geçirilen verilerin kodunu çözmekle kalmaz, aynı zamanda kendi verilerimizi kodlayarak hedefe iletilmek üzere hazırlama işlevi de sağlar. Decoder ayrıca verilerin hashsum'larını

oluşturmamıza olanak tanır ve sağlanan verileri düz metin haline gelene kadar özyinelemeli olarak çözmeye çalışan bir Akıllı Kod Çözme özelliği sağlar (Cyberchef'in "Magic" işlevi gibi).

Dekodere erişmek için, mevcut seçenekleri görüntülemek üzere üst menüden Dekoder sekmesine gidin:



Bu arayüz çok sayıda seçenek sunar.

1. Bu kutu, kodlama veya kod çözme gerektiren verilerin girilmesi veya yapıştırılması için çalışma alanı görevi görür. Burp Suite'in diğer modülleriyle tutarlı olarak, veriler sağ tıklandığında Dekodere Gönder seçeneği aracılığıyla çerçevenin farklı bölümlerinden bu alana taşınabilir.
2. Sağdaki listenin en üstünde, girdiyi metin veya onaltılık bayt değerleri olarak işleme seçeneği vardır.
3. Listede aşağı doğru ilerledikçe, girdiyi kodlamak, kodunu çözmek veya hash etmek için açılır menüler mevcuttur.
4. En sonda bulunan Akıllı Kod Çözme özelliği, girişi otomatik olarak kod çözmeye çalışır.



Giriş alanına veri girdikten sonra, arayüz işlemimizin çıktısını sunmak için kendini kopyalar. Daha sonra aynı seçenekleri kullanarak başka dönüşümler uygulamayı seçebiliriz:

The image shows a screenshot of the Burp Suite interface. It features two input fields. The top field is labeled 'Data' and is empty. The bottom field contains the text '%44%61%74%61'. To the right of each field is a control panel. The top panel has a radio button for 'Text' (selected), a radio button for 'Hex', a help icon, and three dropdown menus labeled 'Decode as ...', 'Encode as ...', and 'Hash ...'. A 'Smart decode' button is also present. The bottom panel has the same controls.

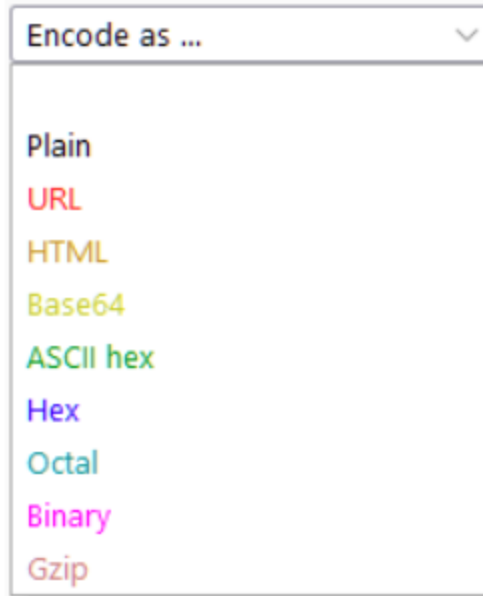
Soru ⇒ Hangi özellik girişin otomatik kodunu çözmeye çalışır?

Cevap ⇒ **Smart decode**

Task 3 Decoder: Encoding/Decoding (Görev 3 Kod Çözücü: Kodlama/Kod Çözme)

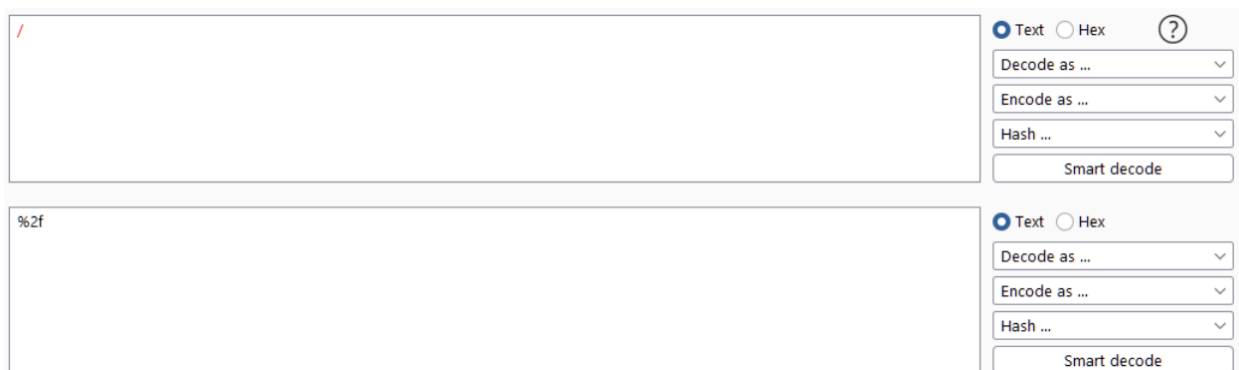
Kod Çözücü ile Kodlama ve Kod Çözme

Şimdi, manuel kodlama ve kod çözme seçeneklerini ayrıntılı olarak inceleyelim. Bunlar ister kod çözme ister kodlama menüsü seçilsin aynıdır:

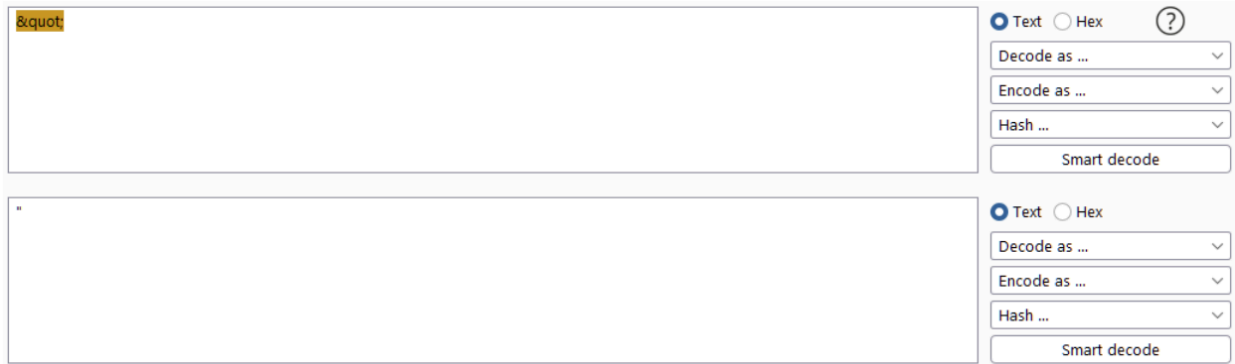


- **Plain** (Sade): Bu, herhangi bir dönüşüm uygulanmadan önceki ham metni ifade eder.
- **URL**: URL kodlaması, bir web isteğinin URL'sindeki verilerin güvenli bir şekilde aktarılmasını sağlamak için kullanılır. Karakterlerin ASCII karakter kodlarıyla onaltılık formatta yer değiştirmesini ve öncesinde bir yüzde sembolü (%) bulunmasını içerir. Bu yöntem, her türlü web uygulaması testi için hayati önem taşır.

