ISSN : 977.2087.526605

JURNAL INFORMATIKA

No. 3 Vol. 2, September – Desember 2011

Penerbit : Jurusan Teknik Informatika
Institut Teknologi Nasional
Penanggung Jawab : Ketua Jurusan Teknik
Informatika Institut
Teknologi Nasional
Pemimpin Redaksi : Dewi Rosmala
Wakil Pemimpin : Uung Ungkawa

Mitra Bestari : Arief Syaichu Rohman

Redaksi Pelaksana : 1. Asep Nana Hermana
2. Jasman Pardede

Sekretaris Redaksi : 1. Rio Korio Utoro
2. Yusuf Miftahudin
3. Rizky Faissa Akbar

DAFTAR ISI

No. 3 Vol. 2, September - Desember 2011

JURNAL INFORMATIKA diterbitkan 3 kali dalam satu tahun.
Berisi tulisan yang diangkat dari hasil penelitian
dan kajian analisis di bidang ilmu pengetahuan dan teknologi.

Alamat redaksi dan tata usaha :

Jurusan Teknik Informatika Institut Teknologi Nasional
Gedung 2 Lantai 2
Jl. PHH. Mustofa 23 Bandung 40124
Telp. 7272215 Fax. 7202892 e-mail : d_rosmala@itenas.ac.id

IMPLENTASI SECURE SOCKET LAYER (SSL) PADA SISTEM INFORMASI E-RESTAURAN BERBASIS WEB

Dewi Rosmala[1], Agung Budi Prasetyo[2]

Jurusan Teknik Informatika
Institut Teknologi Nasional Bandung

ABSTRAK

Pada sebuah bisnis restoran pendataan barang yang keluar dan masuk harus tercatat dan terorganisir agar tidak terjadi penumpukan barang atau bahkan kekurangan stok.Untuk merealisasikan hal tersebut perlu dibuat sebuah sistem yang mengatur lalulintas data dan informasi.Salah satu sistem yang dapat merealisasikannya adalah sistem Point of Sale (POS).Kebanyakan sistem POS yang ada di pasaran berbasis desktop. POS berbasis desktop memiliki kelemahan diantaranya jika perusahaan memiliki cabang yang banyak memerlukan dana yang besar utuk merealisasikannya, juga sulit melakukan singkronisasi data dengan pusat, untuk perbaikan juga membutuhkan waktu yang lama karena harus datang ke tiap cabang untuk memperbaikinya. Untuk itu penulis mengembangkan sistem informasi e-restaurant berbasis web dimana seluruh sumberdaya tersimpan di sebuah server yang berada di internet. E-restaurant memunculkan masalah pengamanan data sensitif dan pribadi di "hutan rimba" internet. Untuk itu e-restaurant harus memenuhi aspek keamanan, yaitu privacy, integrity, authentication, dan availability. Untuk memenuhi aspek tersebut dapat dilakukan enkripsi pada komunikasi tingkat soket dengan memanfaatkan Secure Socket Layer (SSL) sebagai protokol keamanan.

Kata Kunci :*Point Of Sale, e-restaurant, SSL*

ABSTRACT

Intherestaurant business, inventory itemshould berecordedand organized.To realize this neededa systemthat regulatesthe trafficof dataand information. One of thesystemthat canrealize is a Pointof Sale(POS) System. MostPOSsystemson the marketis desktop based. Desktop based POShave drawbackssuch asif the company hasmanybranches, it is a great avenueto make it happen, and thandifficult tosynchronizethe datawith the center, forrepairsalsotake a long timebecause programer mustcome toeach branchto fix it. Because of that authordeveloped theweb-based e-restaurant information systemsthatapply the concept ofe-business in managingthe business processesof arestaurant. E-restaurant raises the issue ofsecuringsensitive andprivacy datain the internet "jungle". Because of thate-restaurant mustfulfill the aspects of security, that isprivacy, integrity, authentication, and availability. To fulfilltheseaspectscan beperformedencryption onthe socket-level communication by usingSecureSocketLayer(SSL) forsecurityprotocols.

Key Word :*Point Of Sale, e-restaurant, SSL*

PENDAHULUAN

Pada sebuah bisnis restoran, pendataan barang sangat penting. Barang yang keluar dan masuk harus tercatat dan terorganisir agar tidak terjadi penumpukan barang atau bahakan kekurangan stok.Untuk memenuhi kebutuhan tersebut maka harus dibuat sebuah sistem yang mengatur lalulintas informasi dan data. Salah satu sistem yang dapat merealisasikannya adalah sistem informasi terkomputasi yang disebut sebagai sistem *Point Of Sale* (POS) [1].

