KEAMANAN WEB BROWSER

Web Browser

The Old Days
Image + Text

Nowadays
Run CGI scripts on Web server
Run Java Script and VBScript
Java Aplet and ActiveX
Plugins

Web Browser Risks

Web server bisa jadi tidak aman

- Batasi pengiriman data sensitif ke web server yang tidak aman

Browser menjalankan *malcode* dalam bentuk *scripts* atau *executables*

Penyerang bisa menyadap trafik jaringan

- Resiko penyadapan dapat dikurangi melalui penggunaan Secure Sockets Layer (SSL) untuk meng-enkripsi data yang ditransmisikan

Penyerang dapat menjalankan serangan *man-in-the-middle attack*

- Aplikasi web yang sessionless sangat potensial untuk menjadi korban serangan *man-in-themiddle attacks* seperti *hijacking and replay*.

Cara kerja Web Browser

Protokol utama yang digunakan Web browser adalah HTTP (Hypertext Transfer Protocol)

HTTP merupakan protokol layer aplikasi yang memungkinkan Web browser untuk meminta halaman web dari dan mengirimkan informasi ke Web server

Web server akan merespons dengan mengirimkan:

- Kode HTML
- Gambar
- Scripts
- Executables

HTTP adalah protokol *stateless* → Tidak mengingat session sebelumnya

Cookies

- Cookies adalah sarana penyimpan informasi yang dibuat oleh suatu Website untuk menyimpan informasi tentang user yang mengunjungi situs yang bersangkutan
  - Penyimpanan informasi ini demi kenyamanan user maupun Website
  - Informasi sensitif dan pribadi merupakan perhatian dari sisi keamanan
- Cookies merupakan file ASCII yang dikirimkan server ke client, lalu client menyimpannya di sistem lokal
- Ketika request baru dikirimkan, server dapat meminta browser untuk mencek apakah ada cookies, jika ada, web server dapat meminta web browser untuk mengirimkan cookie ke web server
  - Cookies ini bisa berasal dari manapun
- Isi cookies dikendalikan oleh Web server dan bisa mengandung informasi mengenai kebiasaan kita ketika mengakses internet
- Informasi yang diperoleh Web server berasal dari Web browser
  - Diperoleh ketika user mengisi form yang mengharuskan memasukkan nama dan e-mail address
    - Informasi ini dapat disimpan di cookie untuk keperluan di masa datang

Cookies: Macam-macam cookies

Persistent Cookies

- Bertahan di local system untuk waktu yang lama (walaupun sistem sudah direboot)
- Tersimpan di dalam hard disk dalam bentuk file *cookies.txt*
  - File ini dapat dibaca dan diedit oleh user atau *system administrator*
  - File ini bisa mengandung informasi yang sensitif
  - Jika suatu saat seorang penyerang dapat masuk ke dalam workstation tempat menyimpan cookie maka penyerang dapat menggunakannya untuk melakukan serangan berikut
  - Karena bisa dimodifikasi maka file cookies dapat digunakan untuk melakukan penyerangan hijacking atau replay attack.
- Kelemahan:
  - File cookies bisa memenuhi hard disk
  - Riwayat kebiasaan surfing dapat digunakan oleh pihak lain

Cookies: Macam-macam cookies

Non-persistent cookies

- Karena kelemahan persistent cookies, banyak situs yang sekarang menggunakan non-persistent cookies
- Nonpersistent cookies disimpan di memori sehingga bila komputer dimatikan cookies-nya juga akan hilang
- Tidak ada jaminan bahwa setiap browser akan dapat menangani nonpersistent cookies secara benar