Örneğin, ASCII karakter kodu 47 olan ileri eğik çizgi karakterinin (/) kodlanması, onaltılık olarak 2F'ye dönüştürür, böylece URL kodlamasında %2F olur. Kod Çözücü, giriş kutusuna bir ileri eğik çizgi yazarak ve ardından Şu şekilde kodla → URL'yi seçerek bunu doğrulamak için kullanılabilir:



HTML: HTML Varlıkları kodlaması, özel karakterleri bir ve işareti (&) ile değiştirir, ardından onaltılık bir sayı veya kaçılan karaktere bir referans gelir ve noktalı virgül (;) ile biter. Bu yöntem, HTML'de özel karakterlerin güvenli bir şekilde oluşturulmasını sağlar ve XSS gibi saldırıların önlenmesine yardımcı olur. Kod Çözücü'deki HTML seçeneği, herhangi bir karakterin HTML kaçış biçimine kodlanmasına veya yakalanan HTML varlıklarının kodunun çözülmesine olanak tanır. Örneğin, daha önce tartışılan bir tırnak işaretinin kodunu çözmek için, kodlanmış versiyonu girin ve Kod Çöz → HTML seçeneğini seçin:



- **Base64:** Yaygın olarak kullanılan bir kodlama yöntemi olan Base64, herhangi bir veriyi ASCII uyumlu bir biçime dönüştürür. Kaputun altındaki işleyiş bu aşamada çok önemli değildir; ancak ilgilenen kişiler altta yatan matematiği burada bulabilirler.
- **ASCII Hex:** Bu seçenek verileri ASCII ve onaltılık gösterimler arasında değiştirir. Örneğin, "ASCII" kelimesi "4153434949" onaltılık sayısına dönüştürülebilir. Her karakter sayısal ASCII gösteriminden onaltılık sayıya dönüştürülür.
- **Hex, Octal ve Binary:** Bu kodlama yöntemleri yalnızca sayısal girdilere uygulanır ve ondalık, onaltılık, sekizlik (sekiz tabanı) ve ikili gösterimler arasında dönüştürme yapar.
- **Gzip:** Gzip, verileri sıkıştırarak tarayıcı aktarımından önce dosya ve sayfa boyutlarını azaltır. Daha hızlı yükleme süreleri, SEO puanlarını artırmak ve kullanıcı rahatsızlığından kaçınmak isteyen geliştiriciler için oldukça arzu edilir. Decoder, genellikle geçerli ASCII/Unicode olmamasına rağmen gzip verilerinin manuel olarak kodlanmasını ve kodunun çözülmesini kolaylaştırır. Örneğin:

Testing Gzip

Decode as ...

Encode as ...

Hash ...

Smart decode

Decode as ...

Encode as ...

Hash ...

Smart decode

Bu yöntemler istiflenebilir. Örneğin, bir ifade ("Burp Suite Decoder") ASCII Hex'e ve ardından octal'e dönüştürülebilir:

Burp Suite Decoder

Decode as ...

Encode as ...

Hash ...

Smart decode

42757270205375697465204465636f646572

Decode as ...

Encode as ...

Hash ...

Smart decode

35526336111075346072246615353744f2356654

Decode as ...

Encode as ...

Hash ...

Smart decode

Bu yöntemler bir arada kullanıldığında, kodladığımız veya kodunu çözdüğümüz veriler üzerinde önemli ölçüde kontrol sahibi olmamızı sağlar.

Her kodlama/kod çözme yöntemi renk kodludur ve uygulanan dönüşümün hızlı bir şekilde tanımlanmasını sağlar.

Hex Format

Verileri ASCII formatında girmek faydalı olsa da, bayt bayt giriş düzenlemesinin gerekli olduğu zamanlar vardır. Bu noktada, kod çözme seçeneklerinin üzerinde seçilebilen "Hex View" yararlı olmaktadır:

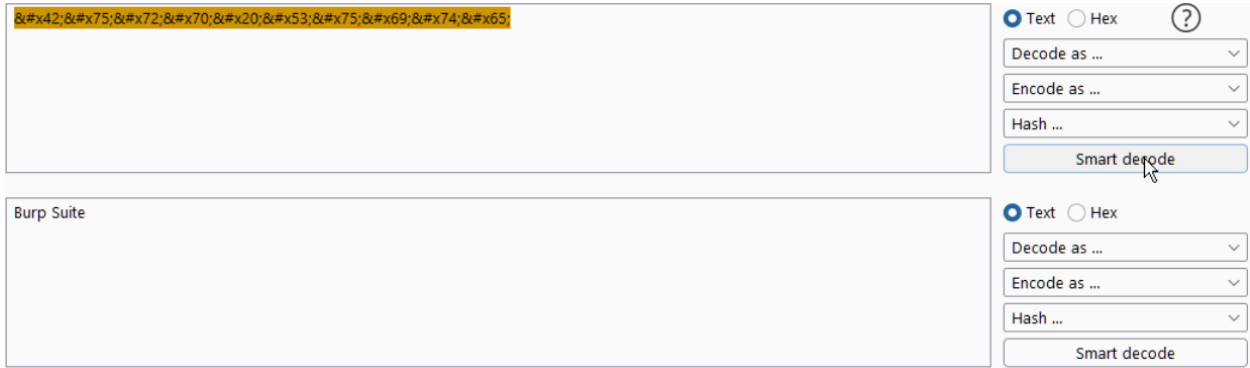


Bu özellik, ikili dosyalar veya ASCII olmayan diğer verilerle çalışırken hayati bir araç olan verilerimizi onaltılık bayt biçiminde görüntülememizi ve değiştirmemizi sağlar.

Smart Decode

Son olarak, Akıllı kod çözme seçeneğimiz var. Bu özellik kodlanmış metni otomatik olarak çözmeye çalışır. Örneğin, Burp Suite otomatik olarak HTML kodlu olarak tanınır ve buna göre kodu çözülür:

Burp Suite



Mükemmel olmasa da, bu özellik bilinmeyen veri parçalarının kodunu çözmek için hızlı bir çözüm olabilir.

Sorular;

İfadeyi Base64 ile kodlayın: [Let's Start Simple](#)

Soru ⇒ Bu metnin base64 kodlu sürümü nedir?

Cevap ⇒ [TGV0J3MgU3RhcnQgU2ltcGx](#)

URL Bu verilerin kodunu çözün: [%4e%65%78%74%3a%20%44%65%63%6f%64%69%6e%67](#)

Soru ⇒ Döndürülen düz metin nedir?

Cevap ⇒ [Next: Decoding](#)

Bu verilerin kodunu çözmek için Akıllı kod çözmeyi kullanın:

`%34%37`

Soru ⇒ Deşifre edilen metin nedir?