Biasanya POS yang ada di pasaran adalah berbasis desktop.POS berbasis desktop memiliki kelemahan diantaranya sulit melakukan singkronisasi data dengan pusat dan membutuhkan dana yang besar untuk membangun sarana komunikasi tersebut juga proses perbaikan yang merepotkan.

Berdasarkan masalah-masalah di atas, penulis pengembangan sistemPOS dimana seluruh sumberdaya tersimpan di sebuah server yang berada di internet.Setiap *user* yang memiliki hak akses dapat mengakses aplikasi ini asalkan komputer mereka terhubung dengan internet. Aplikasi yang dikembangkan adalah sebuah sistem informasi *e-restaurant* berbasis web yang menerapkan konsep *e-business* dalam mengelola proses bisnis sebuah restoran.

E-restaurant memunculkan masalah pengamanan data sensitif dan pribadi di "hutan rimba" internet.Isu tersebut memunculkan masalah kepercayaan pelaku bisnis (*end user)* terhadap *e-restaurant*.Untuk itu *e-restaurant* harus memenuhi empat aspek keamanan, yaitu *privacy*, *integrity*, *authentication*, dan *availability*[3].

Untuk memenuhi aspek tersebut dapat dilakukan enkripsi pada komunikasi tingkat soket dengan memanfaatkan *secure socket layer* (SSL). SSL adalah protokol keamanan yang didesain untuk dijalankan pada TCP/IP.keamanan dijamin dengan menggunakan kombinasi kriptografi kunci publik dan kriptografi kunci simetri bersamaan dengan sebuah infrastruktur sertifikat, dalam penelitian ini algoritma kriptografi yang digunakan adalah RC4. Sertifikat adalah sebuah kumpulan data identifikasi dan kunci publik dalam format yang telah distandarisasi. Data tersebut digunakan dalam proses verifikasi identitas pada internet. Dengan kunci publik pada sertifikat SSL memastikan data transaksi dienkripsi sehingga tidak dapat dibaca oleh pihak lain.

Secara umum cara kerja protokol SSL adalah sebagai berikut :

1. *User* membuka suatu halaman yang mendukung protokol SSL, biasanya diawali dengan https:// pada browser.
2. Kemudian webserver mengirimkan kunci publiknya beserta sertifikat server.
3. Browser melakukan pemeriksaan apakah sertifikat tersebut terpercaya, apakah sertivikat tersebut masih valid dan memang berhubungan dengan situs yang sedang dikunjungi.
4. Setelah yakin browser menggunakan kuncipublik dari webserver untuk melakukan enkripsi terhadap suatu kunci simetri yang dibuatsecara acak dari puhak *user*. Kunci yang terenkripsi kemudian dikirim ke server untuk mengenkripsi data yang diperlukan.
5. Server mengirim kembali data yang diminta *user* dan telah terenripsi dengan kunci simetri tadi.

DASAR TEORI

Secure Socket Layer (SSL)[4]

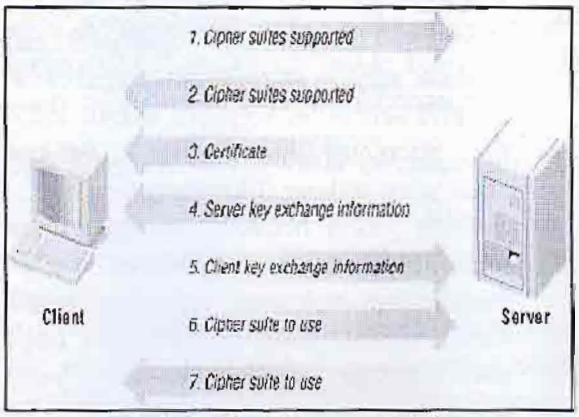
Di internet, enkripsi data selalu menggunakan protokol *Secure Socket Layer* (SSL). Protokol ini di desain oleh Netscape untuk digunakan pada *browser-*

nya. SSL didesain untuk digunakan pada soket.

