Nama : Maisy Rahmawati

NPM : 1806147035

Kelas : Jarkomdat – A

**Log Week 03 – Applciation Layer: Principles of network applications &**

**Web and HTTP**

Materi yang cukup dipahami:

* Network app, yaitu aplikasi jaringan seperti e-mail, website, text messaging, remote login, perr-to-peer file sharing, streaming stored video (YouTube dan Netflx), games, real-time video conferencing, voice over IP (skype), dan lain-lain. Network apps dibangun melalui pembuatan suatu program yang dijalankan (di-running) pada end systems (host) yang berbeda. Dengan adanya network apps maka komunikasi dapat dilakukan melalui jaringan, contohnya komunikasi antara web server dengan web browser. Dalam membuat network apps, tidak perlu adanya penulisan program untuk device dari network-core karena posisi network-core tidak menjalankan user applications.
* Struktur aplikasi (Application architectures) yang mungkin dapat terbentuk dari pembuatan network apps adalah client-server dan peer-to-peer (P2P). Client-server memungkinkan pengiriman objek dengan sistem request-response, di mana client akan melakukan request kepada server melalui HTTP request message untuk mendapatkan suatu objek dan server akan meresponse permintaan dari client melalui HTTP response message. Sedangkan pada P2P, hubungannya bisa bergantian, satu pihak dapat menjadi server atau pun client. Pada struktur P2P, antar peer bebas berbagi informasi.
* Pada client-server architectures, server akan selalu bertidak sebagai host yang aktif dengan IP Address yang tetap sepanjang masa dan merupakan pusat data untuk scaling. Sedangkan client adalah pihak yang berkomunikasi dengan server dengan IP Address yang relatif dinamis (dapat berubah-ubah) dan client tidak dapat berkomunikasi secara langsung dengan sesama client.
* Pada P2P architectures, server tidak akan selamanya menjadi server, dia dapat menjadi client. Peer akan meminta request service kepada peer lainnya dan peer lainnya mengembalikan (return) request, atau disebut sebagai self scalability yaitu peer baru akan menghadirkan kapasitas new service, serta demand dari new service. Peers akan terhubung dan mereka dapat melakukan perubahan pada manajemen IP Address yang kompleks.
* Processes communicating, ditandai dengan program yang berjalan di dalam host. Ada dua macam processes communicating, yaitu dengan host yang sama dan dengan host yang berbeda. Jika dilakukan dengan host yang sama, maka dua proses berkomunikasi menggunakan komunikasi antar-proses yang ditentukan oleh OS. Sedangkan jika hostnya berbeda, maka komunikasi dilakukan dengan bertukar pesan.
* Client process (terjadi pada sisi client) adalah proses di mana komunikasi dimulai. Sedangkan server process (terjadi pada sisi server) adalah proses yang menunggu untuk dihubungi dalam sebuah komunikasi.
* Perbedaan antara TCP dan UDP. TCP merupakan protokol yang bisa dikatakan cukup sensitif karena tidak memperbolehkan adanya packet loss sedikit pun. Sedangkan UDP adalah protokol yang memperbolehkan adanya packet loss, di mana UDP sendiri bertujuan untuk mempercepat transfer data dan UDP memerlukan adanya throughput minimal. TCP sangat reliable dibandingkan dengan UDP karena TCP dapat menghandle packet loss di mana transfer data harus 100% dilakukan. TCP banyak digunakan dalam transfer file (dengan protokol FTP), web browser (dengan protokol HTTP), dan mengirim email (dengan protokol SMTP). Sementara UDP banyak digunakan ketika melakukan streaming audio atau video dan mengirim file multimedia seperti gambar. Namun, itu tidak berarti jika kita membuka streaming multimedia, hanya UDP saja yang digunakan. Banyak juga penggunaan UDP sekaligus TCP dalam streaming multimedia, seperti Youtube.
* SSL (Secure Socket Layer) merupakan protokol kriptografi yang paling banyak digunakan untuk memberikan keamanan melalui komunikasi internet. SSL menyediakan koneksi yang terenkripsi pada TCP di mana transmisi data menjadi terlindungi, misalkan dari client atau web browser ke server atau dari server ke server. SSL juga menyediakan authentication di mana SSL akan memastikan server yang web browser sambungkan adalah server yang benar. Selain itu, SSL juga memastikan adanya integrasi data, di mana data yang diminta atau disampaikan adalah data yang benar-benar dikirimkan.
* Secara umum, konsep besar dari HTTP persistent dan HTTP non persistent sudah cukup dipahami, dimana HTTP persistent memungkinkan client (web browser) tetap terkoneksi dengan server untuk dapat mengirimkan request berkali-kali artinya open connection tetap berlangsung. Sementara HTTP non persistent memungkinkan koneksi client (web browser) langsung terputus tepat setelah server mengirimkan response atas suatu request yang clinet (web browser) minta. Oleh sebab itu, pada HTTP non persistent, client (web browser) perlu melakukan open connection lagi ketika akan mengirimkan request kepada server.

Materi yang masih belum dipahami:

* Konsep dari socket
* Konsep dari addressing processes
* Terkait dengan penggunaan wireshark, terdapat beberapa tools yang masih bingung
* Konsep tentang RTT dan perhitungannya dalam HTTP persistent dan HTTP non persistent

Pertanyaan yang masih menjadi PR selama pembelajaran week 03 dan penggunaan wireshark:

1. Bagaimana kita membedakan antara terjadinya open connection dan close connection di dalam wireshark? Apakah kita dapat membedakannya di dalam wireshark?
2. Bagiamana contoh konkrit dalam kehidupan sehari-hari dari HTTP persistent dan HTTP non persistent?
3. Apa maksudnya open connection dan close connection? Apakah ketika kita mulai membuka halaman website maka dapat dikatakan bahwa kita melakukan open connection? Lalu apakah jika kita menutup halaman website maka kita melakukan close connection?
4. Apakah mungkin terjadi client mengirimkan multiple request message kepada server tetapi server hanya mengirimkan satu response message kepada client? Apakah mungkin juga client mengirimkan satu request message kepada server dan server akan mengirimkan multiple response message kepada client?
5. Ketika melakukan percobaan capture packet dengan wireshark, dilakukan percobaan analisis capture untuk protokol HTTP, di mana web browser membuka halaman website [www.duniailkom.com](http://www.duniailkom.com). Namun, ketika dimasukkan parameter “http” di dalam kotak display-filter, tidak ada yang ditampilkan, mengapa hal tersebut dapat terjadi?