Cevap ⇒ 47

Bu ifadeyi kodlayın: **Encoding Challenge**

Base64 kodlaması ile başlayın. Bunun çıktısını alın ve ASCII Hex'e dönüştürün. Son olarak, hex dizesini octal olarak kodlayın.

Soru ⇒ Son dize nedir?

Cevap ⇒ 24034214a720270024142d541357471232250253552c1162d1206c

Task 4 Decoder: Hashing (Görev 4 Kod Çözücü: Hashing)

HashKodlama/Kod Çözme işlevine ek olarak, Decoder ayrıca verilerimiz için hashsum üretme yeteneği de sunar.

Theory

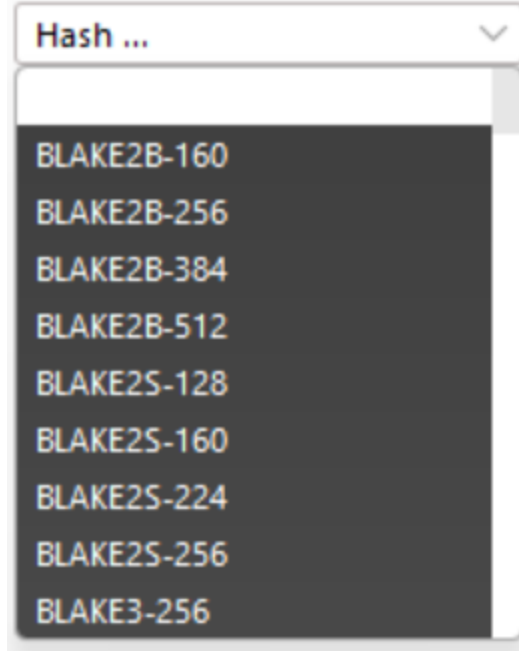
Hashing, verileri benzersiz bir imzaya dönüştüren tek yönlü bir işlemdir. Bir fonksiyonun hashing algoritması olarak nitelendirilebilmesi için ürettiği çıktının geri döndürülemez olması gerekir. Yetkin bir hash algoritması, her veri girişinin tamamen benzersiz bir hash üretmesini sağlar. Örneğin, "MD5sum" metni için bir hashsum üretmek üzere MD5 algoritması kullanıldığında 4ae1a02de5bd02a5515f583f4fca5e8c sonucu elde edilir. "MD5SUM" için aynı algoritmayı kullanmak, girdinin yakın benzerliğine rağmen tamamen farklı bir karma verir: 13b436b09172400c9eb2f69fbd20adad. Bu nedenle, hash'ler dosya ve belgelerin bütünlüğünü doğrulamak için yaygın olarak kullanılır, çünkü dosyada yapılan küçük bir değişiklik bile hash toplamını önemli ölçüde değiştirir.

Not: MD5 algoritması kullanımdan kaldırılmıştır ve çağdaş uygulamalar için kullanılmamalıdır.

Hashing in Decoder (Kod Çözücüde Hashing)

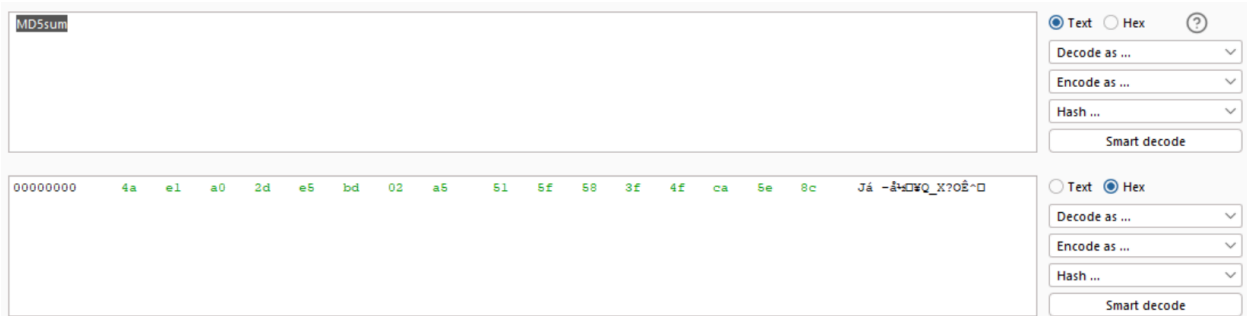
Kod çözücü, veriler için doğrudan Burp Suite içinde karma toplamalar oluşturmamızı sağlar; daha önce tartıştığımız kodlama / kod çözme seçeneklerine benzer şekilde

çalışır. Özellikle, Hash açılır menüsüne tıklıyoruz ve listeden bir algoritma seçiyoruz:



Not: Bu liste kodlama/kod çözme algoritmalarından çok daha uzundur - mevcut birçok hashing algoritmasını görmek için kaydırmaya değer.

Daha önceki örneğimize devam ederek, giriş kutusuna "MD5sum" girelim, ardından "MD5" bulana kadar listeyi aşağı kaydıralım. Bunu uygulamak bizi otomatik olarak Hex görünümüne götürür:



Bir hashing algoritmasının çıktısı saf ASCII/Unicode metin vermez. Bu nedenle, algoritmanın çıktısını onaltılık bir dizeye dönüştürmek gelenekseldir; bu, aşına olabileceğiniz "hash" biçimidir.

İlk örneğimizdeki düzgün hex dizesini oluşturmak için hashsum'a bir "ASCII Hex" kodlaması uygulayarak bunu tamamlayalım.

İşte tüm süreç:

The screenshot shows the hashsum tool interface with three rows of input and output. The first row shows 'MD5sum' as input and a long hex string as output. The second row shows a hex string as input and a long hex string as output. The third row shows a hex string as input and a long hex string as output. The interface includes buttons for 'Text', 'Hex', 'Decode as ...', 'Encode as ...', 'Hash ...', and 'Smart decode'.

sorular

Soru ⇒ Kod Çözücüyü kullanarak, ifadenin SHA-256 hashsum'u nedir: [Let's get Hashing!](#)

Bu sorunun cevabı için bunu bir ASCII Hex dizesine dönüştürün.

Cevap ⇒

[6b72350e719a8ef5af560830164b13596cb582757437e21d1879502072238abe](#)

Soru ⇒ İfadenin MD4 hashsum'unu oluşturun: [Insecure Algorithms](#)

Göndermeden önce bunu base64 (ASCII Hex değil) olarak kodlayın.

Cevap ⇒ [TcV4QGZZN7y7lwYFRMMoeA==](#)

Bağlam içi bir örneğe bakalım:

İlk olarak, bu göreve ekli dosyayı indirin.

Not: Bu dosya, konuşlandırılmış sanal makineden wget

http://MACHINE_IP:9999/AlteredKeys.zip ile de indirilebilir - AttackBox

kullanıyorsanız bunu yararlı bulabilirsiniz.

