Analisis Protokol HTTP Menggunakan Wireshark

CARA MENGANALISIS PROTOKOL HTTP MENGGUNAKAN APLIKASI WIRESHARK

A. TEORI PROTOKOL HTTP

Pengenalan HTTP

HTTP merupakan protocol untuk melakukan transmisi/pengiriman hypermedia document seperti HTML, JavaScript,CSS Image, Video dan sebagainya, HTTP sebagai komunikasi antar web browser(Client) dan web server(Server), namun juga dapat digunakan untuk tujuan lain. HTTP memiliki banyak versi karena spesifikasinya selalu diperbarui, spesifikasi seperti aturan karena user pasti akan bertambah banyak seiringnya waktu. Pada saat ini kebanyakan web berjalan di HTTP 1.1 atau HTTP 2, berikut perbedaan HTTP 1.1 dengan HTTP 2 :

1. Saat ini di HTTP 1.1 merupakan fallback protokol(jika tidak support versi 1.1 maka pindah ke HTTP 2, bukan masalah lagi karena web browser akan tahu bahwa website tersebut menggunakan HTTP 1.1), dimana Web browser secara default akan melakukan request menggunakan HTTP/2, jika web server tidak mendukung, maka web browser akan melakukan fallback ke protkol HTTP/1.1
2. Secara garis besar, spesifikasi HTTP 2 dengan HTTP 1.1 sama saja namun ada improvment, HTTP 1.1 mengirimkan Requst dalam bentuk teks , sedangkan HTTP 2 menggunakan Binary(0 dan 1), untuk pengiriman gambar lebih cepat menggunakan binary dibandingkan HTTP 1.1
3. Selain itu di HTTP 2, menggunakan algoritma kompresi untuk memperkecil request dan mendukung multiplexing , sehingga bisa mengirim beberapa requeset dalam satu connection yang sama, contoh ada foto 2mb, maka akan dikompress foto tersebut di dalam content.

HTTP Terminologi

Saat kita belajar HTTP ada banyak sekali menggunakan terminology, isitilah atau teknologi dan kita perlu mengerti tentang hal tersebut

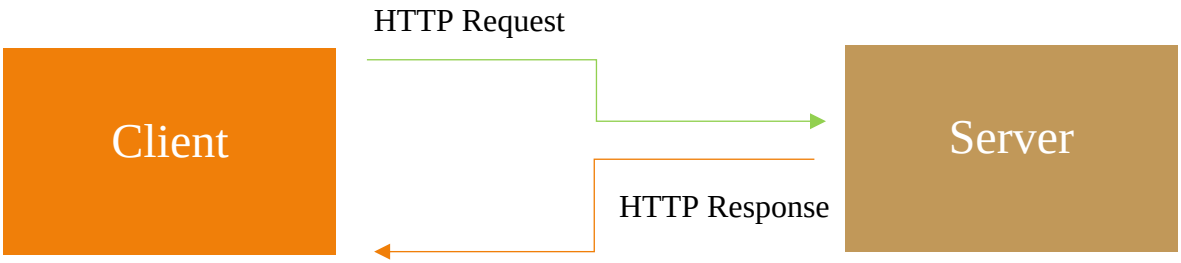
No	Teknologi	Keterangan
1	Web Browser(Client)	merupakan aplikasi yang digunakan untuk mengakses web menggunakan protokol HTTP, contoh aplikasi google chorem, safari, mozila ,dll

2	Web Server(Server)	Web server merupakan aplikasi berjalan di jaringan Internet yang betugas sebagai server
3	TCP(Transmission Control Protokol)	(Protokol berkomunikasi antar client dengan web servernya) adalah salah satu protokol dalam jaringan komputer yang biasa digunakan oleh web, email, FTP ,lainnya
4	IP(Internet Protokol)	IP digunakan sebagai indentitas komputer jaringan
5	URL(Uniform Resource locator)	URL merupakan alamat sebuah dari resource di Web(Link Web)
6	DNS(Domain Name Server)	Tempat yang berisi data katalog pemetaan antara nama domain di URL menuju lokasi IP komputer (sebenarnya itu adalah alamat ip Google yang kita akses lalu diubah menjadi ringkas seperti google.com)

Client-Server

HTTP mengikuti model client-server, dengan client mengirimkan HTTP Request untuk mengirimkan informasi ke server, lalu server akan menerima HTTP Request dari client dan server membalas dengan HTTP Response.