Untuk mengimplementasikan SSL, harus menggunakan soket TCP untuk mengirim paket data. Soket UDP dan transmisi lain tidak dapat di dukung. Meskipun demikian ada tiga keuntungan menggunakan SSL, diantaranya :

1. Ubiquitous. Banyak layanan yang dibangun di atas SSL(seperti HTTPS), untuk berkomunikasi *user* harus menggunakan SSL.
2. Memungkinkan pengembang fokus terhadap logic dari aplikasinya karena SSL diimplementasikan bukan pada layer aplikasi.
3. Tidak mempersulit klien untuk menunjukan sertifikat digitalnya, cukup dengan sertifikat digital pada server.

Protokol SSL melakukan beberapa pertukaran informasi antara klien dan server ketika klien pertama kali terhubung dengan server. Proses ini disebut SSL *handshake,* berikut adalah ilustrasi dari SSL *handshake :*



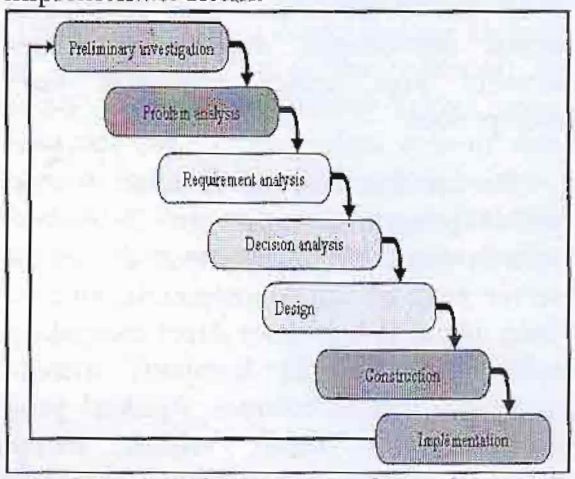
Gambar 1 SSL handsake

Meskipun ada variasi dari proses tersebut, namun secara umum aliran informasi seperti berikut :

1. Klien memulai koneksi ke server dan memberi tahu *chipher suites* yang didukung klien.
2. Server merespon dengan *chiper suites* yang didukung.
3. Server mengirim klien sertifikat yang terenkripsi untuk memverifikasi identitas server.
4. Server memberikan kunci untuk mendekripsi sertifikat.
5. Klien menyelesaikan pertukaran kunci dan bertukar informasi informasi ke server.
6. Berdasarkan kunci dan algoritma yang digunakan, klien memutuskan *chiper suites* yang digunakan.
7. Server memberikan keputusan finas mengenai *chiper suites* yang digunakan.

Metodologi Fast[2]

Metode ini membantu pengembangan sistem yang menyediakan mekanisme untuk memahami dan menganalisis kebutuhan pengguna, melakukan negosiasi, pemilihan solusi yang layak, pembuatan sistem yang terorganisi hingga implementasi sitem.



Gambar 2 tahapan metodologi FAST

Berdasarkan gambar 2 metodologi FAST meliputi beberapa tahap, diantaranya :

1. *Preliminary Investigation/Scope definition*
   Tahap ini merupakan tahap pertama dalam melakukan pengembangan suatu sistem. Dalam tahap ini dilakukan penentuan batasan-batasan sistem dan siapa saja yang akan memakai sitem.
2. *Problem Analysis*
   Tahap ini dilakukan analisis secara menyeluruh terhadap permasalahan dari sistem, penyebab permasalahan tersebut, serta menentukan apakah

permasalahan tersebut dapat diselesaikan.

3. *Requirement Analysis*
   Tahap ini dilakukan analisa terhadap *business requirement* sesuai dengan yang dibutuhkan dan diinginkan *user* yang menggunakan sistem tersebut.
4. *Design*
   Pada tahap ini akan digambarkan dalam bentuk gambar gambae baik itu logikal desain maupun fisikan desain.
5. *Costruction*
   Pada tahap ini dilakukan pembangunan sistem menggunakan bahasa pemrograman tertentu.
6. *Implementation*
   Pada tahap ini adalah tahap implementasi dari tahap desain yang telah dilakukan.

