JARINGAN KOMPUTER – TUGAS PENDAHULUAN MODUL 12
ICMP (INTERNET CONTROL MESSAGE PROTOCOL)

Nama : Muhammad Hamzah Haifan Ma'ruf

NIM : 2311102091

Kelas : S1IF-11-07

Kode Dosen : AIZ

JAWABAN MODUL

A. Teori

1. Hubungan OSI dan TCP/IP Model beserta Penempatan Protokol
   Model OSI (Open Systems Interconnection) terdiri dari tujuh lapisan: Physical, Data Link, Network, Transport, Session, Presentation, dan Application. Sedangkan model TCP/IP terdiri dari empat lapisan: Network Access, Internet, Transport, dan Application. Hubungan antar lapisan ini bersifat kompatibel meskipun jumlahnya berbeda. Lapisan Physical dan Data Link pada OSI disatukan dalam Network Access layer pada TCP/IP. Network layer OSI setara dengan Internet layer TCP/IP. Transport layer tetap sama pada kedua model. Sementara lapisan Session, Presentation, dan Application pada OSI disatukan menjadi Application layer pada TCP/IP. Protokol HTTP, FTP, SMTP, dan DNS berada pada lapisan Application baik di OSI maupun TCP/IP. UDP dan TCP berada pada Transport layer. IP, DHCP, dan ICMP berada pada Network layer OSI dan Internet layer TCP/IP.

2. Peran ICMP dalam Melengkapi Kekurangan IP
   Protokol IP memang tidak menyediakan mekanisme pelaporan kesalahan, karena ia hanya berfungsi untuk pengalamatan dan pengiriman paket tanpa memperhatikan apakah paket berhasil sampai atau tidak. Di sinilah protokol ICMP (Internet Control Message Protocol) berperan, yaitu menyediakan pesan kesalahan dan diagnostik untuk menginformasikan adanya masalah dalam pengiriman data. Dua contoh pesan kesalahan ICMP yang umum digunakan adalah Destination Unreachable, yang dikirim jika paket tidak dapat mencapai tujuan (misalnya karena port tidak tersedia), dan Time Exceeded, yang dikirim jika waktu hidup (TTL) paket habis

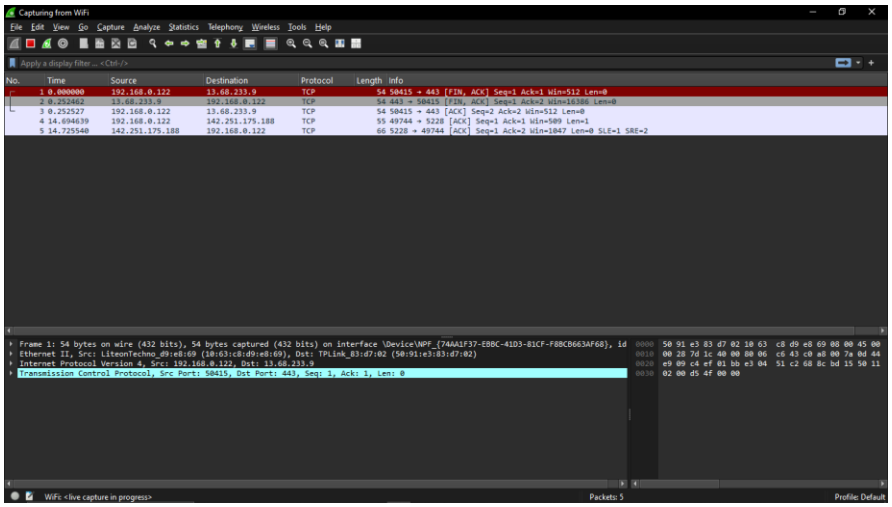
sebelum mencapai tujuan. ICMP tidak digunakan untuk pengiriman data aplikasi karena ia bukan protokol komunikasi end-to-end seperti TCP atau UDP, melainkan hanya digunakan untuk kontrol dan diagnostik pada lapisan jaringan.

