# Keamanan Sistem Informasi Berbasis Web

Subianto | AMIK JTC SEMARANG

# PENGERTIAN

- Dikembangan oleh Tim Berners-Lee
  ketika sedang berada di CERN (kompleks laboratorium percepatan partikel terbesar di dunia yang terletak di perbatasan antara Perancis dan Swis)
- Kemudahan untuk mengakses informasi melalui sistem hypertext
- Mula-mula dikembangkan dengan NeXT, kemudian muncul Mosaic (Windows, Mac, Unix), dan … akhirnya Netscape. Kemudian meledak

# Arsitektur

- Arsitektur
  - Server (apache, IIS)
  - Client (IE, Netscape, Mozilla, Opera, kfm, arena, amaya, lynx)
  - Terhubung melalui jaringan
- Program dapat dijalankan di server (CGI, [java] servlet) atau di sisi client (javascript, java applet)

# ASUMSI

# ASUMSI PENGGUNA

- Server dimiliki dan dikendalikan oleh organisasi yang mengaku memiliki server tersebut
- Dokumen yang ditampilkan bebas dari virus atau itikad jahat lainnya
- Server tidak mencatat atau mendistribusikan informasi tentang user (misalnya kebiasaan browsing)

# ASUMSI WEBMASTER

- Pengguna tidak mencoba merusak web server atau mengubah isinya
- Pengguna hanya mengakses dokumen yang diperkenankan
- Identitas pengguna benar

# ASUMSI PENGGUNA DAN WEBMASTER

- Network bebas dari penyadapan pihak ketiga
- Informasi yang disampaikan dari server ke pengguna terjamin keutuhannya dan tidak dimodifikasi oleh pihak ketiga

# Keamanan Server

- Server (httpd) menyediakan informasi (statis dan dinamis)
- Halaman statis diperoleh dengan perintah GET
- Halaman dinamis diperoleh dengan
  - CGI (Common Gateway Interface)
  - Server Side Include (SSI)
  - Active Server Page (ASP), PHP
  - Servlet (seperti Java Servlet, ASP)

# Web vulnerabilities

- Intercept informasi dari klien
  - Data, password, dll
- Pencurian data di server
  - Data, password, dll
- Menjalankan aplikasi di server
  - Memungkinkan melakukan eksekusi program "ngak benar" di server
- Denial Of Services
- Server Side Scripting, Cgi-Bin
  - Kesalahan pemograman membuka peluang

# Ekploitasi Celah Keamanan

- informasi yang ditampilkan di server diubah sehingga dapatmempermalukan perusahaan atau organisasi anda (dikenal dengan istilah **deface**);
- informasi yang semestinya dikonsumsi untuk kalangan terbatas (misalnya laporan keuangan, strategi perusahaan, atau database client) ternyata berhasil disadap oleh saingan (ini mungkindisebabkan 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 melakukan web surfing);
- server 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).

# Serangan umum pada server

- SMTP servers (port 25)
- RPC servers (port 111)
- NetBIOS shares (ports 135, 139, 445)
  - Blaster worm
  - Sasser worm
- FTP servers (ports 20, 21)
  - wuftpd vulnerabilities
- SSH servers (port 22)
  - OpenSSH, PAM vulnerabilities
- Web servers (ports 80, 443)
  - Apache chunked encoding vulnerability

# ASUMSI PENGGUNA DAN WEBMASTER

- Network bebas dari penyadapan pihak ketiga
- Informasi yang disampaikan dari server ke pengguna terjamin keutuhannya dan tidak dimodifikasi oleh pihak ketiga

# Kelemahan security pada aplikasi web

Berikut adalah kelemahan security teratas pada aplikasi web

1. Masukan (input) yang tidak tervalidasi
2. Broken Access Control
3. Pengelolaan Autentikasi dan Session yang tidak baik
4. Cross site scripting
5. Buffer overflows
6. Injections flaws
7. Denial of Service
8. Pengelolaan konfigurasi yang tidak aman

# 1. Masukan (input) yang tidak tervalidasi

- Aplikasi web menerima data dari HTTP request yang dimasukkan oleh user
- Hacker dapat memanipulasi request untuk menyerang keamanan situs

Hal – hal yang harus diperhatikan ketika mengelola validasi:

- Tidak cukup hanya bergantung pada script client side yang biasa digunakan untuk mencegah masukan form ketika ada input yang invalid
- Penggunaan kode validasi untuk memeriksa masukan tidak mencukupi

# 2. Broken Access Control

- Pada aplikasi yang membedakan akses dengan menggunakan perbedaan ID, hanya menggunakan satu halaman untuk memeriksa user.
- Jika user berhasil melewati halaman login, maka dia bebas melakukan apa saja
- Permasalahan lain adalah:
  - ID yang tidak aman - ID bisa ditebak
  - Ijin file - File yang berisi daftar user bisa dibaca orang lain

# 3. Pengelolaan Autentikasi dan Session yang tidak baik

Beberapa hal yang harus diperhatikan:

- Password strength
- Penggunaan password
- Penyimpanan password
- Session ID Protection

# 4. Cross Site Scripting

- XSS merupakan kependekan yang digunakan untuk istilah cross site scripting.
- XSS merupakan salah satu jenis serangan injeksi code (code injection attack).
- XSS dilakukan oleh penyerang dengan cara memasukkan kode HTML atau client script code lainnya ke suatu situs.