Şimdi aşağıdaki problem spesifikasyonunu okuyun:

"Şakacının biri SSH anahtarımıla oynamış! Dizinde dört anahtar var ve hangisinin gerçek olduğu hakkında hiçbir fikrim yok. Anahtarımın MD5 hashsum değeri 3166226048d6ad776370dc105d40d9f8 - bunu benim için bulabilir misiniz?"

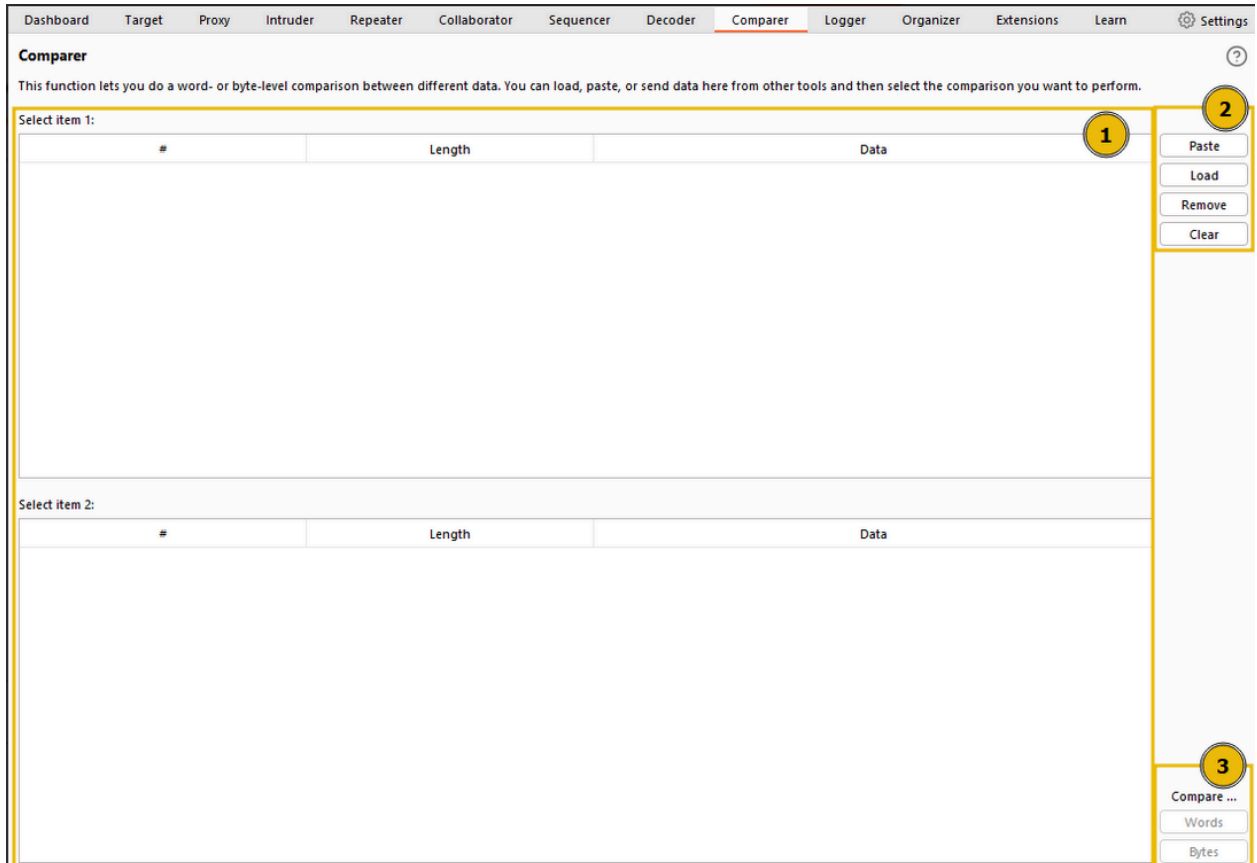
Soru ⇒ Doğru anahtar adı nedir(İpucu ⇒ Anahtarların her birini kopyalayıp Decoder'a yapıştırın ve MD5 algoritması ile tek tek hashleyin. MD5 hashsum değeri soruda belirtilene eşit olan anahtarı arıyorsunuz. Anahtarların sonundaki son satırı kaldırmadığınızdan emin olun!)?

Cevap ⇒ **key3**

Task 5 Comparer: Overview (Görev 5 Karşılaştırma: Genel Bakış)

Karşılaştırıcı, adından da anlaşılacağı gibi, iki veri parçasını ASCII sözcükleriyle veya baytlarla karşılaştırmamızı sağlar.

Önce arayüze bir göz atalım:

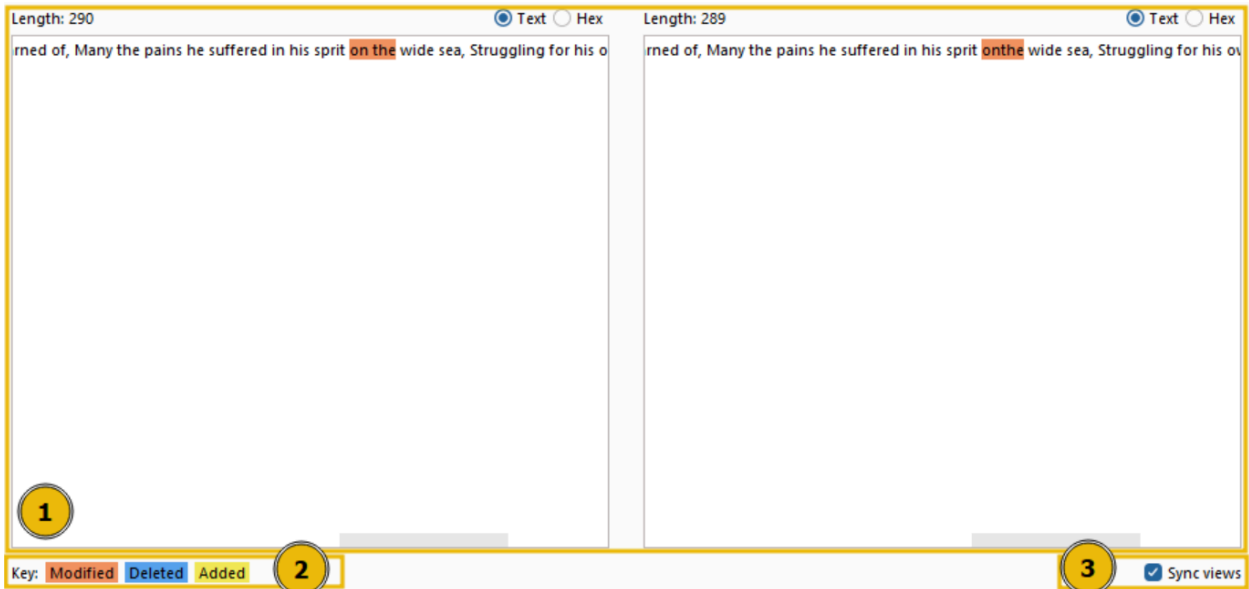


Arayüz üç ana bölüme ayrılabilir:

1. Sol tarafta, karşılaştırılacak öğeleri görüyoruz. Comparer'a veri yüklediğimizde, bu tablolarda satırlar olarak görünür. Karşılaştırmak için iki veri kümesi seçiyoruz.
2. Sağ üstte, panodan veri yapıştırma (Yapıştır), bir dosyadan veri yükleme (Yükle), geçerli satırı kaldırma (Kaldır) ve tüm veri kümelerini temizleme (Temizle) seçeneklerimiz vardır.
3. Son olarak, sağ altta, veri kümelerimizi sözcüklere veya baytlara göre karşılaştırmayı seçebiliriz. Başlangıçta bu düğmelerden hangisini seçtiğiniz önemli değildir çünkü bu daha sonra değiştirilebilir. Bunlar, seçilen verileri karşılaştırmaya hazır olduğumuzda tıkladığımız düğmelerdir.