3. Cara Kerja ICMP Ping dalam Menguji Konektivitas
   ICMP Ping bekerja dengan cara mengirimkan pesan ICMP Echo Request ke host tujuan, lalu menunggu balasan berupa Echo Reply. Ketika pengguna menjalankan perintah ping, sistem akan mengirimkan paket ICMP Echo Request yang ditujukan ke alamat IP tertentu. Jika host tujuan menerima paket tersebut dan dapat merespons, maka ia akan mengirimkan kembali paket ICMP Echo Reply ke pengirim. Dengan mengukur waktu antara pengiriman dan penerimaan balasan, pengguna dapat mengetahui apakah host dapat dijangkau dan seberapa cepat koneksi berlangsung. Fungsi ini sangat berguna dalam pengujian dasar konektivitas antar perangkat dalam jaringan.

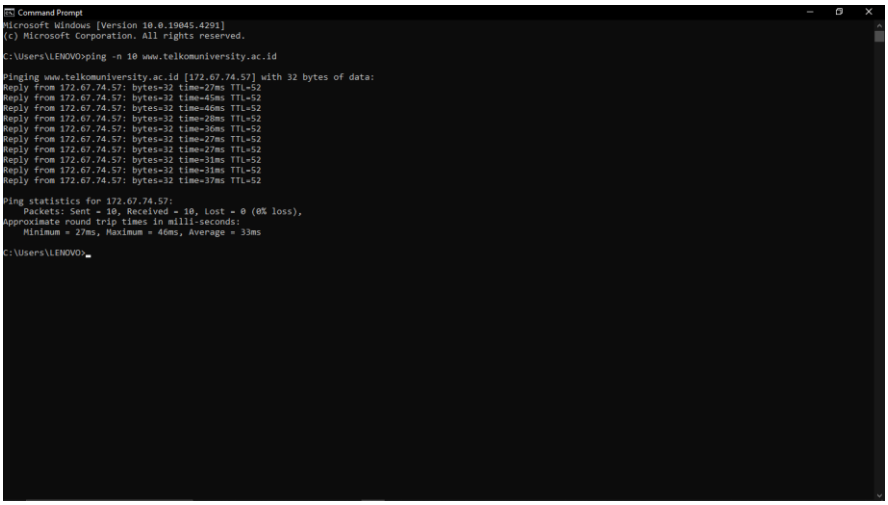
4. Pemanfaatan ICMP dan TTL dalam Proses Traceroute
   Traceroute menggunakan ICMP untuk melacak jalur yang dilalui paket dari sumber ke tujuan dengan memanfaatkan field TTL (Time To Live) dalam header IP. TTL adalah angka yang menandai batas waktu perjalanan sebuah paket, dan nilainya dikurangi satu oleh setiap router yang dilewati. Traceroute mengirimkan paket dengan TTL yang awalnya diatur ke 1. Router pertama akan mengurangi TTL menjadi 0 dan membuang paket tersebut, lalu mengirimkan pesan ICMP Time Exceeded ke pengirim. Proses ini diulang dengan TTL yang dinaikkan satu per satu, sehingga setiap router di sepanjang jalur akan mengirimkan balasan ICMP hingga paket mencapai tujuan akhir. Dengan cara ini, traceroute dapat mencatat alamat setiap router yang dilewati, membentuk peta jalur koneksi dari sumber ke tujuan. Pendekatan bertahap dengan peningkatan TTL diperlukan agar dapat mengidentifikasi setiap titik (hop) dalam rute jaringan.

