BAB V
Keamanan Sistem World Wide Web

Pendahuluan

- World Wide Web (WWW atau Web1) merupakan salah satu “killer applications” yang menyebabkan populernya Internet. WWW dikembangkan oleh Tim Berners-Lee ketika bekerja di CERN (Swiss).

- Kehebatan Web adalah kemudahannya untuk mengakses informasi, yang dihubungkan satu dengan lainnya melalui konsep *hypertext*. Informasi dapat tersebar di mana-mana di dunia dan terhubung melalui *hyperlink*. Informasi lebih lengkap tentang WWW dapat diperoleh di web W3C <http://www.w3.org>.

- Pembaca atau peraga sistem WWW yang lebih dikenal dengan istilah *browser* dapat diperoleh dengan mudah, murah atau gratis. Contoh browser adalah Netscape, Internet Explorer, Opera, kfm (KDE file manager di sistem Linux), dan masih banyak lainnya. Kemudahan penggunaan program browser inilah yang memicu populernya WWW. Sejarah dari browser ini dimulai dari browser di sistem komputer NeXT yang kebetulan digunakan oleh Berners-Lee. Selain browser NeXT itu, pada saat itu baru ada browser yang berbentuk text (text-oriented) seperti “line mode” browser.

WWW – World Wide Web

- Berkembangnya WWW dan Internet menyebabkan pergerakan sistem informasi untuk menggunakannya sebagai basis. Banyak sistem yang tidak terhubung ke Internet tetapi tetap menggunakan basis Web sebagai basis untuk sistem informasinya yang dipasang di jaringan Intranet. Untuk itu, keamanan sistem informasi yang berbasis Web dan teknologi Internet bergantung kepada keamanan sistem Web tersebut.

- Arsitektur sistem Web terdiri dari dua sisi: server dan client. Keduanya dihubungkan dengan jaringan komputer (computer network). Selain menyajikan data-data dalam bentuk statis, sistem Web dapat menyajikan data dalam bentuk dinamis dengan menjalankan program. Program ini dapat dijalankan di server (misal dengan CGI, servlet) dan di client (applet, Javascript). Sistem server dan client memiliki permasalahan yang berbeda.

Asumsi WWW

- Ada asumsi dari sistem Web ini. Dilihat dari sisi pengguna:
  - Server dimiliki dan dikendalikan oleh organisasi yang mengaku memiliki server tersebut. Maksudnya, jika sebuah server memiliki domain www.bni.co.id dan tulisan di layar menunjukkan bahwa situs itu merupakan milik Bank BNI maka kita percaya bahwa server tersebut memang benar milik Bank BNI. Adanya domain yang dibajak merupakan anomali terhadap asumsi ini.

  - Dokumen yang ditampilkan bebas dari virus, trojan horse, atau itikad jahat lainnya. Bisa saja seorang yang nakal memasang virus di web nya. Akan tetapi ini merupakan anomali.

  - Server tidak mendistribusikan informasi mengenai pengunjung (user yang melakukan browsing) kepada pihak lain. Hal ini disebabkan ketika kita mengunjungi sebuah web site, data-data tentang kita (nomor IP, operating system, browser yang digunakan, dll.) dapat dicatat. Pelanggaran terhadap asumsi ini sebetulnya melanggar privacy. Jika hal ini dilakukan maka pengunjung tidak akan kembali ke situs ini.

Asumsi WWW

- **Asumsi dari penyedia jasa (webmaster) antara lain:**

  • Pengguna tidak beritikad untuk merusak server atau mengubah isinya (tanpa ijin).

  • Pengguna hanya mengakses dokumen-dokumen atau informasi yang diijinkan diakses. Seorang pengguna tidak mencoba-coba masuk ke direktori yang tidak diperkenankan (istilah yang umum digunakan adalah "*directory traversal*").

  • Identitas pengguna benar. Banyak situs web yang membatasi akses kepada user-user tertentu. Dalam hal ini, jika seorang pengguna "*login*" ke web, maka dia adalah pengguna yang benar.

Asumsi WWW

- **Asumsi kedua belah pihak:**
  - Jaringan komputer (network) dan komputer bebas dari penyadapan pihak ketiga.
  - Informasi yang disampaikan dari server ke pengguna (dan sebaliknya) terjamin keutuhannya dan tidak dimodifikasi oleh pihak ketiga yang tidak berhak.