Diagram Client Server

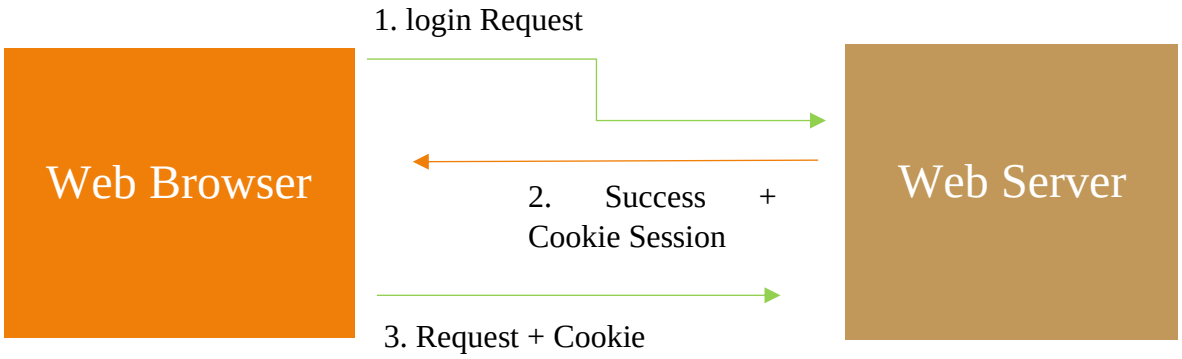


Stateless & Cookie

HTTP merupakan protocol yang stateless artinya tiap HTTP Request merupakan request yang independent(kekuasaan bebas), tidak ada keterkaitan atau hubungan dengan HTTP request sebelum atau setelahnya. Hal ini dilakukan agar HTTP request tidak harus dilakukan dalam sequence(terstruktur), sehingga client bisa melakukan HTTP Request secara bebas tanpa ada aturan harus dimulai dari mana. Jika HTTP bersifat Stateless lalu bagaimana server tahu? Jika client sudah login sebelum mengakses halaman tertentu(login)? Hal tersebut menggunakan fitur

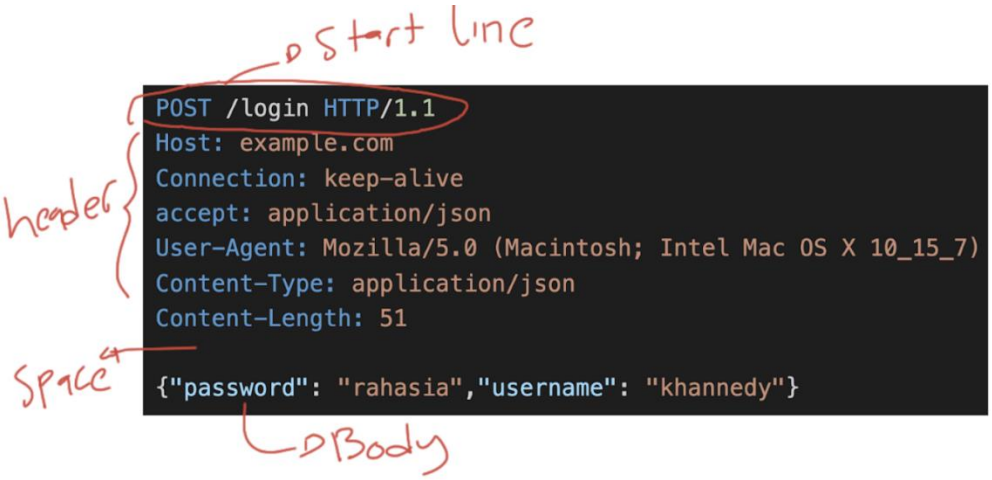
HTTP cookie yang merupakan informasi yang diberikan oleh server lalu client secara otomatis akan menyimpan data tersebut.

Penggunaan HTTP Cookie



HTTP Header

HTTP Header merupakan informasi tambahan yang biasa dikirim di Request atau Response, HTTP Header biasanya digunakan agar informasi tidak harus dikirim melalui Request Body atau Response Body. HTTP Header berisi key : value, dan saat ini sudah banyak sekali standarisasi nama key pada HTTP Header.



