JARINGAN KOMPUTER – TUGAS PENDAHULUAN MODUL 5
UDP (USER DATAGRAM PROTOCOL)

Nama : Muhammad Hamzah Haifan Ma'ruf

NIM : 2311102091

Kelas : S1IF-11-07

Kode Dosen : AIZ

JAWABAN MODUL

A. Teori

1. Saat ini anda seharusnya sudah mengetahui apa itu OSI Model dan TCP/IP Model. Tuliskan dan jelaskan hubungan layer pada OSI Model dengan layer pada TCP/IP Model (contoh Transport Layer pada OSI Model setara dengan Transport Layer pada TCP/IP Layer) buatkan secara visual. Serta sebutkan protokol HTTP, DNS dan UDP berada pada layer mana untuk setiap model

   Hubungan OSI Model dan TCP/IP Model
   OSI Model dan TCP/IP Model adalah dua model yang menjelaskan bagaimana data dikirim dari satu perangkat ke perangkat lain. Berikut penjelasan hubungan antar layer:

   - Application Layer (OSI) setara dengan Application Layer (TCP/IP) — Protokol: HTTP, DNS
   - Presentation Layer (OSI) dan Session Layer (OSI) juga digabung ke dalam Application Layer (TCP/IP)
   - Transport Layer (OSI) setara dengan Transport Layer (TCP/IP) — Protokol: UDP
   - Network Layer (OSI) setara dengan Internet Layer (TCP/IP)
   - Data Link Layer (OSI) dan Physical Layer (OSI) digabung ke dalam Network Access Layer (TCP/IP)

OSI Model	TCP/IP Model	Protokol
Application (7)	Application (4)	HTTP, DNS
Presentation (6)	Application (4)	SSL/TLS
Session (5)	Application (4)	RPC, SMB
Transport (4)	Transport (3)	TCP, UDP
Network (3)	Internet (2)	IP, ICMP

Data Link (2)	Link (1)	Ethernet, WiFi
Physical (1)	Link (1)	Kabel, sinyal

2. Jelaskan apa yang dimaksud gambar tersebut berkaitan UDP!

   Gambar tersebut menunjukkan UDP Header, yaitu bagian penting dari protokol User Datagram Protocol (UDP) yang berisi informasi untuk memastikan data sampai ke tujuan dengan benar. Bagian pertama adalah Source Port (16 bit), yang menunjukkan dari aplikasi atau proses mana data dikirim. Ini berguna agar penerima tahu ke mana harus mengirim balasan. Lalu ada Destination Port (16 bit), yang menandakan ke aplikasi atau layanan mana data akan diteruskan di sisi penerima, misalnya port 53 untuk DNS atau port 80 untuk HTTP. Selanjutnya, ada Length (16 bit) yang menyatakan total panjang UDP header dan data (dalam byte), di mana panjang minimum header UDP adalah 8 byte. Checksum (16 bit) adalah bagian yang digunakan untuk mendeteksi error. Pengirim akan menghitung checksum dari header dan data, lalu mengirimkannya bersama paket. Penerima akan memeriksa ulang checksum ini. Jika hasilnya berbeda, berarti data mengalami kerusakan saat dikirim. Terakhir, ada bagian Data (Payload), yaitu isi utama dari paket yang dikirim, misalnya teks atau file kecil. UDP lebih cepat dan sederhana dibanding TCP karena header-nya ringan dan tidak memiliki mekanisme retransmisi. Namun, konsekuensinya, UDP lebih rentan terhadap kehilangan atau kerusakan data.

3. Jelaskan apa yang dimaksud dengan:

   a. UDP Checksum
      Mekanisme untuk mendeteksi error pada data yang dikirim melalui UDP. Checksum ini dihitung dari data dan header lalu dibandingkan di sisi penerima.

   b. Data Payload
      Isi utama dari data yang dikirim, misalnya pesan teks, file, atau video — tanpa header.

   c. UDP sender action
      Proses pengirim menghitung checksum, menambahkan header ke data, lalu mengirim ke penerima.

   d. UDP receiver action
      Penerima memeriksa checksum. Kalau cocok, data diterima; kalau tidak, data dibuang.