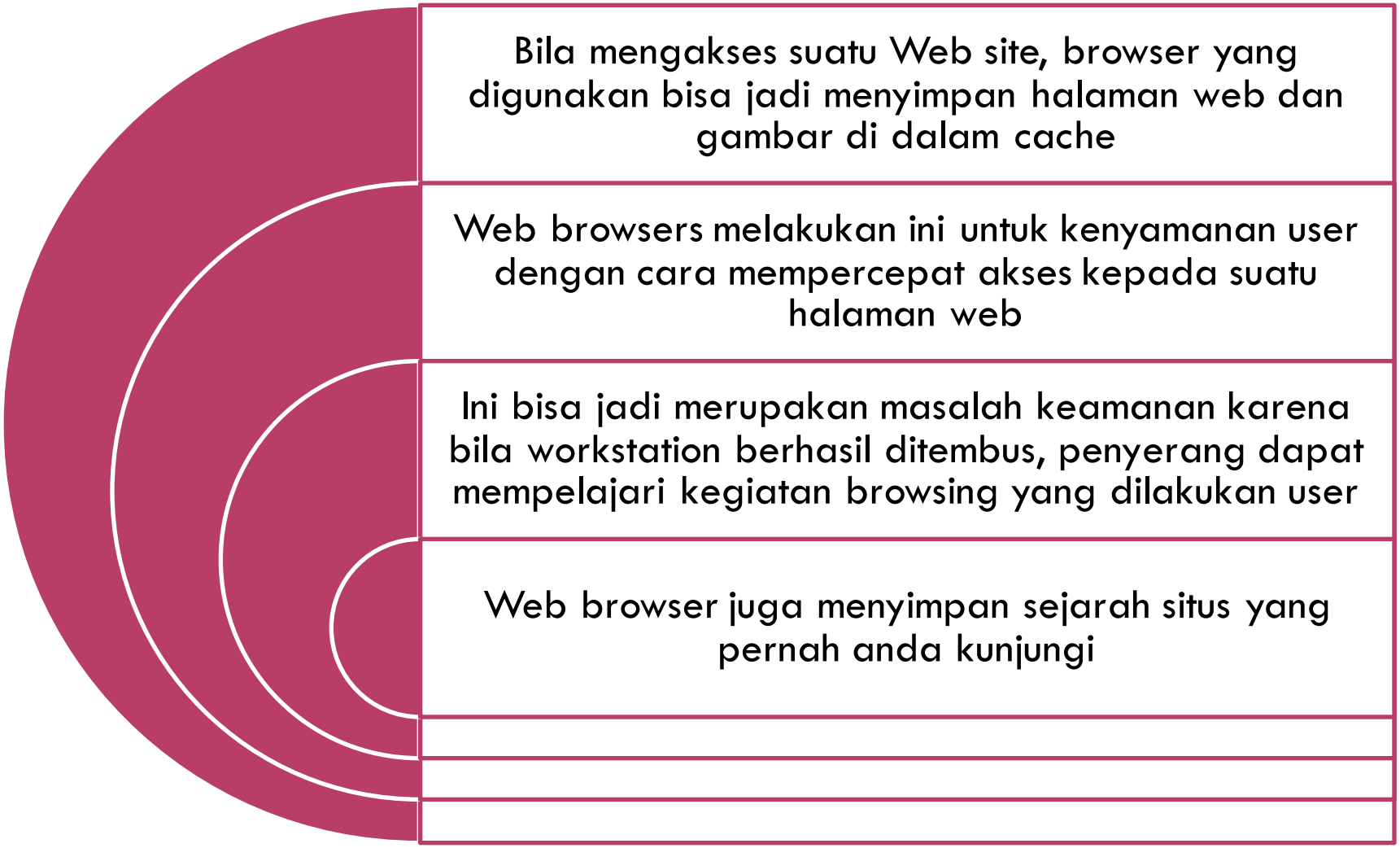
Cookies

Cookies pada umumnya mengandung informasi yang memungkinkan suatu web site dapat mengingat user yang mengunjungi situs tersebut

Informasi yang paling sering disimpan di cookies adalah :