Çoğu Burp Suite modülünde olduğu gibi, sağ tıklayıp Comparer'a Gönder'i seçerek diğer modüllerden Comparer'a veri yükleyebiliriz.

Karşılaştırmak için en az 2 veri kümesi ekledikten ve Kelimeler ya da Baytlar seçeneğine bastıktan sonra, bir açılır pencere bize karşılaştırmayı gösterir:



Bu pencere de üç farklı bölüme sahiptir:

1. Karşılaştırılan veriler pencerenin çoğunu kaplar; metin ya da hex formatında görüntülenebilir. İlk format, önceki pencerede kelime veya bayt olarak

karşılaştırmayı seçmemize bağlıdır, ancak bu, karşılaştırma kutularının üzerindeki düğmeler kullanılarak geçersiz kılınabilir.

2. Karşılaştırma anahtarı sol altta yer alır ve iki veri kümesi arasında hangi renklerin değiştirilen, silinen ve eklenen verileri temsil ettiğini gösterir.
3. Görünümleri senkronize et onay kutusu pencerenin sağ alt tarafındadır. Seçildiğinde, her iki veri setinin formatlarının senkronize edilmesini sağlar. Başka bir deyişle, bunlardan birini Hex görünümüne değiştirirseniz, diğeri eşleşecek şekilde ayarlanacaktır.

Pencere başlığı bulunan toplam farklılık sayısını gösterir.

Soru

Cevap Gerekmemektedir.

Task 6 Comparer: Example (Görev 6 Karşılaştırmacı: Örnek)

Artık Comparer'ın ne yaptığını biliyoruz, ancak bu nasıl yararlı olabilir?

İki (potansiyel olarak çok büyük) veri parçasını hızlı bir şekilde karşılaştırabilmenin kullanışlı olabileceği birçok durum vardır.

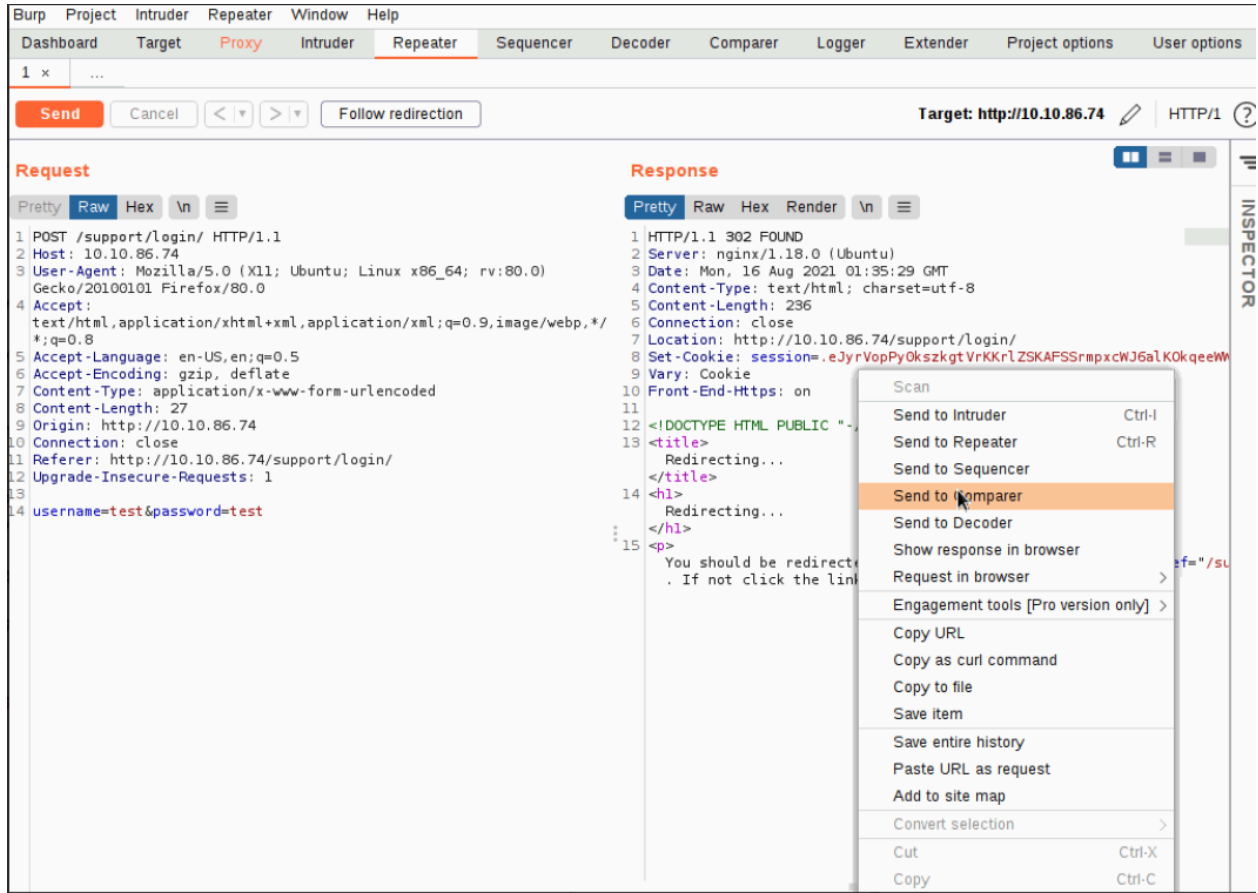
Örneğin, Intruder ile bir oturum açma zorlaması veya kimlik bilgisi doldurma saldırısı gerçekleştirirken, farkların nerede olduğunu ve farkların başarılı bir oturum açmaya işaret edip etmediğini görmek için farklı uzunluklara sahip iki yanıtı karşılaştırmak isteyebilirsiniz.

Pratik Örnek

1. http://MACHINE_IP/support/login adresine gidin

Geçersiz bir kullanıcı adı ve parola ile oturum açmayı deneyin - Burp Proxy'deki isteği yakalayın.

2. Ctrl + R (veya Mac eşdeğeri) tuşlarını kullanarak veya Proxy modülünde isteğe sağ tıklayıp Tekrarlayıcıya Gönder'i seçerek isteği Tekrarlayıcıya gönderin.
3. İsteği gönderin, ardından yanıtı sağ tıklayın ve Karşılaştırmacıya Gönder'i seçin.