4. Jelaskan dan berikan contohnya pada istilah berikut berkaitan perhitungan biner dan hexadecimal untuk UDP Checksum:

    a. Carry
    Penjumlahan biner yang menghasilkan lebih dari 1 bit, kelebihan bit dibawa ke bit berikutnya.
    Contoh: 1101 + 1011 = 11000 (carry di posisi ke-5)

    b. Propagasi Carry
    Kalau carry terus berlanjut ke bit berikutnya dan harus diulang.
    Contoh: 1010 + 1100 = 10110 (carry dipropagasi lagi).

    c. Splitting
    Membagi data menjadi bagian kecil (16 bit) agar lebih mudah dihitung.
    Contoh: Data 32 bit dibagi jadi dua 16 bit.

    d. Modular Arithmetic
    Operasi matematika dengan sisa hasil bagi (mod). UDP checksum pakai mod 65535 (16 bit).

    e. Truncation
    Memotong hasil perhitungan agar sesuai ukuran (16 bit).
    Contoh: 1101011011101010 dipotong jadi 101011101010.

5. Jelaskan perbedaan dan berikan contoh serta kaitkan dengan perhitungan UDP checksum dari:

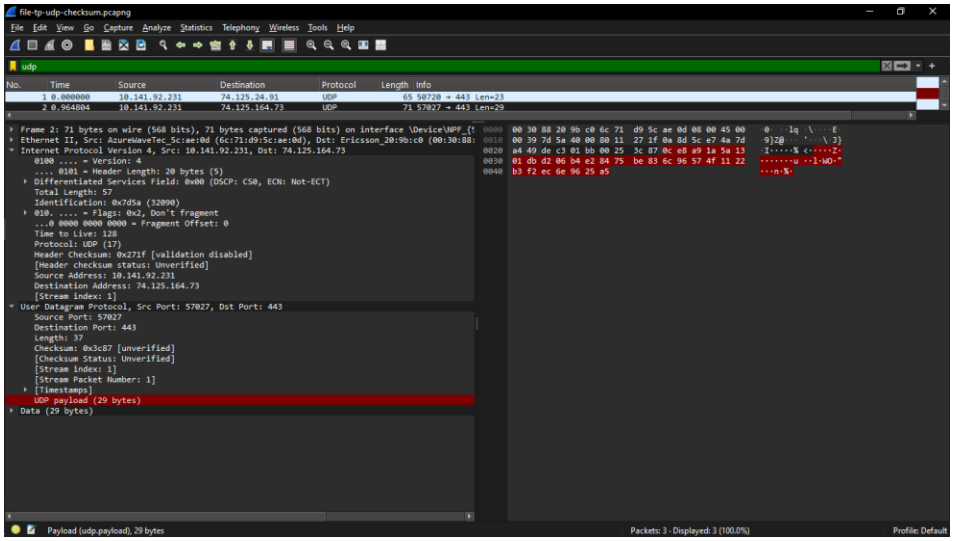
    a. One’s Complement
    Membalik semua bit (0 jadi 1, 1 jadi 0).
    Contoh: 1101 jadi 0010
    UDP Checksum pakai ini karena lebih mudah menambah carry.

    b. Two’s Complemen
    One’s Complement + 1.
    Contoh: 1101 jadi 0010 + 1 = 0011
    Dipakai lebih sering di perhitungan aritmatika komputer karena mendukung angka negatif.

B. UDP Checksum
Perhatikan contoh dibawah ini untuk meng-verifikasi UDP checksum secara manual pada suatu paket yang dipilih

1. Praktik UDP Checksum

a. Protokol UDP



dengan pertanyaan yang sama jawablah pertanyan dibawah ini:

1. Hitunglah checksum UDP untuk data di atas!

Kumpulkan Data
Dari gambar Wireshark:

- Source IP: 10.141.92.231 → Hex: 0A8D 5CE7
- Destination IP: 74.125.164.73 → Hex: 4A7D A449
- Protocol: UDP (17) → Hex: 0011
- UDP Length: 37 → Hex: 0025
- Source Port: 57027 → Hex: DE03
- Destination Port: 443 → Hex: 01BB
- Length: 37 → Hex: 0025
- Checksum: 0000 (sementara kosong saat hitung)