- Session ID
  - Digunakan untuk memaintain state atau membawa informasi otorisasi antar request yang dilakkan web browser
- Waktu dan tanggal dikeluarkannya cookie
- Waktu dan tanggal kadaluarsa cookies
  - Digunakan oleh Web site untuk menentukan pengabaian cookies yang lama
- IP address dari browser
  - Dapat dianggap sebagai uji tambahan terhadap otentikasi request

Caching


Bila mengakses suatu Web site, browser yang digunakan bisa jadi menyimpan halaman web dan gambar di dalam cache
Web browsers melakukan ini untuk kenyamanan user dengan cara mempercepat akses kepada suatu halaman web
Ini bisa jadi merupakan masalah keamanan karena bila workstation berhasil ditembus, penyerang dapat mempelajari kegiatan browsing yang dilakukan user
Web browser juga menyimpan sejarah situs yang pernah anda kunjungi

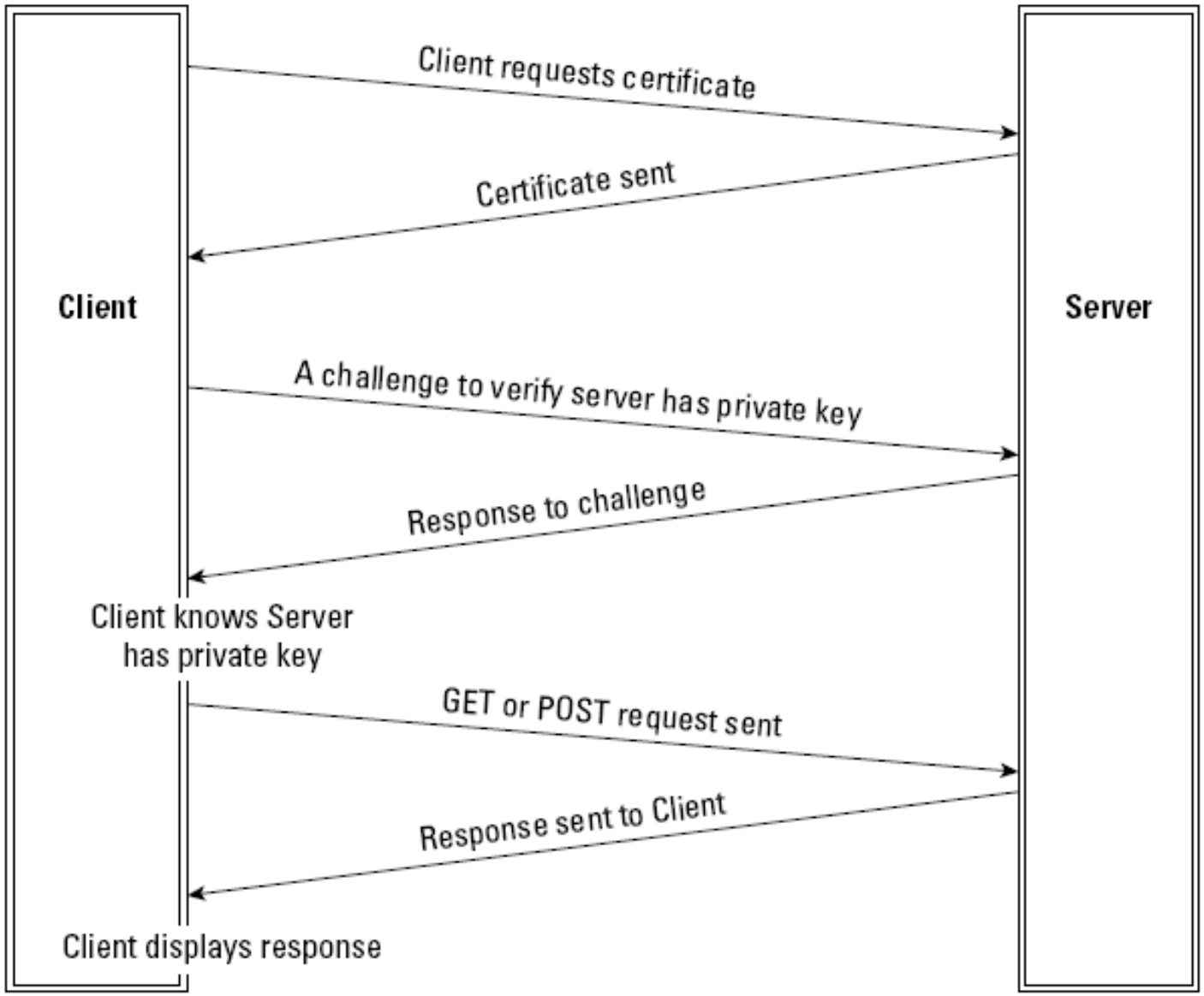
Secure Socket Layer

- SSL dikembangkan oleh Netscape untuk menyediakan layanan keamanan pada pengiriman informasi melalui Internet
- SSL memandaatkan *asymmetric* maupun *symmetric key encryption* untuk membentuk dan mentransfer data pada link komunikasi yang aman pada jaringan yang tidak aman
- Bila digunakan pada browser di client, SSL membentuk suatu koneksi yang aman antara browser pada client dengan server
  - Biasanya berbentuk HTTP over SSL (HTTPS)
  - Proses ini membentuk suatu *encrypted tunnel* antara browser dengan Web server yang dapat digunakan untuk mengirimkan data