ANALISA DAN PERANCANGAN

Scope Definition (Definisi Lingkup)

E-restaurant mengelola proses bisnis dari restoran, berikut adalah proses bisnis yang dikelola oleh e-restaurant :

1. Pengelolaan Cabang
   Dalam pengelolaan cabang merupakan awal dari proses bisnis dari sebuah restoran. Diasumsikan dalam satu domain aplikasi ini digunakan oleh restoran A. Pengelolaan cabang menyimpan seluruh data cabang dari restoran A termasuk pusat.
2. Pengelolaan *Staff*
   Dalam pengelolaan staff tersimpan data dari seluruh staff yang ada pada restoran A baik itu di pusat ataupaun di cabang.
3. Pengelolaan *Supplier*
   Dalam pengelolaan supplier tersimpan data dari seluruh *supplier* yang berhubungan dengan restoran A.
4. Penglolaan *Order*
   Dalam pengelolaan order tersimpan data *order*. Setiap cabang mengakses data order masing masing. *Order* tiap cabang dibedakan dengan kode tertentu.
5. Pengelolaan *Purchasing Order*
   Dalam pengelolaan *purchashinh order* tersimpan data *order* yang telah dibeli. Setiap cabang mengakses data *purchasing order* masing masing. *Purchasing order* tiap cabang dibedakan dengan kode tertentu.
6. Pengelolaan Shipping
   Dalam pengelolaan *shipping* tersimpan data *purchashinh order* yang telah diterima yang kemudian akan menambahkan kuantiti dari *inventory*. cabang mengakses data *shipping* masing masing. *Shipping* tiap cabang dibedakan dengan kode tertentu.
7. Pengelolaan *Inventory*
   Dalam pengelolaan *inventory* tersimpan data bahan mentah yang digunakan oleh restoran A. setiap cabang memiliki data *inventory* yang sama namun setiap cabang memiliki kuantiti yang berbeda.
8. Pengelolaan Menu
   Dalam pengelolaan menu tersimpan data menu beserta resep. Resep berguna untuk mengurangi stok dari *inventory*. Setiap cabang mengakses menu dan resep yang sama.
9. Pengelolaan Penjualan
   Dalam pengelolaan penjualan tersimpan data transaksi penjualan. Setiap cabang memiliki mengakses data transaksi penjualan masing masing. Data transaksi setiap cabang dibedakan berdasarkan kode tertentu.

Problem Analysis (Analisis Masalah)

Dari definisi lingkup didapat aplikasi e-restaurant mengelola berbagai data pribadi dan sensitif yang tidak semua orang boleh mengakses atau mengubah. E-restaurant menyimpan semua data tersebut pada pada sebuah server di *cloud*. *Cloud* biasanya tersedia sebagai layanan kepada siapa saja di internet, untuk itu e-restaurant harus membatasi siapa saja yang boleh mengakses. Berikut adalah aspek keamanan yang harus miliki e-restaurant :

1. *Privacy*
   Data pribadi seperti *username* dan password atau data sensitif seperti harga barang harus dijaga dari orang yang tidak berhak mengakses. Serangan terhadap aspek privacy misalnya adalah usaha untuk melakukan penyadapan (*sniffing*).
   Usaha yang dapat dilakukan untuk menignkatkan keamanan pada aspek *privacy* adalah dengan menggunakan teknologi kriptografi.teknik riptografi didapat dengan mengimplementasikan SSL. Protokol SSL mempunyai suatu proses enkripsi yang digunakan untuk menyamarkan data asli yang dikirimkan klien ke server.
2. *Data Integrity*
   Data penting seperti *supplier*, *inventory*, dan transaksi merupakan salah satu data penting yang tidak boleh di ubah tanpa persetujuan dari pemilik informasi tersebut (admin atau manager). Contoh serangan pada aspek *data integrity* adalah "*man in the middle attack*" dimana seseorang menempatkan diri di tengah transmisi client dengan server. Penyerang menerima paket data yang dikirim client kepada server, lalu penyerang mengubah data tersebut sebelum meneruskan kepada server.
   Dengan protokol SSL, proses pengubahan data akan menjadi tidak mungkin bisa dilakukan, karena adanya paket data yang dikirim sudah dienkripsi, sehingga tidak bisa di baca oleh penyerang.
3. *Autentication*
   Aspek *Autentication* berhubungan dengan metoda untuk menyatakan bahwa informasi betul betul asli, orang yang mengakses betul betul orang yang dimaksud, atau server yang dituju betul betul server yang asli. Harus ada parameter yang dapat membuktikan keaslian dari client yang mengakses atau server yang dituju.
4. *Availability*
   Aspek *Availability* berhubungan dengan ketersediaan informasi ketika dibutuhkan.

Design (Perancangan)

Pada tahap ini akan dibahas mengenai gambaran umum sistem yang mencakup diagram aliran data, serta pemodelan keamanan.