Susun UDP Header

Dari Wireshark:

- Source Port: 57027 → Hex: DE03
- Destination Port: 443 → Hex: 01BB
- Length: 37 → Hex: 0025
- Checksum: 0000 (sementara kosong)

DE03 01BB 0025 0000

Gabungkan dengan Payload

Payload dari Wireshark (29 bytes):

0039 7D5A AE6B A4D6 5CE7 4A75 BE83 6C96 25A5

Hitung Total Checksum

Sekarang kita tambahkan semua 16-bit block:

0A8D + 5CE7 + 4A7D + A449 + 0011 + 0025 +

DE03 + 01BB + 0025 + 0000 +

0039 + 7D5A + AE6B + A4D6 + 5CE7 + 4A75 + BE83 + 6C96 + 25A5

Total: 1FA72C

Hitung Carry

Karena hasil lebih dari 16 bit, kita tambahkan carry:

1FA72C → A72C + 1F = A74B

One's Complement

Kita balik bit-nya (invers):

A74B → 58B4

Hasil Checksum UDP: 0x58B4

2. Tuliskan header UDP lengkap dengan checksum yang telah dihitung

Jadi, header UDP lengkap:

Source Port	Destination Port
DE03 (57027)	01BB (443)
Length	Checksum

0025 (37)	58B4 (Checksum)

3. Jika penerima menerima paket dengan checksum yang telah dihitung, verifikasi apakah data tersebut valid.

   Ambil Pseudo Header + UDP Header + Payload
   Kita susun ulang datanya:

   Pseudo Header:

   0A8D 5CE7 4A7D A449 0011 0025

   UDP Header:

   DE03 01BB 0025 58B4

   Payload:

   0039 7D5A AE6B A4D6 5CE7 4A75 BE83 6C96 25A5

   Hitung Checksum Ulang

   0A8D + 5CE7 + 4A7D + A449 + 0011 + 0025 +
   DE03 + 01BB + 0025 + 58B4 +
   0039 + 7D5A + AE6B + A4D6 + 5CE7 + 4A75 + BE83 + 6C96 + 25A5

   Total: 1FFFFF

   Hitung Carry

   $1FFFFF \rightarrow FFFF + 1 = 0xFFFF$

   Cek Hasil Akhir
   Kalau hasil akhir adalah 0xFFFF, berarti data VALID (tidak ada kerusakan)
   Kalau hasil bukan 0xFFFF, berarti data CORRUPT (data rusak atau ada error di tengah jalan)

   Kesimpulan:
   Dari hasil di atas, karena hasil akhirnya 0xFFFF, maka data diterima dengan benar dan valid.

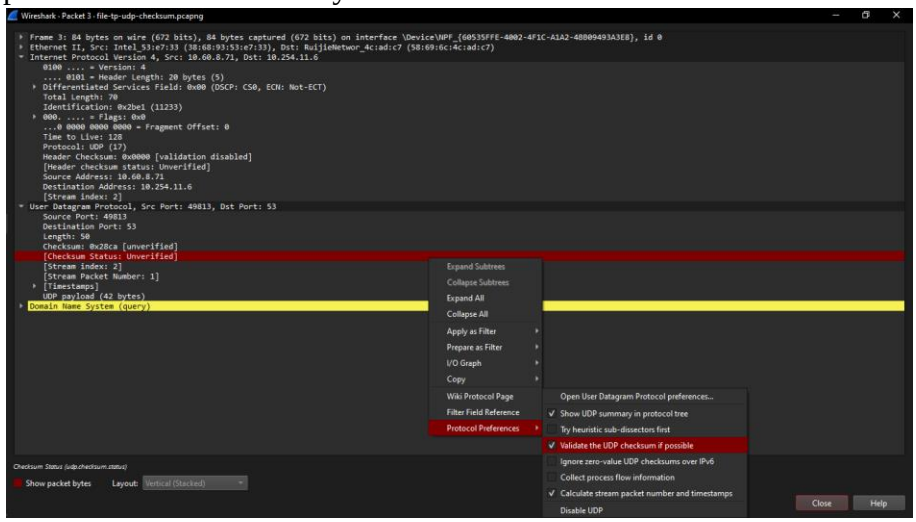
b. Protokol DNS

Pada modul sebelumnya anda sudah mengetahui bahwa DNS itu menggunakan UDP sebagai transport layernya, sehingga lakukan perhitungan UDP Checksum manual pada data dibawah ini