Contoh HTTP Header

No	HTTP Header	Keterangan
1	Host	Authority pada URL (wajib sejak versi HTTP/1.1)
2	Content-Type	Tipe data dari HTTP Body

3	User-Agent	Informasi user agent (seperti browser dan sistem operasi)
4	Accept	Tipe data yang diterima oleh Client
5	Authorization	Credential untuk autentikasi (misal username + password)

HTTP Request dan HTTP Response

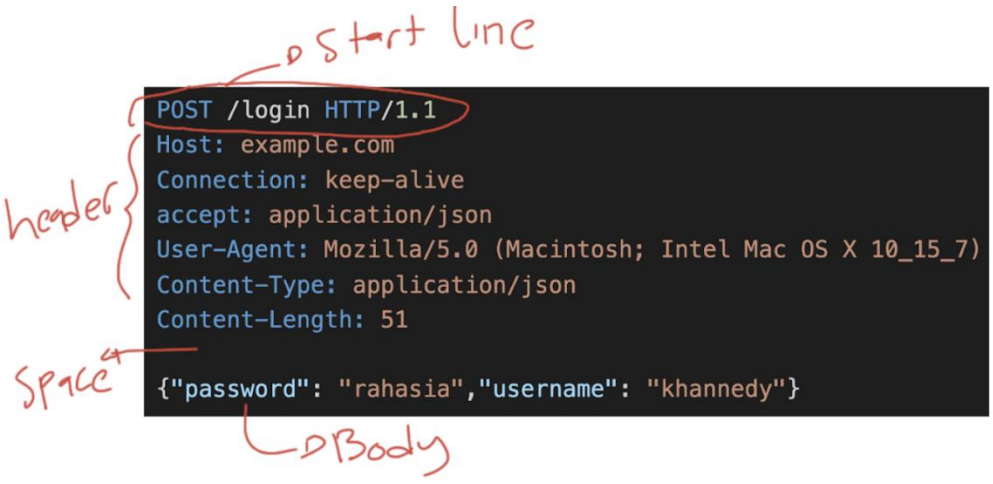
HTTP Message

Merupakan Pesan HTTP Request dan HTTP Response secara detail, HTTP Message memiliki standariasai format(sudah ada aturaran baku tidak asal membuatnya sesuka kita) dengan demikian, jika kita ingin membuat Client dan server sendiri, sebenarnya kita bisa lakukan asal kita mengikuti standarisasi format HTTP message.

HTTP Request

Merupakan proses Client yang mengirimkan request ke server dalam bentuk HTTP request , HTTP request berisikan informasi seperti lokasi resource, data yang dikirim jika ada , dan lain-lain. HTTP request akan diterima oleh server selanjutnya server akan memproses request yang diminta oleh client tersebut (data disimpan oleh web server) seperti kita upload story facebook minta ke web server lalu web server menerima dan disimpan di facebook.

HTPP Message untuk Request :



HTTP Response

Merupakan pesan request dari client lalu server memproses HTTP request, setelah itu dibalas dengan HTTP Respone oleh server .HTTP response biasanya berisikan data yang diminta oleh client dalam HTTP request.

HTTP message untuk Response :

HTTP Method

HTTP Method merupakan metode HTTP Request untuk penanda/aksi meminta data ke server sehingga tujuannya mengambil data, buat data baru, update data, atau hapus data. Dalam HTTP Request, hal yang pertama kita perlu tentukan adalah HTTP Method, HTTP Method mirip seperti kategori request .Ada banyak HTTP Method yang dapat kita gunakan ketika membuat HTTP Request, dan kita bisa sesuaikan sesuai dengan kebutuhan yang kita inginkan.

Jenis-jenis HTTP Method :

No	Method	Tujuan /Semantik
1	GET	Menerima data
2	POST	Buat data baru atau upload data
3	HEAD	Menimpa data atau update data tersebut
4	DETELE	Hapus data
5	OPTION	Sebagai mendeskripsikan opsi komunikasi yang tersedia(menampilkan get,post,head)
6	TRACE	Untuk Debbugging

HTTP Status

HTTP Status merupakan kode HTTP Response yang mengindikasikan apakah sebuah request yang diterima Server sukses, gagal atau adah lain yang diketahui oleh client

HTTP status diklasifikasikan(dikategorikan) dalam lima group yaitu :