Asumsi-asumsi di atas bisa dilanggar sehingga mengakibatkan adanya masalah keamanan.

Asumsi WWW

- **Asumsi kedua belah pihak:**
  - Jaringan komputer (network) dan komputer bebas dari penyadapan pihak ketiga.
  - Informasi yang disampaikan dari server ke pengguna (dan sebaliknya) terjamin keutuhannya dan tidak dimodifikasi oleh pihak ketiga yang tidak berhak.

Asumsi-asumsi di atas bisa dilanggar sehingga mengakibatkan adanya masalah keamanan.

Keamanan Server WWW

- Keamanan server WWW biasanya merupakan masalah dari seorang administrator. Dengan memasang server WWW di sistem anda, maka anda membuka akses (meskipun secara terbatas) kepada orang luar. Apabila server anda terhubung ke Internet dan memang server WWW anda disiapkan untuk publik, maka anda harus lebih berhati-hati sebab anda membuka pintu akses ke seluruh dunia!

- Server WWW menyediakan fasilitas agar client dari tempat lain dapat mengambil informasi dalam bentuk berkas (file), atau mengeksekusi perintah (menjalankan program) di server. Fasilitas pengambilan berkas dilakukan dengan perintah "GET", sementara mekanisme untuk mengeksekusi perintah di server dapat dilakukan dengan "CGI" (Common Gateway Interface), Server Side Include (SSI), Active Server Page (ASP), PHP, atau dengan menggunakan *servlet* (seperti pernggunaan *Java Servlet*). Kedua jenis servis di atas (mengambil berkas biasa maupun menjalankan program di server) memiliki potensi lubang keamanan yang berbeda.

Keamanan Server WWW

- Adanya lubang keamanan di sistem WWW dapat dieksploitasi dalam bentuk yang beragam, antara lain:
  - informasi yang ditampilkan di server diubah sehingga dapat mempermalukan perusahaan atau organisasi anda (dikenal dengan istilah *deface*1);
  - informasi yang semestinya dikonsumsi untuk kalangan terbatas (misalnya laporan keuangan, strategi perusahaan anda, atau database client anda) ternyata berhasil disadap oleh saingan anda (ini mungkin disebabkan salah setup server, salah setup router / firewall, atau salah setup authentication);
  - informasi dapat disadap (seperti misalnya pengiriman nomor kartu kredit untuk membeli melalui WWW, atau orang yang memonitor kemana saja anda melakukan *web surfing*);
  - server anda diserang (misalnya dengan memberikan *request* secara bertubi-tubi) sehingga tidak bisa memberikan layanan ketika dibutuhkan (*denial of service attack*);
  - untuk server web yang berada di belakang firewall, lubang keamanan di server web yang dieksploitasi dapat melemahkan atau bahkan menghilangkan fungsi dari firewall (dengan mekanisme *tunneling*).

Membatasi akses melalui Kontrol Akses

- Sebagai penyedia informasi (dalam bentuk berkas-berkas), sering diinginkan pembatasan akses. Misalnya, diinginkan agar hanya orang-orang tertentu yang dapat mengakses berkas (informasi) tertentu. Pada prinsipnya ini adalah masalah kontrol akses. Pembatasan akses dapat dilakukan dengan:
  - membatasi domain atau nomor IP yang dapat mengakses;
  - menggunakan pasangan userid & password;
  - mengenkripsi data sehingga hanya dapat dibuka (dekripsi) oleh orang yang memiliki kunci pembuka.
  - Mekanisme untuk kontrol akses ini bergantung kepada program yang digunakan sebagai server.

Proteksi halaman dengan menggunakan password

- Salah satu mekanisme mengatur akses adalah dengan menggunakan pasangan *userid* (*user identification*) dan *password*. Untuk server Web yang berbasis Apache, akses ke sebuah halaman (atau sekumpulan berkas yang terletak di sebuah directory di sistem Unix) dapat diatur dengan menggunakan berkas ".htaccess".