# Jenis XSS

- **Reflected XSS**

Reflected XSS merupakan tipe XSS yang paling umum dan yang paling mudah dilakukan oleh penyerang. Penyerang menggunakan *social engineering* agar tautan dengan kode berbahaya ini diklik oleh pengguna. Dengan cara ini penyerang bisa mendapatkan *cookie* pengguna yang bisa digunakan selanjutnya untuk membajak *session* pengguna.

Mekanisme pertahanan menghadapi serangan ini adalah dengan melakukan validasi input sebelum menampilkan data apapun yang di-*generate* oleh pengguna. Jangan percayai apapun data yang dikirim oleh pengguna.

**Stored XSS**

Stored XSS lebih jarang ditemui dan dampak serangannya lebih besar. Sebuah serangan stored XSS dapat berakibat pada seluruh pengguna. Stored XSS terjadi saat pengguna diizinkan untuk memasukkan data yang akan ditampilkan kembali. Contohnya adalah pada *message board*, buku tamu, dll. Penyerang memasukkan kode HTML atau *client script code* lainnya pada posting mereka.

Serangan ini lebih menakutkan. Mekanisme pertahanannya sama dengan *reflected XSS*: jika pengguna diizinkan untuk memasukkan data, lakukan validasi sebelum disimpan pada aplikasi..

# 5. Buffer Overflow

- Aplikasi dan Operating System (OS) menyimpan untuk sementara perintah yang mereka dapat di memori tertentu yang biasa disebut buffer memory. Kalau OS atau program tidak bisa dikode secara sempurna maka hacker bisa membuat komputer korban jadi terganggu dengan mengirimkan perintah yang dibuat khusus yang membuat gangguan jadi berlangsung lebih lama.
- **Buffer overruns pada kebanyakan Web server, DNS overflow, Serangan DNS, Mengelabui cache DNS**

# 6. Injection Flaws

- Penyerang mengirimkan “inject” calls ke OS atau resource lain, seperti database
- Salah satu yang terkenal adalah SQL Injection

# 7. Denial Of Service Attack (DoS Attack)

- Denial Of Service Attack (DoS Attack) adalah jenis serangan terhadap sebuah komputer atau server di dalam jaringan internet dengan cara menghabiskan sumber (resource) yang dimiliki oleh komputer tersebut sampai komputer tersebut tidak dapat menjalankan fungsinya dengan benar sehingga secara tidak langsung mencegah pengguna lain untuk memperoleh akses layanan dari komputer yang diserang tersebut.

- Cara mencegahnya yaitu dengan cara memproteksi sistem dengan paket filtering atau firewall.

# 8. Pengelolaan konfigurasi yang tidak aman

- Lubang keamanan yang tidak ditambal  (patched)
- Ijin file dan direktori yang tidak baik
- Account default dengan password default

# Web Security

- Authentikasi
  - FORM HTML
  - Basic, Digest
  - Klien Side + Server Side Scripting
- Manajemen Sesi
- Menggunakan Layer lain
  - S-HTTP ( discontinoued)
  - HTTPS ( HTTP ovel SSL)
  - IPSec
- Konfigurasi Web Server
  - Hak Akses
  - Indexes
  - Penempatan File

# Authentikasi

- FORM HTML

  <form action="modules.php?name=Your_Account" method="post">...

  <br><input type="hidden" name="op" value="login"> ...

  </form>

  - Tidak di enkripsi
- BASIC | Algortima Base64, Mudah di Dekrip
- DIGEST | Alghoritma Digest Ex: MD5 (Belum 100% di support)
- CS + SS Script

# Manajemen Sesi

- Hiden Form Field

  ***<input type="hidden" name="uniqueticket"***

  View page Source
- Cookies

  User harus mennghidupkan fasilitas

  Poisoned cookies
- Session Id
- URL Rewriting

  ***http://login.yahoo.com/config/login?.tries=&.src=ym&.last=&promo=&.intl=us***

# Manajemen Sesi

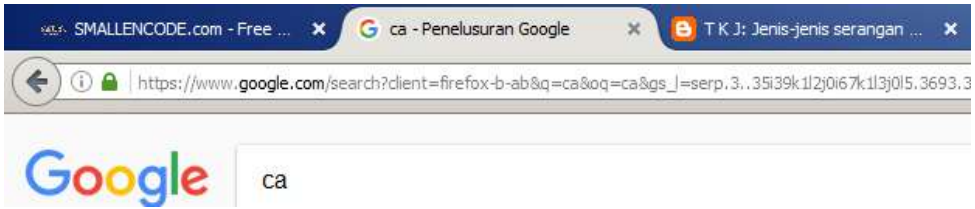
```php
<?php
ob_start();
session_start();

if(!isset($_SESSION["ses_user"]) and
!isset($_SESSION["ses_pass"]))
{
        header('Location:login.php');
}
?>
```

```php
<?php
  session_start();
  unset($_SESSION['ses_user']);
  unset($_SESSION['ses_pass']);
  header('Location:index.php');
?>
```