1. Infotmational Response(100-199)

Informasi response mengindikasi bahwa request telah diterima dan dimengeri, namun client diminta untuk menunggu tahapan akhir response Pada kenyataanya, informational response sangat jarang sekali digunakan

https://developer.mozilla.org/en-US/docs/Web/HTTP/Status#information_responses

2. Successful Response (200-299)

Succesful Response merupkana kode yang mengindikasi bahwa request yang dikirim oleh client telah diterima, dimengerti dan sukses diproses oleh server

https://developer.mozilla.org/en-US/docs/Web/HTTP/Status#successful_responses

3. Redirect (300-399)

Redirect status code mengindikasi bahwa client harus melakukan aksi selanjutnya untuk menyelesaikan request, biasanya redirect status code digunakan ketika lokasi sebuah resource berubah,sehingga server meminta client untuk berpindah ke URL Lain, contoh seperi kita akan ingin login menggunakan facebook , login terlebih dahulu jika sudah login maka kita diarahkan ke halaman facebooknya

https://developer.mozilla.org/en-US/docs/Web/HTTP/Status#redirection_messages

4. Client Error(400-499)

Client error status code merupakan indikasi bahwa request yang dikirimkan oleh client tidak diterima oleh server dikarenakan request yang dikirim dianggap tidak valid

contoh client mengirim body yang salah, client melakukan request ke tanpa autentikasi di resource ke tanpa autentikasi di resource yang mewajibkan autentikasi dan dll(data client error server tidak menerimanya)

https://developer.mozilla.org/en-US/docs/Web/HTTP/Status#client_error_responses

5. Server Error(500-599)

server error status code mengindikasi bahwa terjadi kesalahan di Server, biasanya ini ketika ada masalah di server, seperti misalnya tidak bisa terkoneksi ke basis data, terdapat jaringan eror di server, dan lain-lain

https://developer.mozilla.org/en-US/docs/Web/HTTP/Status#server_error_responses

HTTP Body

HTTP body merupakan data yang bisa dikirim di HTTP Request atau , data yang diterima dari HTTP, artinya client bisa mengirim data ke server menggunakan HTTP body, begitu juga sebaliknya server bisa memnberikan body di response menggunakan HTTP Body (bisa mengirim data melalui body bolak balik), contoh client akan menonton youtube yang akan dipilih oleh user berupa body yang dikirim juga ke server maka server juga akna mengirimkan itu data response sebuah body juga

HTTP body akan mengirim data sesuai dengan Content-type yang ada di HTTP header

- Content-TYPE dengan HTTP body :

HTTP Body erat kaitannya dengan Header key Content-type, biasanya agar client dan server mudah mengerti isi HTTP body, HTTP Message akan memiliki Header Content-type, yang berisi informasi tipe data HTTP body , HTTP body bisa berisikan teks(htlm, js, css,json) atau binary(image,video,audio), data content-type sudah memiliki standarisasi, misalnya bisa kita lihat di link berikut

B. LANGKAH – LANGKAH CAPTURING HTTP MENGGUNAKAN WIRESHARK

1. **Siapkan Aplikasi Wireshark**

   Aplikasi wireshark merupakan aplikasi untuk mengecek/menangkap perjalan data komuniksi diperangkat kita

2. **Konektivitas Komunikasi**

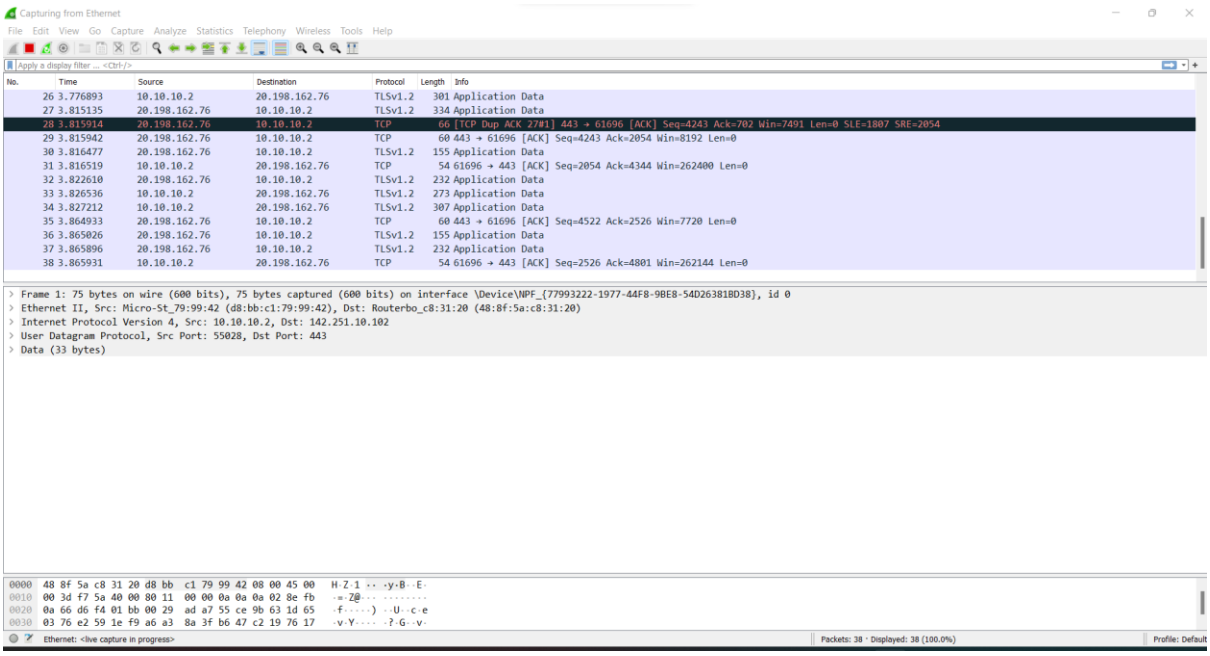
   Pastikan perangkat kalian sudah terhubung dengan intenet menggunakan media komunikasi sesuai dengan gunakan, misal disini saya menggunakan media komunikasi yatitu Ethernet yang sudah terhubung ke internet. Jika kalian tidak bisa mengakses internet, bisa menggunakan aplikasi xampp untuk menjalankan web server lalu cari localhostnya, karena localhost sudah HTTP jadi lebih mudah