4. Tekrarlayıcı sekmesinde, kimlik bilgilerini şu şekilde değiştirin:

Username : `support_admin`

Password : `w58ySK4W`

İsteği tekrar gönderin, ardından yeni yanıt Comparer'a iletin.

İki yanıt kelime kelime karşılaştırın. Temel farklılıkları belirleyebilir misiniz?

Cevap Gerekmemektedir.

Task 7 Sequencer: Overview (Görev 7 Sıralayıcı: Genel Bakış)

Sequencer, "belirteçlerin" entropisini veya rastgeleliğini değerlendirmemizi sağlar. Belirteçler bir şeyi tanımlamak için kullanılan dizelerdir ve ideal olarak kriptografik olarak güvenli bir şekilde oluşturulmalıdır. Bu belirteçler, oturum çerezleri veya

form gönderimlerini korumak için kullanılan Siteler Arası İstek Sahteciliği (CSRF) belirteçleri olabilir. Bu belirteçler güvenli bir şekilde oluşturulmamışsa, teorik olarak, gelecek belirteç değerlerini tahmin edebiliriz. Örneğin, söz konusu token parola sıfırlamaları için kullanılıyorsa, bunun sonuçları önemli olabilir.

Sequencer arayüzüne bakarak başlayalım:

The screenshot shows the Burp Suite Sequencer interface. The top navigation bar includes tabs for Dashboard, Target, Proxy, Intruder, Repeater, Collaborator, Sequencer (selected), Decoder, Comparer, and Logger. Below the navigation bar, there are sub-tabs for Live capture, Manual load, and Sequencer settings. The main content area is divided into two sections:

Select live capture request

Send requests here from other tools to configure a live capture. Select the request to use, configure the other options below, then click "Start live capture".

On the left, there are buttons for "Remove" and "Clear". To the right is a table with columns: # ^, Host, and Request. The table is currently empty. Below the table is a "Start live capture" button.

Token location within response

Select the location in the response where the token appears.

There are three radio button options:

- ☐ Cookie: [dropdown menu]
- ☐ Form field: [dropdown menu]
- ☒ Custom location: [text input field]

A "Configure" button is located to the right of the Custom location input field.

Sequencer ile token analizi yapmanın iki ana yolu vardır:

- **Live Capture** (Canlı Yakalama): Bu daha yaygın bir yöntemdir ve Sequencer için varsayılan alt sekmedir. Canlı yakalama, analiz için Sequencer'a bir belirteç oluşturacak bir istek iletmemizi sağlar. Örneğin, sunucunun bir çerezle yanıt vereceğini bilerek Sequencer'a bir oturum açma uç noktasına POST isteği iletmek isteyebiliriz. İletilen istek ile Sequencer'a canlı bir yakalama başlatması talimatını verebiliriz. Daha sonra otomatik olarak aynı isteği binlerce kez yapacak ve üretilen token örneklerini analiz için saklayacaktır. Yeterli sayıda örnek topladıktan sonra Sequencer'ı durdurur ve yakalanan tokenları analiz etmesine izin veririz.

- **Manual Load (Manuel Yükleme):** Bu, önceden oluşturulmuş token örneklerinin bir listesini analiz için doğrudan Sequencer'a yüklememizi sağlar. Manuel Yüklemeyi kullanmak, hedefimize binlerce istek yapmamız gerekmediği anlamına gelir; bu da gürültülü ve yoğun kaynak kullanımı anlamına gelebilir. Bununla birlikte, önceden oluşturulmuş belirteçlerin büyük bir listesine sahip olmamızı gerektirir.

Bu odada canlı çekimlere odaklanacağız.

Soru ⇒ Sequencer neyi değerlendirmemize izin veriyor?

Cevap ⇒ **Entropy**

Task 8 Sequencer: Live Capture (Görev 8 Sequencer: Canlı Yakalama)

Harika, yönetici giriş formunda kullanılan anti-bruteforce belirteci üzerinde entropi analizi için Sequencer'ın canlı yakalamasını kullanma sürecine dalalım.

İlk olarak, Proxy'de `http://MACHINE_IP/admin/login/` adresine bir istek yakalayın. İsteğe sağ tıklayın ve Sıralayıcıya Gönder'i seçin.

"Yanıt İçindeki Token Konumu" bölümünde Çerez, Form alanı ve Özel konum arasında seçim yapabiliriz. Bu durumda `loginToken`'ı test ettiğimiz için, "Form alanı" radyo düğmesini seçin ve açılır menüden `loginToken`'ı seçin:



Bu durumda, diğer tüm seçenekleri varsayılan değerlerinde güvenle bırakabiliriz. Bu yüzden, Canlı yakalamayı başlat düğmesine tıklayın.

Canlı yakalamanın devam ettiğini belirten ve şu ana kadar yakalanan token sayısını gösteren yeni bir pencere açılacaktır. Yeterli sayıda token yakalanana kadar bekleyin (yaklaşık 10.000 yeterli olacaktır); ne kadar çok tokenımız olursa analizimiz o kadar hassas olacaktır.

Yaklaşık 10.000 token yakalandığında, Duraklat'a tıklayın ve ardından Şimdi analiz et düğmesini seçin:



Yakalamayı Durdurmayı da seçebileceğimizi unutmamak önemlidir. Ancak, duraklatmayı seçerek, raporda belirtecin entropisini doğru bir şekilde hesaplamak için yeterli örnek yoksa yakalamaya daha sonra devam etme seçeneğini koruyoruz.

Analizin periyodik olarak güncellenmesini isteseydik, "Otomatik analiz" onay kutusunu da seçebilirdik. Bu seçenek Burp'e her 2000 istekten sonra entropi analizi yapmasını söyleyerek, Sequencer'a daha fazla örnek yüklendikçe giderek daha doğru hale gelecek sık güncellemeler sağlar.

Bu noktada, yakalanan belirteçleri daha sonra analiz etmek üzere kopyalamayı veya kaydetmeyi seçebileceğimizi de belirtmek gerekir.

Şimdi analiz et düğmesine tıklandığında Burp, token'ın entropisini analiz edecek ve bir rapor oluşturacaktır.