# SSL

- Untuk Semua Protokol TCP
  - Telnet -> SSH
  - HTTP -> HTTPS
- Public Key Server
- Hashing
  - MD5 + SHA
- CA Sekarang -> TLS




# Web Security

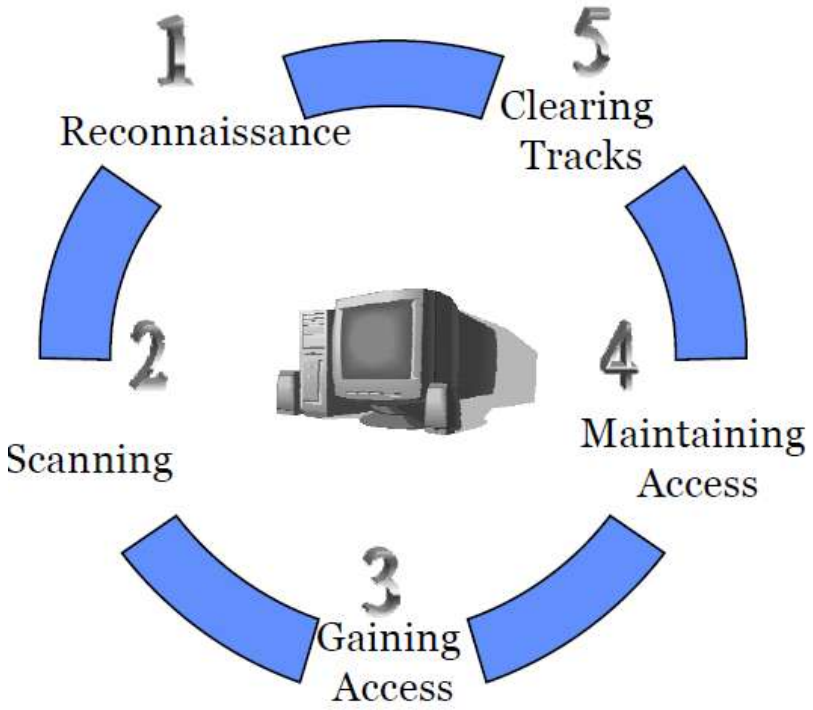
- Membatasi akses melalui Kontrol Akses
- Proteksi halaman dengan menggunakan password
- Secure Socket Layer
- Mengetahui Jenis Server
- Keamanan Program CGI
- Keamanan client WWW
- Pelanggaran Privacy
- Penyisipan Trojan Horse

# 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

- 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 home pae 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*

# Fase-Fase Serangan



# Reconnaissance

- Finding as much information about the target as possible before launching the first attack packet
- Reconnaissance techniques -> Low tech methods, General web searches, Whois databases, DNS

# Scanning

- Setelah fase pengintaian, penyerang dipersenjatai dengan beberapa informasi penting tentang infrastruktur sasaran: segenggam nomor telepon, nama domain, alamat IP, dan informasi kontak teknis
- Kebanyakan penyerang kemudian menggunakan pengetahuan ini untuk memindai sistem sasaran mencari bukaan.
- Fase scanning ini mirip dengan pencuri memutar kenop pintu dan mencoba untuk membuka jendela untuk menemukan jalan ke rumah korban.
- Vulnerability Scanning Tools, Network Mapping, Traceroute, Cheops, Port Scanning, Nmap

# Gaining Access

- Pada tahap ini, penyerang telah selesai memindai jaringan target, mengembangkan inventarisasi sistem target dan potensi kerentanan pada mesin tersebut.
- Selanjutnya, penyerang ingin mendapatkan akses pada sistem sasaran.
- Pendekatan khusus untuk mendapatkan akses sangat tergantung pada tingkat keterampilan penyerang, dengan script kiddies sederhana memancing untuk eksploitasi dan penyerang yang lebih canggih menggunakan pendekatan yang sangat pragmatis.
- Script Kiddie Exploit Trolling, Password Guessing Attacks, Password Guessing through Login Scripting, Password Cracking, Password Cracking Tools, Web Application Attacks, Web Application Attacks

# Maintaining Access

- Setelah mendapat akses ke sistem korban, hacker selanjutnya akan mencoba mempertahankan akses ini supaya dikemudian hari, atau bila diperlukan, mereka dapat masuk kembali ke sistem ini dengan mudah melalui backdoor atau pintu belakang yang telah dibuat oleh hacker tersebut.

# Maintaining Access

- Setelah mendapat akses ke sistem korban, hacker selanjutnya akan mencoba mempertahankan akses ini supaya dikemudian hari, atau bila diperlukan, mereka dapat masuk kembali ke sistem ini dengan mudah melalui backdoor atau pintu belakang yang telah dibuat oleh hacker tersebut.

# Clearing Tracks

- Tahap ini merupakan tahapan yang paling sulit untuk dilakukan dan merupakan Tahapan yang banyak dilupakan oleh para hacker.
- Umumnya mereka meninggalkan jejak di log file.