Secure Socket Layer

- Salah satu cara untuk meningkatkan keamanan server WWW adalah dengan menggunakan enkripsi pada komunikasi pada tingkat socket. Dengan menggunakan enkripsi, orang tidak bisa menyadap data-data (transaksi) yang dikirimkan dari/ke server WWW. Salah satu mekanisme yang cukup populer adalah dengan menggunakan *Secure Socket Layer* (SSL) yang mulanya dikembangkan oleh *Netscape*.

- Selain server WWW dari Netscape, beberapa server lain juga memiliki fasilitas SSL juga. Server WWW *Apache* (yang tersedia secara gratis) dapat dikonfigurasi agar memiliki fasilitas SSL dengan menambahkan software tambahan (SSLeay - yaitu implementasi SSL dari Eric Young - atau OpenSSL- yaitu implementasi Open Source dari SSL). Bahkan ada sebuah perusahaan (*Stronghold*) yang menjual Apache dengan SSL.

Mengetahui Jenis Server

- Informasi tentang web server yang digunakan dapat dimanfaatkan oleh perusak untuk melancarkan serangan sesuai dengan tipe server dan operating system yang digunakan. Seorang penyerang akan mencari tahu software dan versinya yang digunakan sebagai web server, kemudian mencari informasi di Internet tentang kelemahan web server tersebut.
- Informasi tentang program server yang digunakan sangat mudah diperoleh. Cara yang paling mudah adalah dengan menggunakan program “telnet” dengan melakukan telnet ke port 80 dari server web tersebut, kemudian menekan tombol return dua kali. Web server akan mengirimkan respon dengan didahuli oleh informasi tentang server yang digunakan. Program *Ogre* (yang berjalan di sistem Windows) dapat mengetahui program server web yang digunakan. Sementara itu, untuk sistem UNIX, program *lynx* dapat digunakan untuk melihat jenis server dengan menekan kunci “sama dengan” (=).

Keamanan Program CGI

- Common Gateway Interface (CGI) digunakan untuk menghubungkan sistem WWW dengan software lain di server web. Adanya CGI memungkinkan hubungan interaktif antara user dan server web. CGI seringkali digunakan sebagai mekanisme untuk mendapatkan informasi dari user melalui “fill out form”, mengakses database, atau menghasilkan halaman yang dinamis.

- Meskipun secara prinsip mekanisme CGI tidak memiliki lubang keamanan, program atau skrip yang dibuat sebagai CGI dapat memiliki lubang keamanan (baik secara sengaja dibuat lubang keamanannya ataupun tidak sengaja). Pasalnya, program CGI ini dijalankan di server web sehingga menggunakan resources web server tersebut.

Potensi lubang keamanan yang dapat terjadi dengan CGI

- Seorang pemakai yang nakal dapat memasang skrip CGI sehingga dapat mengirimkan berkas password kepada pengunjung yang mengeksekusi CGI tersebut.
- Program CGI dipanggil berkali-kali sehingga server menjadi terbebani karena harus menjalankan beberapa program CGI yang menghabiskan memori dan *CPU cycle* dari web server.
- Program CGI yang salah konfigurasi sehingga memiliki otoritas seperti sistem administrator sehingga ketika dijalankan dapat melakukan perintah apa saja. Untuk sistem UNIX, ada saja administrator yang salah seting sehingga server web (httpd) dijalankan oleh root.
- CGI guestbook yang secara otomatis menambahkan informasi ke dalam halaman web seringkali disalahgunakan oleh orang yang nakal dengan mengisikan link ke halaman pornografi atau diisi dengan sampah (junk text) sehingga memenuhi disk pemilik web.
- Teks (informasi) yang dikirimkan ke CGI diisi dengan karakter tertentu dengan tujuan untuk merusak sistem. Sebagai contoh, banyak search engine yang tidak melakukan proses “sanitasi” terhadap karakter yang dituliskan oleh user. Bagaimana jika user memasukkan “abcd; rm -rf /” atau “%; drop table” dan sejenisnya. (Tujuan utama adalah melakukan attack terhadap SQL server di server.)