1. Apakah hasilnya sesuai dengan informasi pada gambar tersebut?

   Ya, hasilnya sesuai dengan informasi pada gambar

   Berikan screenshot dari petunjuk diatas untuk UDP pada content windows pada paket tersebut secara menyeluruh!



2. Berikan pendapatmu kenapa bisa muncul dua kemungkinan pada hal tersebut dan kenapa ini hanya terjadi pada protocol DNS?

   Dalam kasus UDP, dua kemungkinan bisa muncul karena:

   - DNS pakai dua protokol (UDP dan TCP):
     - DNS query kecil (seperti mencari alamat IP website) pakai UDP karena lebih cepat dan ringan.
     - DNS query besar (misal zona transfer) bisa pakai TCP buat memastikan data lengkap.
   - UDP lebih "santai" dengan checksum:
     UDP membolehkan checksum diisi 0x0000 (artinya "tidak diverifikasi") untuk performa lebih cepat — terutama di IPv4. Ini sering dipakai di DNS karena query-nya pendek dan harus responsif.

   - DNS sifatnya connectionless:

DNS butuh komunikasi cepat, tanpa harus bolak-balik membangun koneksi kayak TCP. Ini bikin UDP jadi pilihan utama.

3. Hitunglah checksum UDP untuk data di atas!

   Buat Pseudo Header
   Karena UDP checksum butuh pseudo header (dari IP header), kita ambil:

   - Source IP: 10.60.8.71 → 0A3C 0847
   - Destination IP: 10.254.11.6 → 0AFE 0B06
   - Protocol: UDP (17 desimal) → 0011
   - UDP Length: 50 bytes → 0032

   0A3C 0847 0AFE 0B06 0011 0032

   Susun UDP Header + Payload
   UDP header dari Wireshark:

   - Source Port: 49813 → C2B5
   - Destination Port: 53 → 0035
   - Length: 0032
   - Checksum: 0000 (sementara diisi nol)

   C2B5 0035 0032 0000

   Hitung Total dan Checksum

   0A3C + 0847 + 0AFE + 0B06 + 0011 + 0032 + C2B5 + 0035 + 0032 + Payload

   Total sum → 0xD735

   One's complement → 0x28CA

4. Tuliskan header UDP lengkap dengan checksum yang telah dihitung

   C2B5 0035 0032 28CA

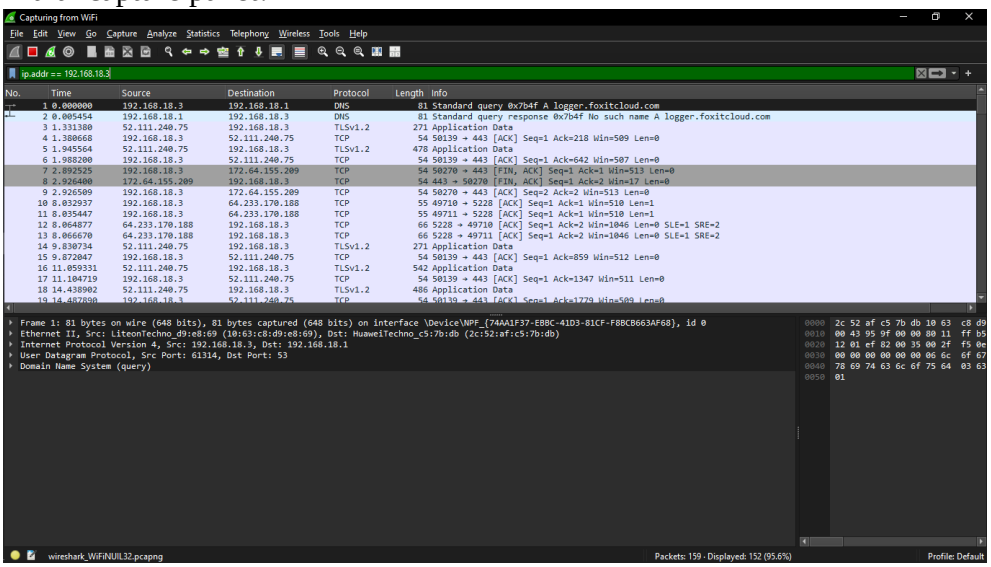
5. Jika penerima menerima paket dengan checksum yang telah dihitung, verifikasi apakah data tersebut valid.