- Pihak yang berusaha menyadap koneksi antara browser dengan server tidak dapat men-dekripsi informasi yang dipertukarkan di antara keduanya
- Integritas informasi dibentuk menggunakan algoritma hashing
- *Confidentiality* dijamin oleh enkripsi

Session SSL yang tipikal

1. Browser atau client meminta sertifikat server
2. Server mengirimkan sertifikat ke browser
3. Setelah menerima sertifikat server, browser akan mencek apakah sertifkat berasal dari certificate authority (CA) yang dapat dipercaya
   - Web browser mengirimkan challenge ke server untuk menjamin bahwa server memiliki private key yang sesuai dengan public key yang ada di dalam sertifikat
   - Challenge ini mengandung *kunci simetris* yang akan digunakan untuk mengenkripsi trafik SSL traffic
   - Hanya pemiliki private key yang dapat mendekripsi challenge.
4. Web server merespon challenge dengan suatu pesan pendek yang dienkripsi menggunakan kunci simetris yang diperoleh dari challenge. Setelah menerima message ini, browser yakin bahwa dia berkomunikasi dengan organisasi yang tepat
5. Sekarang browser dan server memiliki kunci simteris yang sama
6. Setiap message dari browser dapat dienkripsi menggunakan kunci simetris
   - Web server menggunakan kunci simteris yang sama untuk mendekripsi trafik
7. Dengan proses yang serupa, setiap response dari server akan dienkripsi menggunakan kunci simetris