B. Packet Trace and Cek

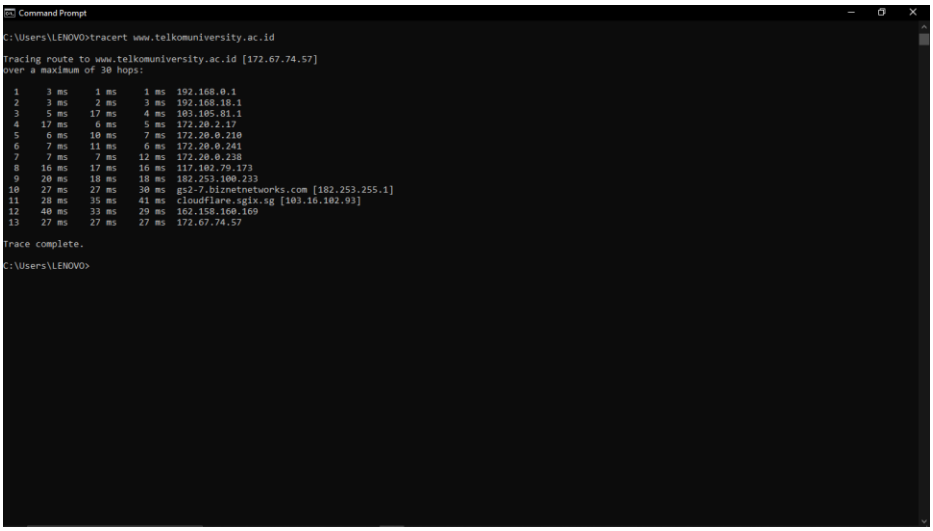
1. Buka Command Prompt (CMD) atau Terminal
2. Buka Wireshark dan Mulai Capture