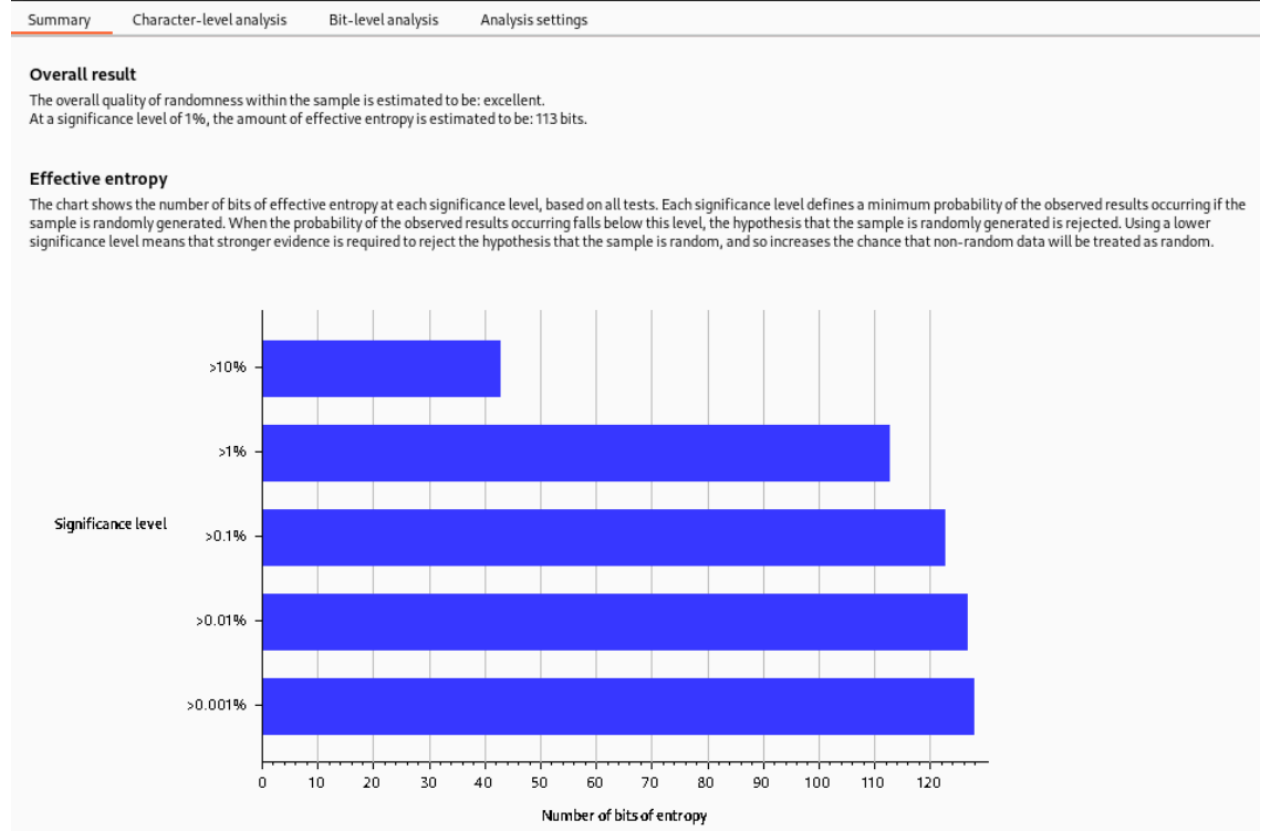
Soru ⇒ Rastgeleliğin genel kalitesinin ne olduğu tahmin edilmektedir?

Cevap ⇒ **excellent**

Task 9 Sequencer: Analysis (Görev 9 Sıralayıcı: Analiz)

Artık tokenımızın entropi analizi için bir raporumuz olduğuna göre, şimdi onu analiz etme zamanı!

Oluşturulan entropi analizi raporu dört ana bölüme ayrılmıştır. Bunlardan ilki Sonuçların Özeti'dir. Özet bize aşağıdakileri vermektedir:



- **Overall result**(Genel sonuç): Bu, token oluşturma mekanizmasının güvenliğine ilişkin geniş bir değerlendirme sağlar. Bu durumda, entropi seviyesi belirteçlerin muhtemelen güvenli bir şekilde üretildiğini gösterir.
- **Effective entropy** (Etkin entropi): Bu, belirteçlerin rastgeleliğini ölçer. Etkin 117 bit entropi nispeten yüksektir, bu da belirteçlerin yeterince rastgele olduğunu ve dolayısıyla tahmin veya kaba kuvvet saldırılarına karşı güvenli olduğunu gösterir.
- **Reliability** (Güvenilirlik): 1'lik anlamlılık düzeyi, sonuçların doğruluğuna %99 oranında güven duyulduğu anlamına gelmektedir. Bu güven düzeyi oldukça yüksektir ve etkin entropi tahmininin doğruluğu konusunda güvence sağlamaktadır.
- **Sample** (Örnek): Bu, belirteçlerin sayısı ve özellikleri de dahil olmak üzere entropi test işlemi sırasında analiz edilen belirteç örnekleri hakkında ayrıntılar

sağlar.

Özet rapor genellikle token oluşturma sürecinin güvenliğini değerlendirmek için yeterli bilgi sağlarken, bazı durumlarda daha fazla araştırmanın gerekli olabileceğini unutmamak önemlidir. Karakter düzeyinde ve bit düzeyinde analiz, özellikle özet sonuçlar potansiyel endişelere yol açtığında, belirteçlerin rastgeleliği hakkında daha ayrıntılı bilgiler sağlayabilir.

Entropi raporu token üretim mekanizmasının güvenliğine dair güçlü bir gösterge sunsa da daha kesin kanıtlara ihtiyaç vardır. Diğer faktörler de tokenların güvenliğini etkileyebilir ve olasılık ve istatistiğin doğası gereği her zaman bir miktar belirsizlik söz konusudur. Bununla birlikte, %1 anlamlılık düzeyine sahip 117 bitlik etkin entropi, sağlam bir şekilde güvenli bir token oluşturma sürecine işaret etmektedir.

Soru ⇒ Bir sonraki göreve geçmek için aşağıdaki kutucuğa tıklayın.

Cevap ⇒ **Cevap Gerekmemektedir.**

Task 10 Organizer: Overview (Görev 10 Düzenleyici: Genel Bakış)

Burp Suite'in Organizer modülü, daha sonra tekrar ziyaret etmek isteyebileceğiniz HTTP isteklerinin kopyalarını saklamanıza ve açıklama eklemenize yardımcı olmak için tasarlanmıştır. Bu araç, sızma testi iş akışınızı düzenlemek için özellikle yararlı olabilir. İşte temel özelliklerinden bazıları:

- Daha sonra incelemek istediğiniz talepleri saklayabilir, daha önce ilginç olarak tanımladığınız talepleri kaydedebilir veya daha sonra bir rapora eklemek istediğiniz talepleri kaydedebilirsiniz.
- Proxy veya Repeater gibi diğer Burp Modüllerinden Burp Organizer'a HTTP istekleri gönderebilirsiniz. Bunu, isteğe sağ tıklayıp Organizer'a Gönder'i seçerek veya varsayılan kısayol tuşu Ctrl + O'yu kullanarak yapabilirsiniz. Organizer'a gönderdiğiniz her HTTP isteği, Organizer'a gönderdiğiniz noktada kaydedilen orijinal isteğin salt okunur bir kopyasıdır.