Session SSL tipikal



SSL

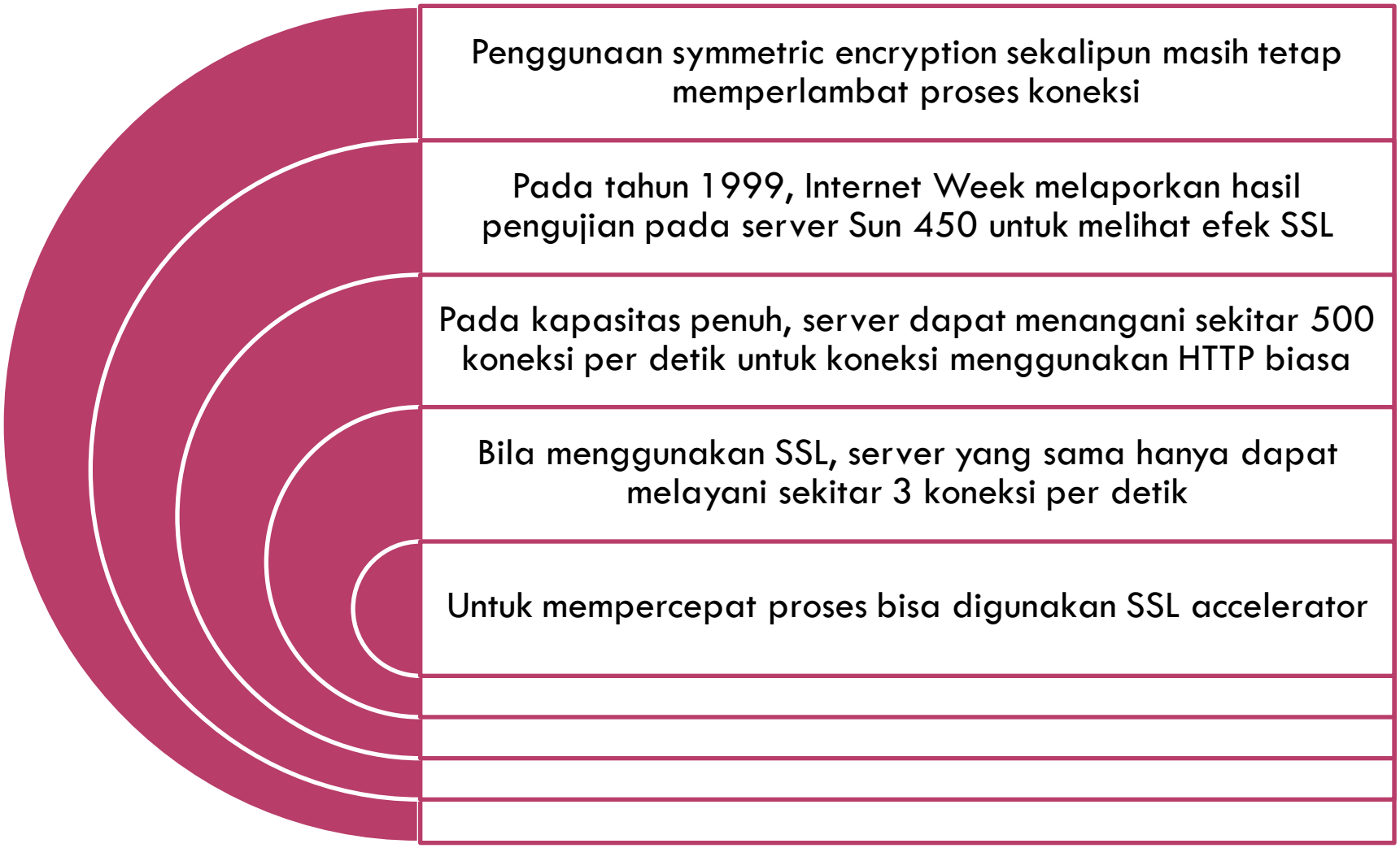
- Pada umumnya software Web server mendukung HTTPS
- Situs-situs e-commerce biasanya menggunakan HTTPs juga untuk memperoleh informasi rahasia user
- Server diotentikasi oleh client melalui sebuah *digital certificate*
- Kekuatan enkripsi biasanya ditentukan oleh server tetapi biasanya dipilih berdasarkan kemampuan browser client
- Tipe enkripsi yang digunakan di dalam browser tergantung kepada apakah browser tersebut akan dijual di USA atau di luar USA
- Versi enkripsi untuk pemakaian di luar USA menggunakan *weak encryption*
- Pemakaian enkripsi untuk di dalam USA menggunakan strong encryption
- Weak encryption: 40-bit or 56-bit encryption
- Strong encryption : 128-bit encryption
  - 128-bit encryption adalah 300,000,000,000,000,000,000,000,000 lebih kuat daripada 40-bit encryption.
- Netscape menyediakan dua versi browser : strong (in USA) and weak encryption (outside USA)
- Internet explorer default-nya menyediakan weak encryption
  - Kalau ingin strong encryption harus di-patch