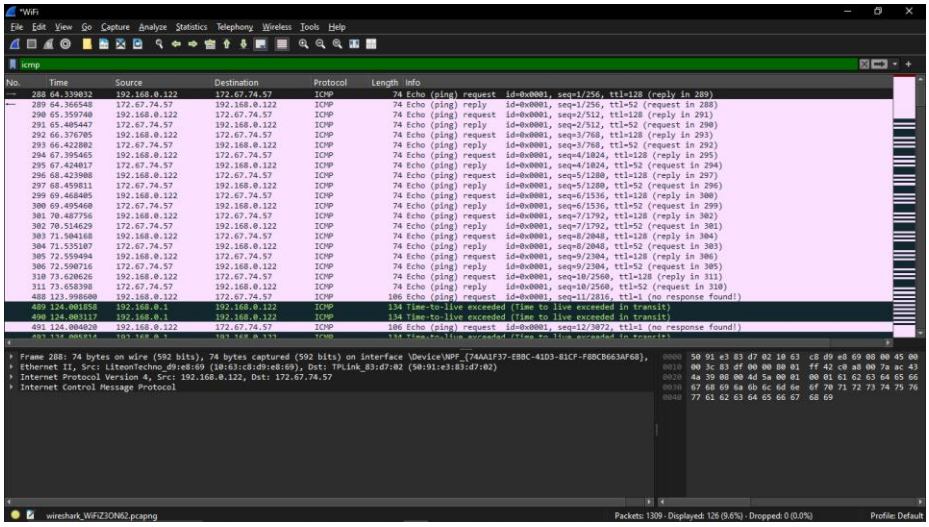
3. Jalankan Perintah Ping



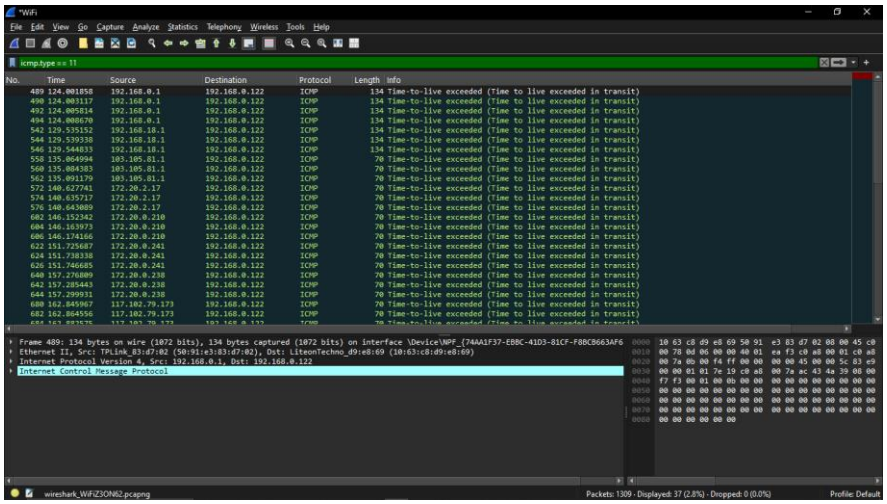
4. Jalankan Perintah Traceroute



5. Hentikan Capture di Wireshark
6. Filter Wireshark untuk Menampilkan Paket Ping



7. Filter Wireshark untuk Menampilkan Paket Traceroute



C. Socket Programming with ICMP

1. Perbedaan Fungsi run_ping_windows() dan run_traceroute_windows()
   Fungsi run_ping_windows() dan run_traceroute_windows() pada file icmp_client.py memiliki perbedaan pada jenis perintah sistem yang dijalankan. Fungsi run_ping_windows() digunakan untuk mengirim perintah ping ke host menggunakan perintah ping -n yang bersifat khusus untuk sistem operasi Windows. Sedangkan run_traceroute_windows() digunakan untuk melakukan traceroute menggunakan perintah tracert, juga hanya tersedia di Windows. Kedua fungsi ini menggunakan modul subprocess untuk menjalankan perintah terminal secara langsung dari dalam script Python. Karena perintah ping -n dan tracert merupakan sintaks yang hanya dikenali oleh Command Prompt di Windows, maka kedua fungsi ini tidak dapat dijalankan di sistem operasi lain seperti Linux atau macOS yang menggunakan perintah ping -c dan traceroute. Oleh sebab itu, fungsinya terbatas hanya untuk lingkungan Windows.

2. Mekanisme simulate_traceroute() dan Alasan Penyesuaian HOPS
   Fungsi simulate_traceroute() dalam icmp_client.py bekerja dengan cara mensimulasikan proses traceroute menggunakan protokol UDP, bukan ICMP asli. Fungsi ini mengirimkan paket UDP ke port-port tertentu yang mewakili setiap hop (lompatan) dalam perjalanan paket menuju server tujuan. Port-port tersebut ditangani oleh icmp_server.py, yang harus dijalankan dan siap menerima paket di rentang port yang telah ditentukan, misalnya dari port 33434 hingga 33444 untuk mensimulasikan 10 hop. Daftar HOPS harus disesuaikan karena setiap port UDP pada server merepresentasikan satu node/router virtual. Client akan mengirimkan

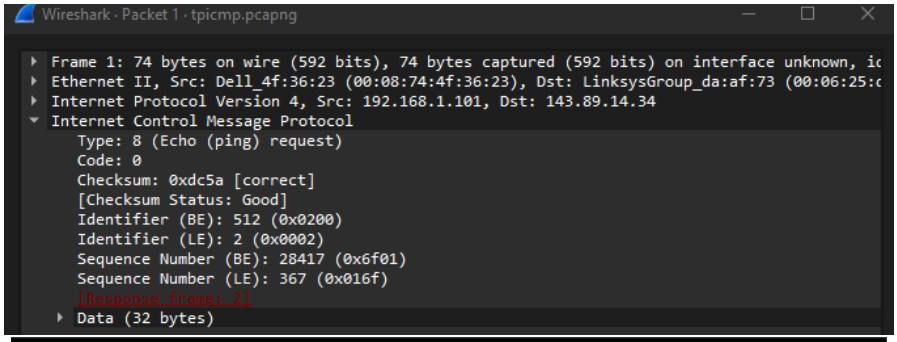
pesan ke port pertama, menerima balasan, lalu meningkat ke port berikutnya untuk mensimulasikan perjalanan paket. Proses ini meniru perilaku TTL (Time To Live) dalam traceroute asli. Tanpa penyesuaian jumlah port dan hops, simulasi tidak akan berjalan dengan baik karena client tidak akan mendapat respons dari port yang tidak aktif.