1. Blok Diagram
   Untuk menuhi aspek keamanan yang telah dijabarkan pada tahap analisis masalah penulis memanfaatkan protokol keamanan SSL. Blok diagrammenggambarkan proses pengamanan menggunakan SSL pada sistem informasi e-restaurant berbasis web.

Gambar 3Blok Diagram

Proses berawal dari permintaan koneksi klient terhadap server. Kemudian server memberikan sertifikat SSL. Selanjutnya klien memeriksa sertifikat, jika terpercaya maka koneksi dilanjutkan dengan memberikan kunci enkripsi untuk sesi tersebut. Server menghubungkan klien dengan form login. Paket data yang dikirim sudah terenkripsi dengan kunci yang ditentukan sebelumnya, kemudian didekripsi begitu sampai di server. Server melakukan pengecekan *user*name dan password, jika terdaftar maka server akan melanjutkan memberikan sumber daya yang diminta klien dan mengenkripsi paket data tersebut dengan kunci yang ditentukan sebelumnya.

2. Perancangan Sistem
   Untuk menggambarkan aliran data dan proses bisnis pada sistem informasi e-restaurant berbasis web penulis menggunakan DFD.

   DFD menggambarkan proses bisnis dari restoran mulai dari pengelolaan staff hingga proses penjualan. Pada pengelolaan preorder, dikelola item apa saja yang akan di order. Pada pengelolaan purchase order item yang telah melewati proses preorder akan di pesan ke suppliernya masing masing. Pada pengelolaan shipping akan dicek berapa item yang di pesan, berapa item yang di terima, dan berapa item yang belum di terima setelah itu baru masuk ke inventory. Menu di buat dengan meracik resep dari item yang ada di inventory. Setelah menu siap dapat dilakukan transaksi penjualan.

3. Perancangan Basis Data
   Berdasarkan referensi dari DFD yang telah dirancang sebelumnya, tahap selanjutnya adalah perancangan database meggunakan ERD dan TRD.

   Pada ERD dan TRD terdiri dari 18 tabel yang disesuaikan dengan DFD yang telah dibuat sebelumnya ditambah *user*table dan grouptable untuk keperluan *login*, juga satuan dan konversi untuk keperluan satuan.Total keseluruhan tabel adalah 22 tabel.

4. Akses Kontrol
   Pada hak akses dijelaskan aktor yang terlibat langsung dengan aplikasi e-restaurant. Berikut adalah tabel mengenai hak akses pada aplikasi e-restaurant.

Tabel 1 Hak akses

Form/Aktor	Admin	Manager	Kasir
Form Branch	√		
Form Staff	√		
Form Supplier	√	√	
Form Inventory	√	√	
Form Shippping	√	√	
Form Order	√	√	√
Form Purchase Order	√	√	√
Form Sales	√	√	√

Seperti informasi yang tertera pada Tabel 1. Aktor yang terlibat langsung pada aplikasi e-restaurant diklasifikasikan sebagai berikut :

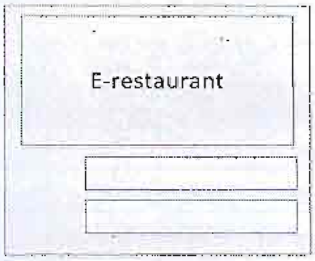
a. Admin
   Admin bertindak sebagai pemilik perusahaan yang menggunakan aplikasi *e-restaurant*. Admin dapat mengakses seluruh form yang ada pada aplikasi *e-restaurant*. Tugas dari admin adalah melakukan pengolahan data master.
b. Manager
   Manager memiliki tugas yang sama dengan admin, hanya saja manager tidak dapat melakukan pengelolaan cabang dan staff.
c. Kasir
   Kasir bertugas melakukan semua proses transaksi mulai dari transaksi *order* (Pemesanan),

Purchasing Order (Pembelian), hingga *Sales* (Penjualan).