3. **Membuka Aplikasi Wireshark dan memilih Interface**

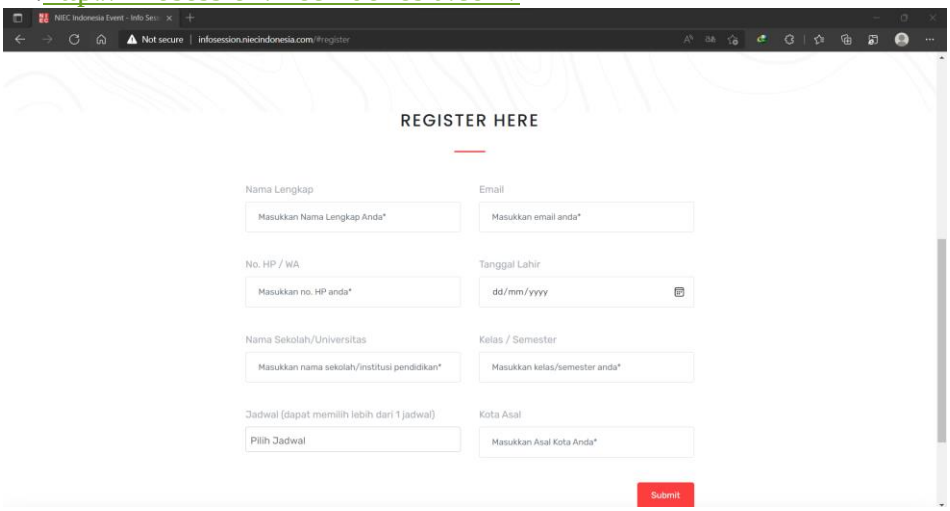
   Buka aplikasi wireshark lalu pilih interface atau media transmisi yang kalian gunakan disini saya menggunakan interface ethernet maka pilih ethernet

4. **Tampilan Filter Wireshark**

   Setelah memilih interface tampilan awal muncul seperti pada gambar di bawah dan terlihat ada paket data yang sudah berjalan sesuai dengan protokolnya, itu karenakan ada backgournd data yang berjalan dioperasi kalian.