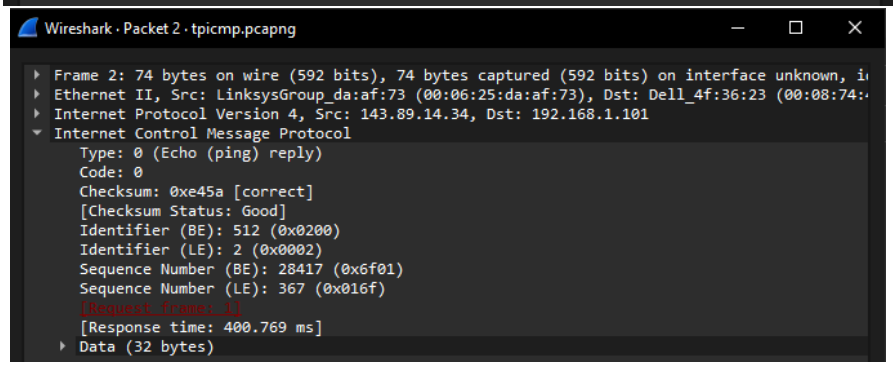
3. Langkah Menjalankan Simulasi dengan icmp_server.py dan icmp_client.py
   Untuk menjalankan simulasi ping dan traceroute lokal menggunakan script icmp_server.py dan icmp_client.py, pengguna perlu menyiapkan minimal dua terminal. Terminal pertama digunakan untuk menjalankan server dengan perintah python icmp_server.py, di mana server akan mendengarkan pada sejumlah port UDP yang telah ditentukan. Port ini harus mencerminkan jumlah hop yang ingin disimulasikan. Terminal kedua digunakan untuk menjalankan client dengan perintah python icmp_client.py. Di sisi client, pengguna dapat memilih opsi 4 untuk simulate_ping() atau opsi 5 untuk simulate_traceroute(), kemudian memasukkan alamat target seperti localhost atau 127.0.0.1. Selama simulasi berjalan, client akan mengirimkan pesan UDP ke berbagai port pada server, dan server akan mengembalikan respons yang menggambarkan hop yang dilewati. Dengan konfigurasi ini, simulasi dapat berjalan secara lokal tanpa perlu koneksi ke jaringan eksternal.

4. Perbandingan ICMP Asli dan Simulasi ICMP
   Pendekatan pada opsi 1 dan 2, yang menggunakan fungsi run_ping_windows() dan run_traceroute_windows(), memanfaatkan protokol ICMP asli melalui perintah bawaan sistem operasi untuk menguji koneksi ke host eksternal seperti google.com atau yahoo.com. Protokol ICMP ini mengirim paket Echo Request dan menerima Echo Reply, serta menggunakan pesan Time Exceeded dalam traceroute. Sebaliknya, opsi 4 dan 5 menggunakan fungsi simulate_ping() dan simulate_traceroute() yang hanya menyimulasikan perilaku ICMP dengan mengirimkan paket UDP ke port lokal tertentu. Simulasi ini tidak melibatkan ICMP sama sekali dan hanya bergantung pada script Python server yang sudah dikonfigurasi. Perbedaan lainnya terletak pada fleksibilitas: ICMP asli memerlukan koneksi internet dan perintah sistem, sedangkan simulasi dapat dijalankan sepenuhnya secara lokal. Selain itu, ICMP asli kadang membutuhkan hak akses administrator terutama di sistem Linux, sedangkan simulasi UDP lebih ringan dan tidak membutuhkan hak khusus. Dengan demikian, ICMP asli lebih merepresentasikan kondisi nyata, sedangkan simulasi lebih cocok untuk pembelajaran dan eksperimen di lingkungan lokal.