SSL

Web browser yang dikonfigurasi dengan baik akan memperingatkan user akan adanya masalah pada sertifikat server bila menemui terjadinya hal-hal berikut:

- Sertifikat tidak ditandatangani oleh suatu *recognized certificate authority*
- Sertifikat invalid atau kadaluarsa
- Common name sertifikat tidak match dengan nama domain server

SSL


Penggunaan symmetric encryption sekalipun masih tetap memperlambat proses koneksi
Pada tahun 1999, Internet Week melaporkan hasil pengujian pada server Sun 450 untuk melihat efek SSL
Pada kapasitas penuh, server dapat menangani sekitar 500 koneksi per detik untuk koneksi menggunakan HTTP biasa
Bila menggunakan SSL, server yang sama hanya dapat melayani sekitar 3 koneksi per detik
Untuk mempercepat proses bisa digunakan SSL accelerator

Secure HTTP (SHTTP)

- SHTTP merupakan pengembangan dari protokol HTTP yang dikembangkan oleh Enterprise Integration Technologies
- Secara fungsional SHTTP serupa dengan HTTPS yaitu sama-sama dirancang untuk menyediakan transaksi dan message yang aman melalui Web
- Beberapa perbedaan SHHTP dengan HTTPS:
  - SSL bersifat connection-oriented dan bekerja pada layer transport
  - SSL membentuk koneksi yang aman yang dapat dilalui oleh message
  - SHTTP bersifat transaction-oriented dan bekerja pada layer aplikasi
  - Pada SHHTP setiap message dienkripsi agar aman ketika ditransmisikan
  - Tidak ada pipa aman yang dibentuk antar entitas
  - SSL dapat digunakan bersama protokol TCP/IP yang lain seperti FTP dan TELNET
  - SHTTP khusus untuk HTTP
  - HTTPS banyak diterapkan sedangkan penggunaan SHTTP hanya terbatas (tidak semua web browser mendukung SHTTP)

Web Browser Attacks

Hijacking

- Man-in-the-middle attack; penyerang mengambil alih session

Replay

- Man-in-the-middle attack; data yang dikirim diulangi (replayed)

Penyebaran malcode (viruses, worms, dsb.)