Keamanan client WWW

- **Pelanggaran Privacy**

  - Ketika kita mengunjungi sebuah situs web, browser kita dapat "dititipi" sebuah "*cookie*" yang fungsinya adalah untuk menandai kita. Ketika kita berkunjung ke server itu kembali, maka server dapat mengetahui bahwa kita kembali dan server dapat memberikan setup sesuai dengan keinginan (*preference*) kita. Ini merupakan servis yang baik. Namun data-data yang sama juga dapat digunakan untuk melakukan *tracking* kemana saja kita pergi.

  - Ada juga situs web yang mengirimkan script (misal Javascript) yang melakukan interogasi terhadap server kita (melalui browser) dan mengirimkan informasi ini ke server. Bayangkan jika di dalam komputer kita terdapat data-data yang bersifat rahasia dan informasi ini dikirimkan ke server milik orang lain.

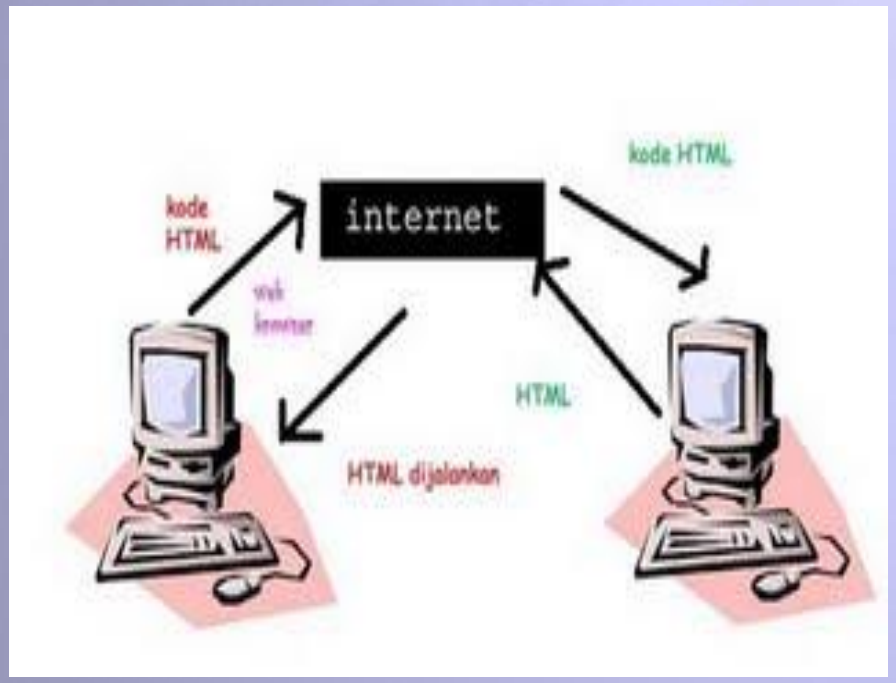
Keamanan client WWW

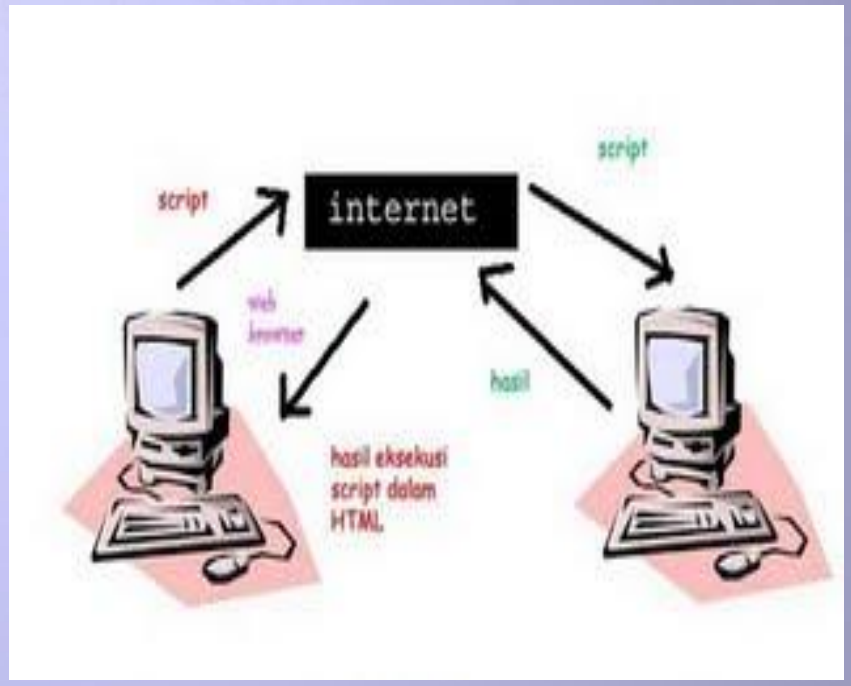
- **Penyisipan Trojan Horse**

  - Cara penyerangan terhadap client yang lain adalah dengan menyisipkan virus atau trojan horse. Bayangkan apabila yang anda download adalah virus atau trojan horse yang dapat menghapus isi harddisk anda. Salah satu contoh yang sudah terjadi adalah adanya web yang menyisipkan trojan horse Back Orifice (BO) atau Netbus sehingga komputer anda dapat dikendalikan dari jarak jauh. Orang dari jarak jauh dapat menyadap apa yang anda ketikkan, melihat isi direktori, melakukan reboot, bahkan memformat harddisk!

Aplikasi Web

- Pada awalnya aplikasi *Web* dibangun hanya dengan menggunakan bahasa yang disebut HTML (*HyperText Markup Language)*. Pada perkembangan berikutnya, sejumlah skrip dan objek dikembangkan untuk memperluas kemampuan HTML.
- Aplikasi *Web* itu dapat dibagi menjadi *Web* statis dan *Web* dinamis
- Dari sisi teknologi yang digunakan untuk membentuk *web* dinamis terdapat dua pengelompokan, yaitu teknologi pada sisi *client* dan teknologi pada sisi *server*.





Aplikasi Web-Client Side Programming