D. Wireshark ICMP

1. Analisis Paket 1 dan 2 – ICMP Echo Request dan Reply





Jenis Pesan:

- Paket 1: Type = 8 (Echo Request)
- Paket 2: Type = 0 (Echo Reply)

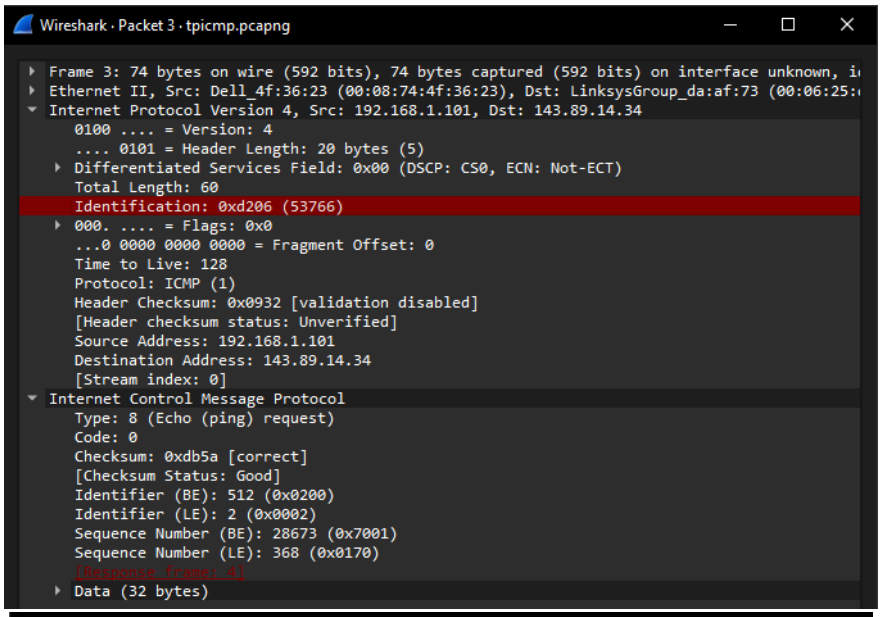
Identifier dan Sequence Number:

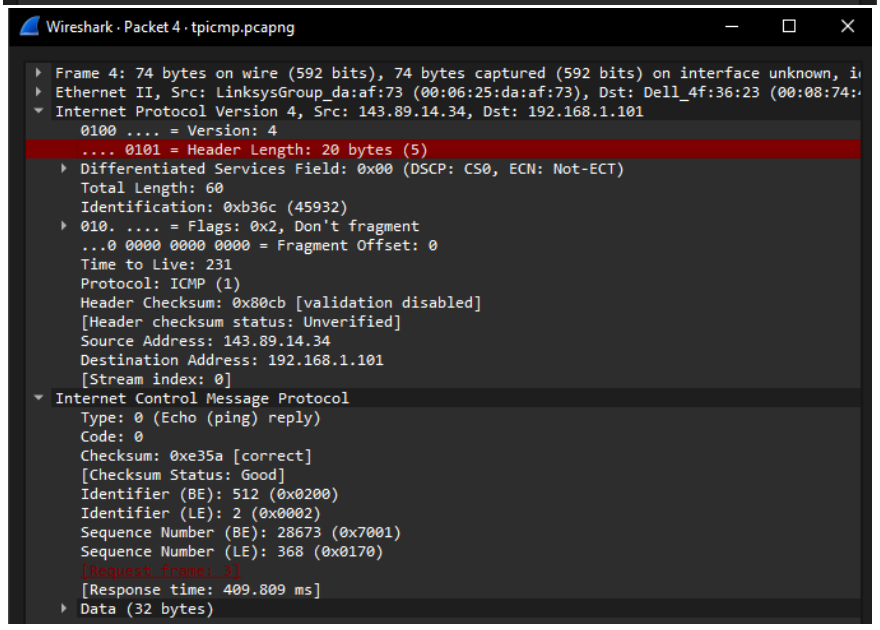
- Identifier (BE): 512 (0x0200)
- Identifier (LE): 2 (0x0002)