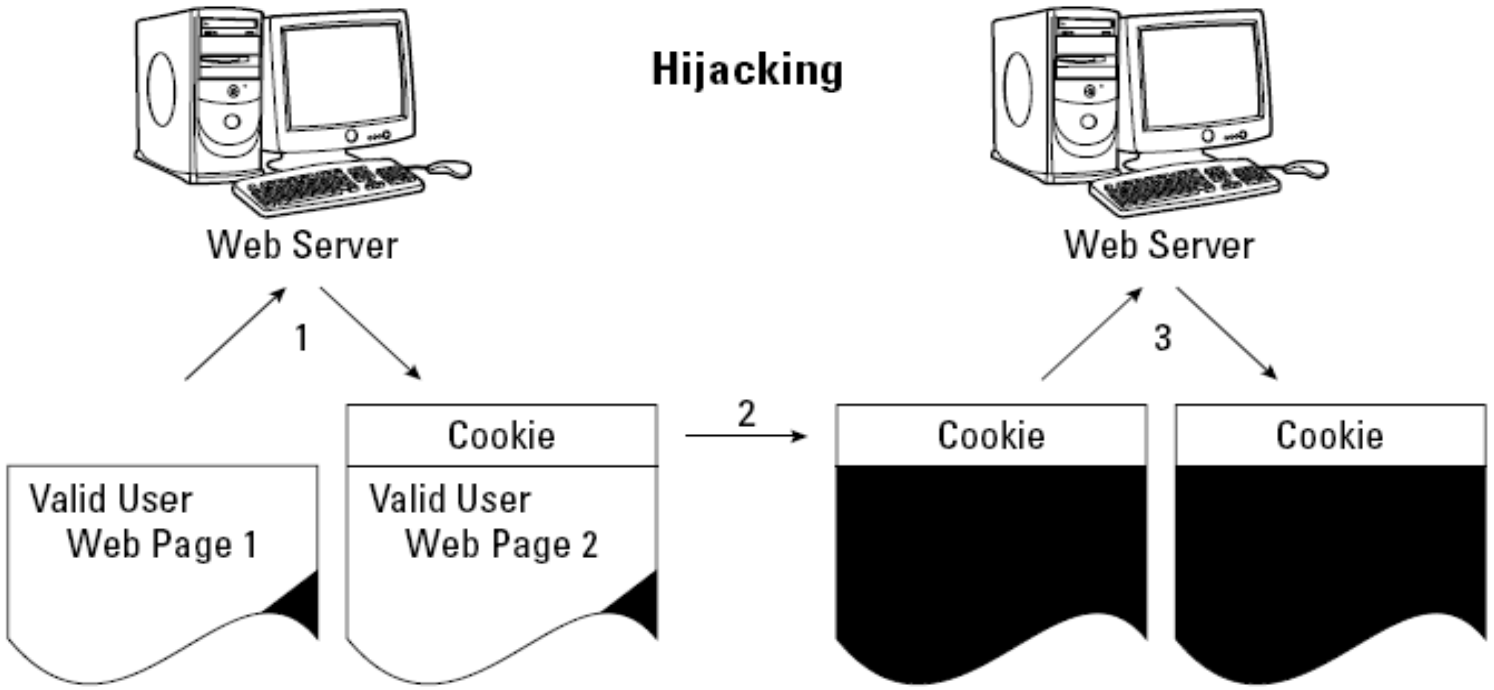
Menjalankan *executables* yang berbahaya pada host

Mengakses file pada host

- Beberapa serangan memungkinkan browser mengirimkan file ke penyerang
- File dapat mengandung informasi personal seperti data perbankan, passwords dsb.