- Teknologi *Web* pada sisi *client* diimplementasikan dengan mengirimkan kode perluasan HTML atau program tersendiri dan HTML ke *client*.
- *Client*lah yang bertanggung jawab dalam melakukan proses terhadap seluruh kode yang diterima.
- Kelemahan pendekatan seperi ini adalah terdapat kemungkinan bahwa *browser* pada *client* tidak mendukung fitur kode perluasan HTML.
- Kelebihan teknologi pada sisi *client*, yaitu memungkinkan penampilan yang bersifat dinamis.
- Contoh teknologi pada sisi *client*, yaitu Kontrol ActiveX, Java Applet, dan Skrip sisi-*client*.

Aplikasi Web-Server Side Programming

- Teknologi *Web* pada sisi server memungkinkan pemrosesan kode di dalam *server* sehingga kode yang sampai pada pemakai berbeda dengan kode asli pada server. Contoh teknologi yang berjalan di *server*, yaitu CGI, ASP, JSP, PHP dan lain sebagainya. Keuntungan penggunaan teknologi pada sisi *server* adalah sebagai berikut:
  - Mengurangi lalu lintas jaringan dengan cara menghindari percakapan bolak-balik antara *client* dan server.
  - Mengurangi waktu pemuatan kode, mengingat *client* hanya mengambil kode HTML saja.
  - Mencegah masalah ketidakkompatibelan *browser*.
  - *Client* dapat berinteraksi dengan data yang ada pada *server*.
  - Mencegah *client* mengetahui rahasia kode (mengingat kode yang diberikan ke *client* berbeda dengan kode asli pada *server*) (Nugroho, 2004).

Bahan Bacaan

- Informasi lebih lanjut mengenai keamanan sistem WWW dapat diperoleh dari sumber on-line sebagai berikut.
  - <http://www.w3.org/Security/Faq/>
  - Nalneesh Gaur, “Assessing the Security of Your Web Applications,” Linux Journal, April 2000, hal. 74-78.
  - Netscape’s cookie Security FAQ
  - http://search.netscape.com/assist/security/faqs/cookies.html

Merci bien
ありがとう
Matur Nuwun
Hatur Nuhun
Obrigado
Dank
Thanks
Matur se Kelangkong
Syukron
Kheili Mamnun
ευχαριστιες
Danke
Grazias
谢谢
Terima Kasih


Irawan_afrianto@yahoo.com

Irawan.afrianto

@Irawan_afrianto

+628170223513