Sequence Number:

- Sequence Number (BE): 28417 (0x6f01)
- Sequence Number (LE): 367 (0x016f)

2. Perbandingan Paket 3 dan 4 – TTL dan Fragmentasi





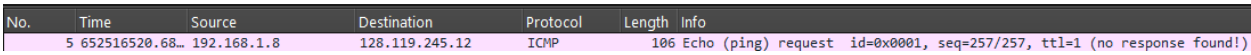
Perbandingan TTL:

- Echo Request Time to Live: 128
- Echo Reply Time to Live: 231

TTL menurun tiap kali paket melewati router. Nilai TTL awal yang tinggi pada Request dan nilai lebih kecil pada Reply menunjukkan jumlah hop yang dilewati

paket. 010. .... = Flags: 0x2, Don't fragment, Fragment Offset: 0 tidak terjadi fragmentasi.

3. TTL=1 Tanpa Balasan – Paket 5



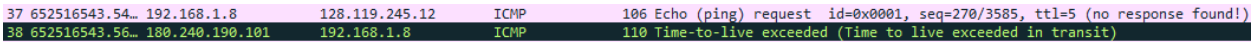
No.	Time	Source	Destination	Protocol	Length	Info
5	652516520.68…	192.168.1.8	128.119.245.12	ICMP	106	Echo (ping) request id=0x0001, seq=257/257, ttl=1 (no response found!)

Penyebab Tidak Ada Balasan:
TTL (Time To Live) adalah batas jumlah hop. Saat TTL = 1, maka:

- Paket akan berkurang menjadi 0 di router pertama.
- Router tidak meneruskan paket.
- Router akan mengirimkan ICMP Time Exceeded, namun jika diblok atau gagal, tidak ada balasan.

Makna TTL=1:
Digunakan untuk menguji hop pertama dalam traceroute. Paket tidak bisa menjangkau tujuan karena TTL habis di awal lintasan.

4. Traceroute dengan TTL=5 – Paket 37 dan 38



37	652516543.54…	192.168.1.8	128.119.245.12	ICMP	106 Echo (ping) request id=0x0001, seq=270/3585, ttl=5 (no response found!)
38	652516543.56…	180.240.190.101	192.168.1.8	ICMP	110 Time-to-live exceeded (Time to live exceeded in transit)

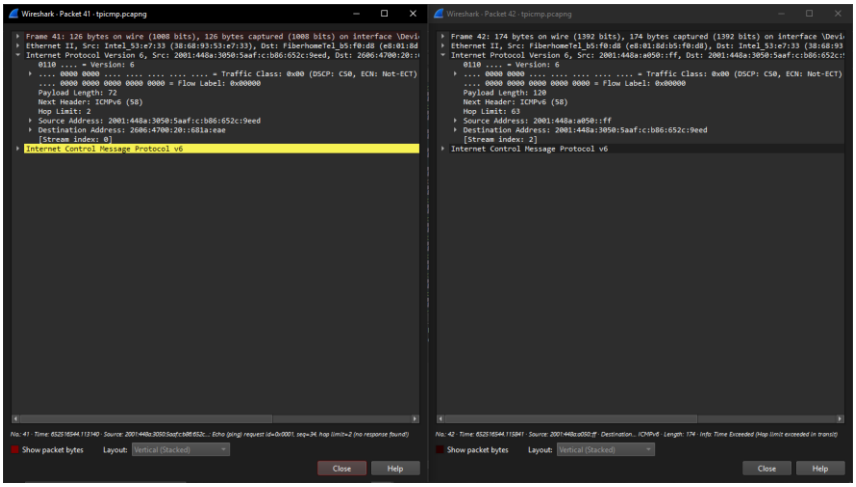
Ilustrasi Jalur:
[Host] —> [Router 1] —> [Router 2] —> [Router 3] —> [Router 4] —> [Router 5: 180.240.190.101]