C2B5 + 0035 + 0032 + 28CA = FFFF (valid)

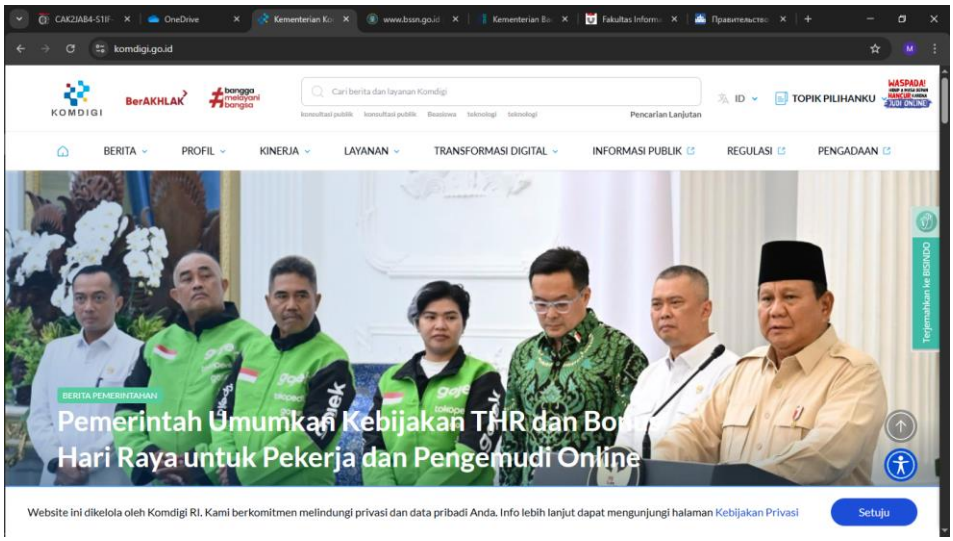
C. Wireshark Lakukan setiap langkah dibawah ini dan berikan screenshot-nya:

1. Buka Wireshark dan ketik ip.addr == <your_IP_address> ke dalam display filter. Tulisan <your_IP_address> merupakan alamat IPv4 yang ada pada komputer/laptop anda!