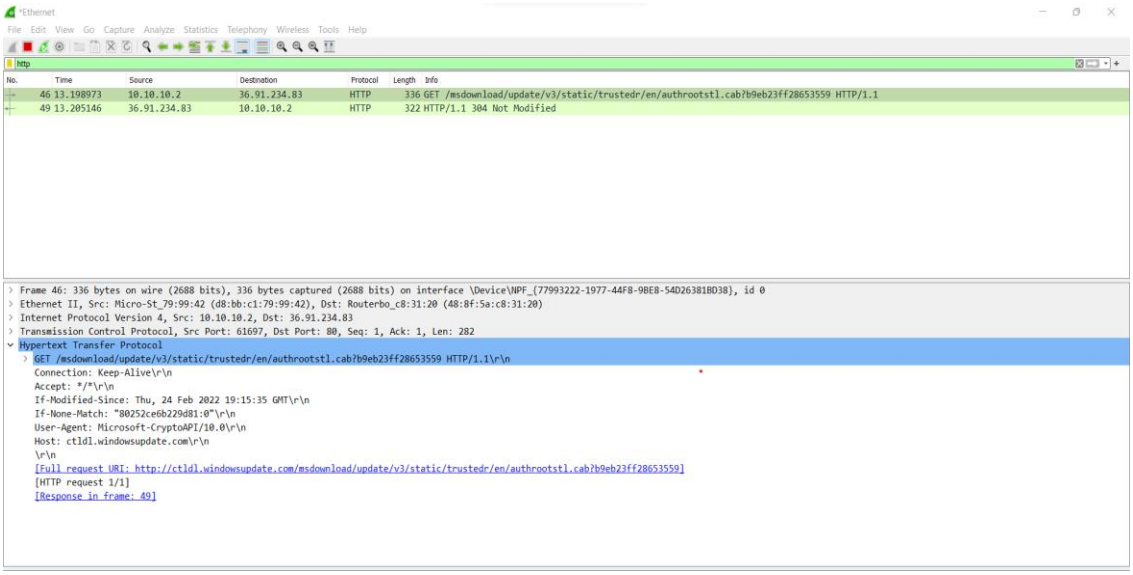
5. **Cara menggunakan Ptokol HTPP**
   Bukalah web browser kalian lalu cari website dengan menggunakan protocol HTTP, misal seperti ini :http://infosession.niecindonesia.com/



6. **Mencari Filter Protokol HTTP di Wireshark**
   Untuk mencari filter protocol HTTP diwireshark kalian harus membuka website HTTP sesuai dengan Langkah 5 sebelumnya. Lalu kalian buka wireshark cari kolom search filter lalu ketik ‘http’, sehingga muncullah ptokol http diwireshark kalian.

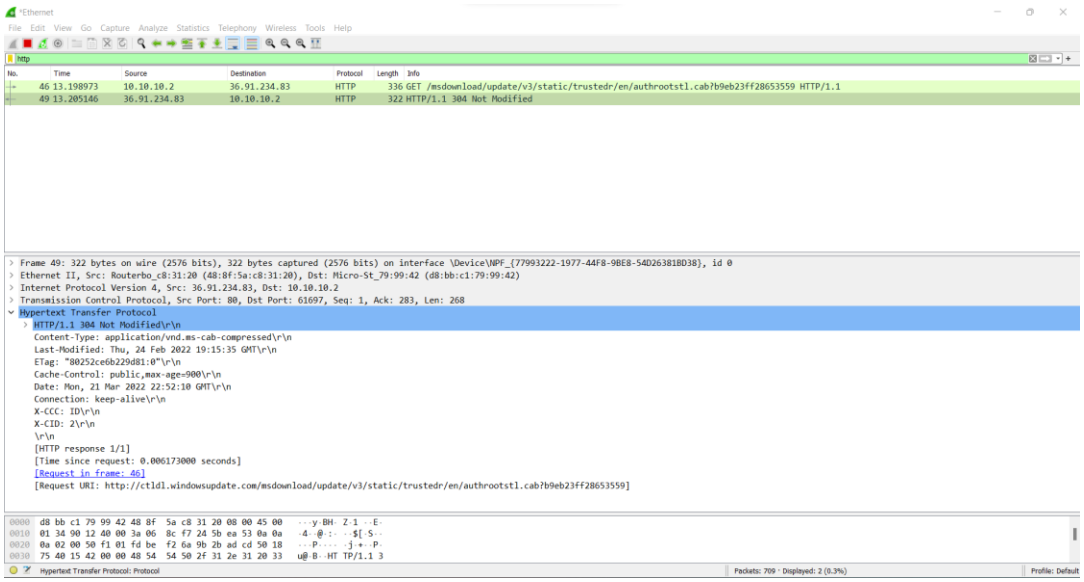
7. **Detail HTTP Request (Client)**

   Pilih bagian atas , karena dibagian atas tersebut merupakan Request Client yang berisikan method GET , komponen-kompoenn yang terdapat di HTTP Message Request dan juga HTTP Status



8. **Detail HTTP Response (Server)**

   Pilih bagian kedua atau dibawahnya HTTP Request , bagian tersebut merupakan HTTP Response (Server) yang berisikan komponen-kompoen yang terdapat di HTTP Message Response dan juga HTTP Status



Sumber:

- https://www.youtube.com/watch?v=92Rjzrq4oIg&t=4707s&ab_channel=ProgrammerZamanNow
- https://developer.mozilla.org/en-US/docs/Web/HTTP