Cara Traceroute Bekerja:

- Traceroute mengirimkan Echo Request dengan TTL bertahap mulai dari 1, lalu 2, dst.
- Setiap kali TTL mencapai nol di router tertentu, router mengirimkan ICMP Time Exceeded.
- Dengan begitu, traceroute dapat merekam hop satu per satu berdasarkan TTL.

Peran TTL dan Time Exceeded:
TTL adalah "batas hidup paket". Saat TTL habis, router akan membalas dengan Type 11 (Time Exceeded), dan client mencatat alamat IP router tersebut.

5. ICMPv6 Echo Request dan Time Exceeded – Paket 41 dan 42



Paket 41:

- Echo Request dari 2001:448a:3050:... ke 2606:4700:20::...
- Hop limit = 2 → tidak sampai tujuan

Paket 42:

- Time Exceeded dari router 2001:448a:a050::ff
- Artinya hop limit sudah habis, paket dibuang

Kenapa tidak ada balasan?

- Karena hop limit = 0 sebelum sampai ke tujuan

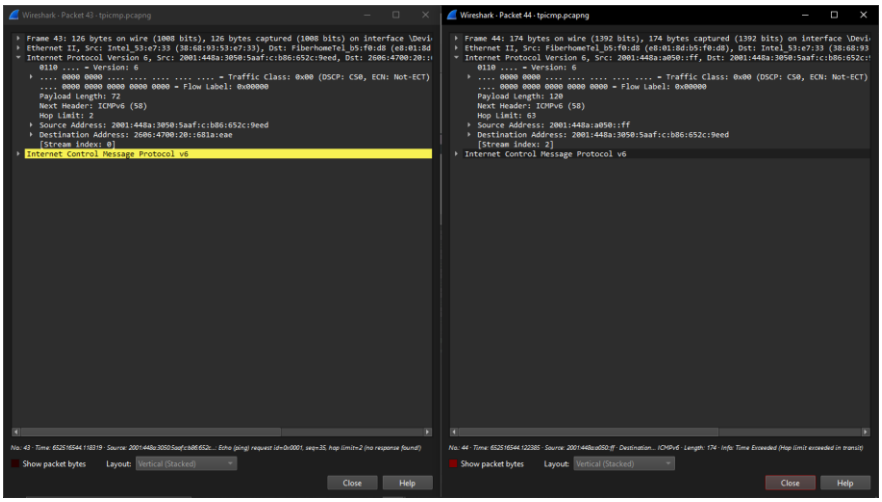
Fungsi Time Exceeded:

- Memberi tahu pengirim bahwa paket gagal lanjut

Perbedaan ICMPv4 vs ICMPv6:

- ICMPv4: pakai TTL
- ICMPv6: pakai Hop Limit
- Fungsi sama → bantu traceroute

6. ICMPv6 Time Exceeded & Hop Limit (Paket 43 & 44)



Paket 43:

- ICMPv6 Echo Request (hop limit = 2), tidak mendapat balasan
- Alamat sumber: 2001:448a:3050:..., tujuan: 2606:4700:20::...

Paket 44:

- ICMPv6 Time Exceeded, dikirim oleh router 2001:448a:a050::ff
- Artinya hop limit habis → paket dibuang

Perbedaan ICMPv6 & ICMPv4:

- ICMPv6 → pakai Hop Limit, ICMPv4 → pakai TTL
- Fungsi sama: deteksi lintasan pada traceroute

Kenapa error terjadi?

- Karena hop limit mencapai 0, router tidak meneruskan paket
- Sebagai gantinya, router kirim pesan “Time Exceeded” ke pengirim