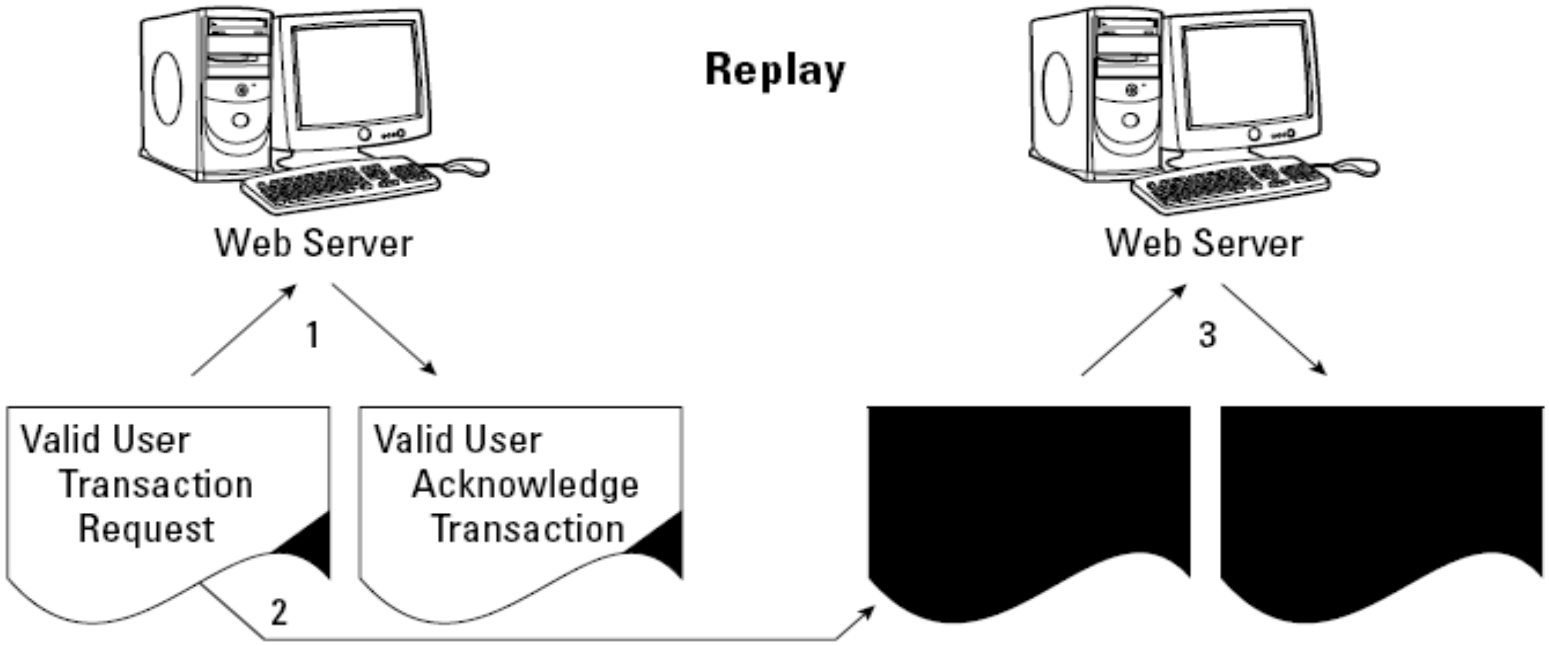
Pencurian informasi pribadi

Hijacking Attack



1. A valid user does some web activity that results in their acquiring a Cookie.
2. The Cookie is stolen or captured by an attacker.
3. The Cookie is transmitted with the attacker's attempt to access the application. The Cookie authenticates the attacker as a valid user. The attacker gets access to the application.

Replay Attack



1. A valid user does some web activity such as "Transfer $5,000 from account A to account B". There may or may not be a cookie.
2. The web page holding the transaction request is stolen or captured by an attacker.
3. The web page is re-transmitted. The transaction is repeated - an additional $5,000 is transferred. The attacker can re-transmit numerous times.
4. Depending on whether the attacker had to do spoofing, the final acknowledgment transaction may go back to the valid user's IP address where it is dropped because no session is open.

Langkah-langkah Untuk Menaikkan Tingkat Keamanan Browser

- Selalu mengupdate web browser menggunakan patch terbaru
- Mencegah virus
- Menggunakan situs yang aman untuk transaksi finansial dan sensitif
- Menggunakan *secure proxy*
- Mengamankan lingkungan jaringan
- Tidak menggunakan informasi pribadi
- Hati-hati ketika merubah setting browser

General Recommendations

- Hati-hati ketika merubah konfigurasi browser
- Jangan membuat konfigurasi yang mendukung scripts dan macros
- Jangan langsung menjalankan program yang anda download dari internet
- Browsing ke situs-situs yang aman
  - Mengurangi kemungkinan adanya malcode dan spyware
- Konfigurasi homepage harus hati-hati
  - Lebih baik gunakan blank.
- Jangan mempercayai setiap links (periksa dulu arah tujuan link itu)
- Jangan selalu mengikuti link yang diberitahukan lewat e-mail
- Jangan browsing dari sistem yang mengandung data sensitif
- Lindungi informasi anda kalau bisa jangan gunakan informasi pribadi pada web
- Gunakan stronger encryption
  - Pilih 128-bit encryption
- Gunakan browser yang jarang digunakan
  - Serangan banyak dilakukan pada web browser yang populer
- Minimalkan penggunaan plugins
- Minimalkan penggunaan cookies
- Perhatikan cara penanganan dan lokasi penyimpanan *temporary files*

TERIMA KASIH