The screenshot displays the Burp Suite interface with the **Proxy** tab selected. The **HTTP history** table shows the following data:

#	Host	Method	URL	Params	Edited	Status code	Length	MIME type	Extension	Title	Comment	TLS	IP
8	https://10-10-148-3.p.thml...	GET	/assets/js/scripts.js			200	601	script	js			✓	54.246.5.175
7	https://10-10-148-3.p.thml...	GET	/assets/js/bootstrap.bundle.min...			200	78727	script	js			✓	54.246.5.175
1	https://10-10-148-3.p.thml...	GET	/				6802	HTML		Bastion Hosting		✓	54.246.5.175

A context menu is open for the selected request (ID 1), showing options such as **Add to scope**, **Send to Intruder**, **Send to Repeater**, **Send to Sequencer**, **Send to Organizer** (highlighted), **Send to Comparer (request)**, **Send to Comparer (response)**, **Show response in browser**, **Request in browser**, **Engagement tools [Pro version only]**, **Show new history window**, **Add comment**, **Highlight**, **Delete item**, **Clear history**, **Copy URL**, **Copy as curl command (bash)**, **Copy links**, **Save item**, and **Proxy history documentation**.

The **Request** pane shows the raw HTTP request details, including headers like **Host**, **Sec-Ch-Ua**, **Sec-Ch-Ua-Mobile**, **Sec-Ch-Ua-Platform**, **Upgrade-Insecure-Requests**, **User-Agent**, **Accept**, **Sec-Fetch-Site**, **Sec-Fetch-Mode**, **Sec-Fetch-User**, **Sec-Fetch-Dest**, **Accept-Encoding**, **Accept-Language**, and **Connection**.

The **Inspector** pane shows the **Request attributes**, **Request headers**, and **Response headers**.

- İstekler, istek dizin numarası, isteğin yapıldığı zaman, iş akışı durumu, isteğin gönderildiği Burp aracı, HTTP yöntemi, sunucu ana bilgisayar adı, URL dosya yolu, URL sorgu dizesi, istekteki parametre sayısı, yanıtın HTTP durum kodu, yanıtın bayt cinsinden uzunluğu ve aldığınız notlar gibi sütunları içeren bir tabloda saklanır.

The screenshot shows the Burp Suite interface with the 'Organizer' tab selected. At the top, there is a navigation bar with tabs: Dashboard, Target, Proxy, Intruder, Repeater, Collaborator, Sequencer, Decoder, Comparer, Logger, Organizer (active), Extensions, Learn, and Settings. Below the navigation bar is a filter bar that says 'Filter: Showing all items'. The main area displays a table of HTTP requests:

#	Time	Status	Tool	Method	Host	Path	Query	Param count	Status code	Length	Notes
1	18:26:03 27 Jul 2023	→ New	Proxy	GET	10-10-148-3.p.thmla...	/		0	200	6802	This is the home
2	18:26:04 27 Jul 2023	→ New	Proxy	GET	10-10-148-3.p.thmla...	/assets/js/scripts.js		0	200	601	This is the script
3	18:33:19 27 Jul 2023	→ New	Repeater	GET	10-10-148-3.p.thmla...	/		0	200	6802	

Below the table, there are two panels: 'Request' and 'Response'. The 'Request' panel shows the raw HTTP request details, including headers like 'Host: 10-10-148-3.p.thmlabs.com', 'User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/115.0.5790.102 Safari/537.36', and 'Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8'. The 'Response' panel shows the raw HTTP response details, including headers like 'HTTP/1.1 200 OK', 'Server: nginx/1.14.0 (Ubuntu)', 'Date: Thu, 27 Jul 2023 10:26:05 GMT', 'Content-Type: text/html; charset=utf-8', 'Connection: close', 'Front-End-Https: on', 'Content-Length: 6613', and the body content starting with '<!DOCTYPE html>' and '<html lang=en>'. To the right of these panels is a 'Notes' panel with the text 'This is the homepage'.

İstek ve yanıtı görüntülemek için:

1. Herhangi bir Organizör öğesine tıklayın.
2. Talep ve yanıtın her ikisi de salt okunurdur. Talep veya yanıt içinde arama yapabilir, talebi seçebilir ve ardından talebin altındaki arama çubuğunu kullanabilirsiniz.

#	Time	Status	Tool	Method	Host	Path	Query	Param count	Status code	Length	Notes
1	18:26:03 27 Jul 2023	New	Proxy	GET	10-10-148-3.p.thmla...	/		0	200	6802	This is the homep
2	18:26:04 27 Jul 2023	New	Proxy	GET	10-10-148-3.p.thmla...	/assets/js/scripts.js		0	200	601	This is the scrip
3	18:33:19 27 Jul 2023	New	Repeater	GET	10-10-148-3.p.thmla...	/		0	200	6802	

Request
Pretty Raw Hex
8 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
9 Sec-Fetch-Site: none
10 Sec-Fetch-Mode: navigate
11 Sec-Fetch-User: ?1
12 Sec-Fetch-Dest: document
13 Accept-Encoding: gzip, deflate
14 Accept-Language: en-US,en;q=0.9
15 Connection: close
16

Response
Pretty Raw Hex Render
1 HTTP/1.1 200 OK
2 Server: nginx/1.14.0 (Ubuntu)
3 Date: Thu, 27 Jul 2023 10:33:20 GMT
4 Content-Type: text/html; charset=utf-8
5 Connection: close
6 Front-End-Https: on
7 Content-Length: 6613
8
9 <!DOCTYPE html>
10 <html lang=en>
11 <head>
12 <title>
0 matches

Notes
This is a test note
Inspector Notes

Soru ⇒ Kaydedilen istekler salt okunur mu? (evet/hayır)

Cevap ⇒ **yea**

Task 11 Conclusion (Görev 11 Sonuç)

Burp Suite Diğer Modüller odasını tamamladığınız için tebrikler!

Özetlemek gerekirse, Decoder verileri kodlamanıza ve kodlarını çözmenize olanak tanıyarak aktarılan bilgilerin okunmasını ve anlaşılmasını kolaylaştırır. Comparer, iki veri kümesi arasındaki farklılıkları tespit etmenizi sağlar, bu da güvenlik açıklarını veya anormallikleri belirlemede çok önemli olabilir. Sıralayıcı, belirteçler üzerinde entropi analizi yapılmasına yardımcı olarak, bunların oluşumunun rastgeleliği ve sonuç olarak güvenlik seviyeleri hakkında içgörüler sağlar. Düzenleyici, daha sonra tekrar ziyaret etmek isteyebileceğiniz HTTP isteklerinin kopyalarını saklamanıza ve açıklama eklemenize olanak tanır.

Bu araçlar hakkında temel bir anlayışa sahip olmak, web uygulaması sızma testi söz konusu olduğunda sizi daha geniş bir beceri seti ile donatır.

Bu modülün bir sonraki ve son odasında Burp Suite Extensions aracını keşfedeceksiniz.

Soru ⇒ Organizer, Decoder, Sequencer ve Comparer'ın nasıl kullanılacağını anladım!

Cevap ⇒ **Cevap Gerekmemektedir.**