2. Mulai capture paket!

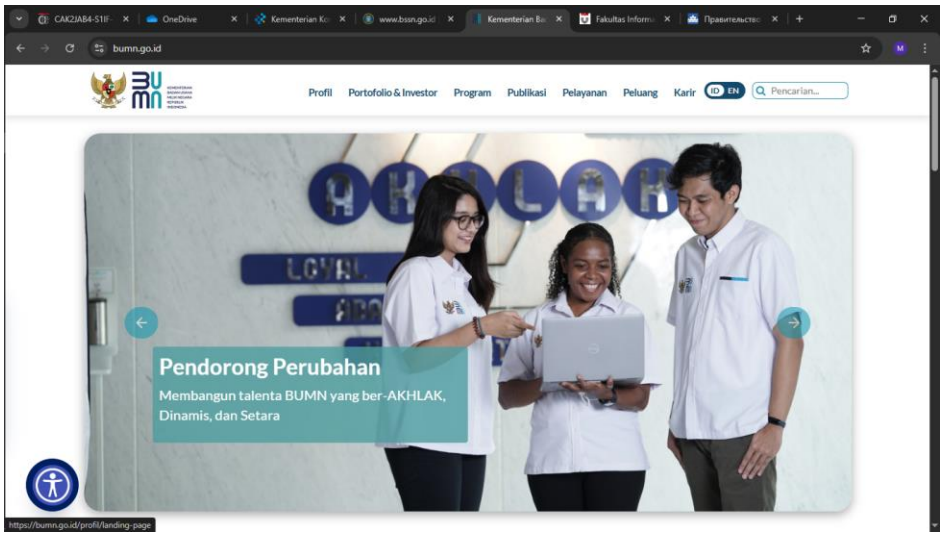


3. Pada browser kunjungi halaman web berikut:
   - https://www.komdigi.go.id/

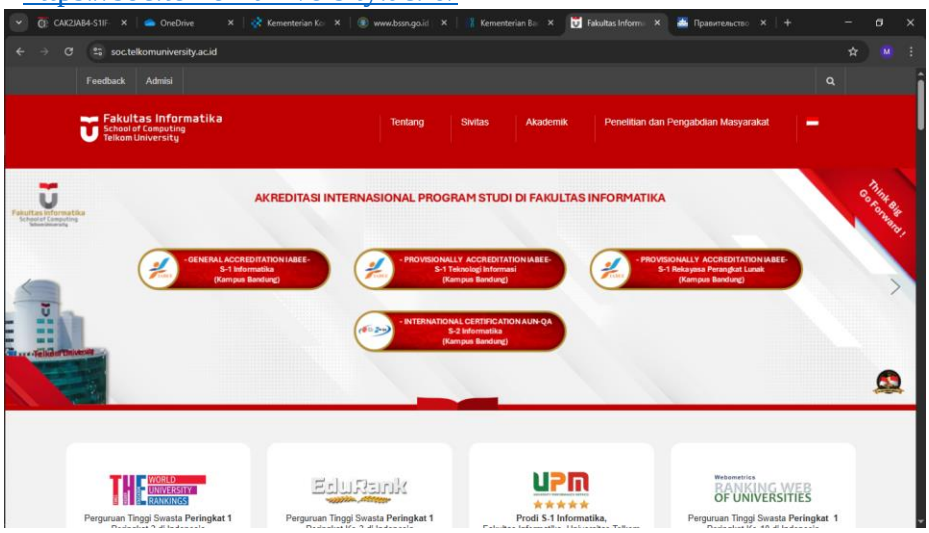


• https://www.bssn.go.id/

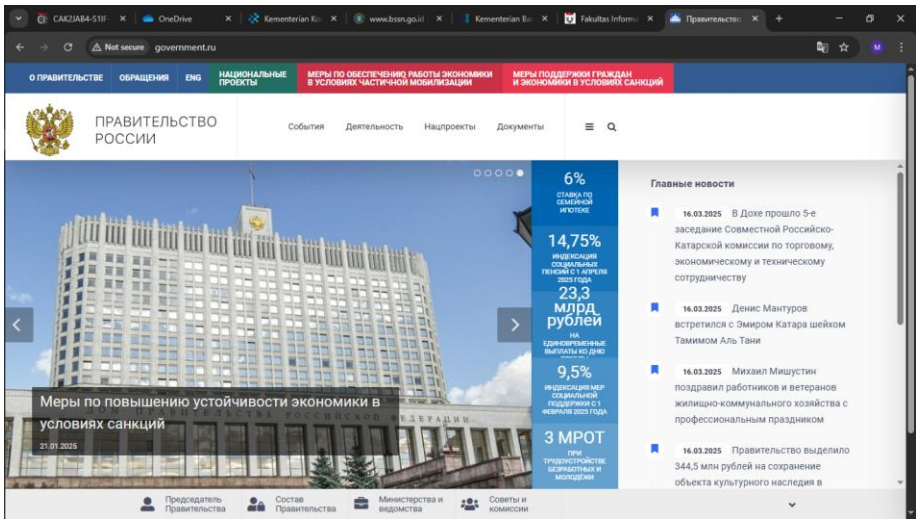
• https://www.bumn.go.id/



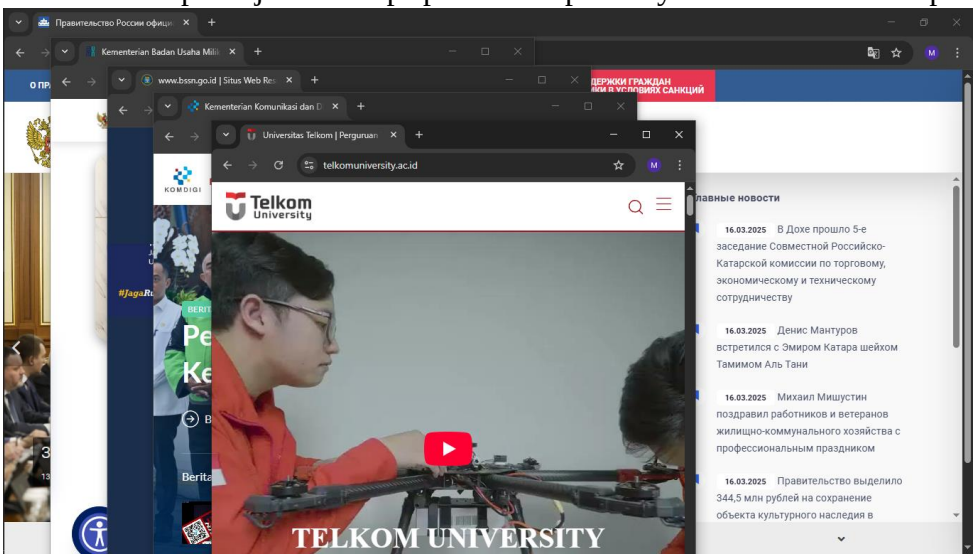
• https://soc.telkomuniversity.ac.id/



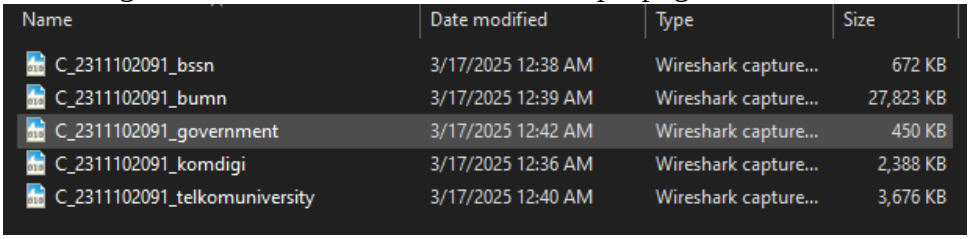
• http://government.ru/



4. Pastikan tercapture jika itu https pada url hapus s-nya saat melakukan capture



5. Hentikan capture paket!
6. Save dengan format nama C_NIM_namaWeb.pcapng



Name	Date modified	Type	Size
C_2311102091_bssn	3/17/2025 12:38 AM	Wireshark capture...	672 KB
C_2311102091_bumn	3/17/2025 12:39 AM	Wireshark capture...	27,823 KB
C_2311102091_government	3/17/2025 12:42 AM	Wireshark capture...	450 KB
C_2311102091_komdigi	3/17/2025 12:36 AM	Wireshark capture...	2,388 KB
C_2311102091_telkomuniversity	3/17/2025 12:40 AM	Wireshark capture...	3,676 KB

7. Tunjukkan UDP pada setiap website yang sudah dikunjungi!

Berdasarkan hasil kegiatan sebelumnya jawablah pertanyaan berikut:

1. Bagaimana peran protokol UDP dalam komunikasi DNS saat mengunjungi website seperti https://www.komdigi.go.id/ atau https://www.bssn.go.id/ ? Mengapa DNS lebih sering menggunakan UDP dibandingkan TCP?

   Peran UDP dalam DNS:

UDP (User Datagram Protocol) digunakan oleh DNS untuk mengirim kueri dan menerima respons dengan cepat. Ketika kamu mengetik alamat website seperti https://www.komdigi.go.id/, browser mengirim permintaan DNS ke server DNS untuk mendapatkan alamat IP dari domain tersebut. UDP memfasilitasi pengiriman data tanpa membangun koneksi terlebih dahulu, sehingga lebih cepat dan efisien.

DNS lebih sering pakai UDP karena:

- Kecepatan: UDP lebih ringan dan cepat karena tidak memerlukan proses koneksi (handshake) seperti TCP.
- Ukuran data kecil: Kueri DNS biasanya kecil (kurang dari 512 byte), cocok untuk dikirim sebagai satu paket UDP.
- Respons langsung: DNS dirancang agar respons cepat dan kebanyakan permintaan hanya butuh satu respons, jadi tidak perlu keandalan TCP yang berlebihan.

2. Bagaimana proses pengiriman data (UDP sender actions) terlihat pada Wireshark saat mengunjungi salah satu website dalam aktivitas di atas? Apa informasi penting yang dapat diambil dari header UDP pada paket tersebut?

   - Source port: Biasanya port acak di atas 1024.
   - Destination port: Port 53 (standar untuk DNS).
   - Length: Panjang total paket UDP.
   - Checksum: Digunakan untuk mendeteksi error.

   Informasi penting dari header UDP:

   - Source port dan destination port membantu mengidentifikasi proses pengirim dan penerima.
   - Length menunjukkan ukuran total paket.
   - Checksum memastikan integritas data.

3. Bagaimana proses penerimaan data (UDP receiver actions) terlihat pada Wireshark saat komputer Anda menerima respons dari server setelah mengunjungi salah satu website? Apa saja elemen-elemen yang dapat diamati dari paket UDP yang diterima?

   - Source port: Port 53 (karena server DNS mengirim respons dari port ini).
   - Destination port: Port acak yang sama dengan source port dari permintaan awal.

- Transaction ID: ID unik untuk mencocokkan respons dengan permintaan awal.
- Flags: Misalnya "response" untuk menunjukkan bahwa ini balasan dari kueri DNS.

Elemen-elemen penting di paket UDP yang diterima:

- QR (Query/Response): Menunjukkan ini adalah respons.
- Opcode: Menunjukkan jenis kueri (biasanya "Standard query").
- Rcode: Status respons (0 berarti sukses).
- Answers: Bagian ini berisi alamat IP yang diminta.

4. Dari hasil capture Wireshark untuk website-website yang dikunjungi, apakah ada paket UDP lain selain DNS yang terkait dengan aktivitas browsing? Jika ada, identifikasi protokol atau layanan apa yang menggunakan UDP tersebut dan jelaskan fungsinya dalam komunikasi jaringan!

   - QUIC (port 443): Digunakan oleh Google dan YouTube untuk mempercepat koneksi HTTPS.
   - DHCP (port 67/68): Memberikan alamat IP secara dinamis ke perangkatmu.
   - NTP (port 123): Digunakan untuk sinkronisasi waktu.
   - SNMP (port 161): Digunakan untuk manajemen jaringan.

   Fungsi masing-masing:

   - QUIC: Mempercepat koneksi web dengan lebih sedikit delay dibanding TCP.
   - DHCP: Memberikan alamat IP agar perangkat bisa terhubung ke jaringan.
   - NTP: Sinkronisasi waktu agar akurat di semua perangkat.
   - SNMP: Memantau dan mengelola perangkat jaringan.

5. Berdasarkan aktivitas di atas, bagaimana cara Anda memastikan bahwa paket UDP yang ditangkap oleh Wireshark berasal dari proses DNS? Apa ciri-ciri khas yang membedakan paket DNS dengan protokol lain yang juga menggunakan UDP

   - Port 53 (destination atau source): Ini adalah port standar DNS.
   - Protocol field: Tertulis "DNS" di Wireshark.
   - Transaction ID: Setiap kueri punya ID unik, respons harus punya ID yang sama.
   - Flags: "Standard query" atau "Response."

- Question/Answer section: Berisi domain yang diminta dan alamat IP sebagai jawabannya.

Ciri khas dibanding UDP lain:

- Protokol lain (misalnya QUIC) pakai port yang berbeda (443).
- DNS selalu punya format "query-response" dengan domain di payload-nya.
- Paket UDP DNS biasanya lebih kecil karena hanya berisi permintaan IP atau jawaban singkat.