5. Perancangan *User Interface*

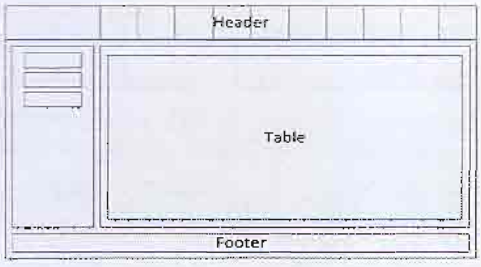
Secara keseluruhan aplikasi e-restaurant terdiri dari 12 form. Dari seluruh form dikelompokan berdasarkan kesamaan desain, yaitu form utama, form detail, dan form login. Berikut adalah desain layout dari aplikasi e-restaurant :

a. Form Login



Gambar 4 Layout Form Login

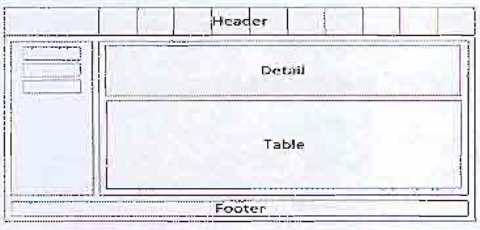
b. Form Utama



Gambar 5 Layout Form Utama

Desain form utama pada bagian header terdapat tombol untuk mengakses form-form yang tersedia. Di bagian kiri terdapat tombol untuk melakukan perintah dasar seperti insert, update dan delete. Di bagian tengah terdapat tabel yang berisikan data dari form tersebut.

c. Form Detail



Gambar 6 Layout Form Detail

Tidak berbeda jauh dengan form utama, form detail juga memiliki desain header dan footer yang sama. Yang membedakan hanyalah di bagian tengah form, tepatnya di atas tabel terdapat bagian yang berisikan informasi detail dari form tersebut.

IMPLEMENTASI DAN PENGUJIAN

Implementasi

Setelah melakukan analisa dan perancangan pada bab 3 tahapan selanjutnya adalah implementasi, berikut adalah tahapan implementasi sistem informasi e-restaurant berbasis web.

Implementasi Perangkat Keras

Perangkat keras yang digunakan untuk mengimplementasikan sistem adalah sebagai berikut :

1. Prosessor Genuine Intel Core i5 @ 2.4 GHz, 3MB L3 chace
2. Memory 4 GB RAM
3. VGA ATI Mobility Radeon HD 5470 dengan memory 512 MB
4. Kapasistas Hardisk 640 GB

Implementasi Perangkat lunak

Perangkat lunak yang digunakan untuk mengimplementasikan sistem adalah sebagai berikut :

1. Java Development Kit (JDK) versi 1.6.0.33
2. Netbeans versi 7.1
3. Glassfish versi 3.1.2

Implementasi Secure Socket Layer (SSL)

Untuk mengamankan komunikasi antara server dengan klien digunakan protokol keamanan SSL. Salahsatu bagian yang perlu kita konfigurasi untuk membangun komunikasi SSL pada server adalah sebuah sertifikat digital.

Sertifikat digital berisi kunci publik dan data data pemilik kunci publik. Sertifikat digital diberikan oleh Certification Authorities (CA). Dalam aplikasi e-restaurant sertifikat digital dibuat dengan kelas yang sudah di sediakan java. Berikut adalah tahapan pengimplementasian SSL pada aplikasi e-restaurant.

1. Membuat *Certificate Private Key*

   Tahapan pembuatan key yang dapat dilakukan dengan memanfaatkan class keytool pada java dan private key yang ada pada glassfish. Selanjutnya panggil command dan tulisakan perintah berikut :

```
Keytool -genkey -alias eresto-alias -keyalg RSA -keypass changeit -storepass changeit -keustore keystore.jks
```

   Setelah ditekan enter tahapan berikutnya adalah mengisi data yang ditanyakan untuk melengkapi data pribadi pemilik sertifikat.

2. Membuat *Certificate*

   Setelah selesai membuat key yang akan digunakan oleh sertifikat, tahapan selajutnya adalah pembuatan sertifikat itusendiri. Melalui command tuliskan kode berikut:

```
Keytool -export -alias eresto-alias -storepass changeit -file eresto.cer -keystore.jks
```

   Setelah ditekan enter maka sertifikat yang dibuat tersimpan dengan nama eresto.cer.

3. Mengatur *Certificate*

   Agar aplikasi server dapat mengenali sertifikat yang sudah dibuat, perlu ditambahkan pada daftar dari *trusted certificate* pada file bernama cacerts.jks. melalui command tuliskan kode berikut :

```
Keytool -import -v -trustcacerts -alias eresto-alias -file eresto.cer -keystore cacerts.jks -keypass changeit
```

   Setelah ditekan enter maka akan muncul data pemilik seperti :
   CN (*Certificate Name*) : Agung Budi
   OU(*Organizational Unit*) : Informatika
   O (*Organizational*) : Itenas
   L (*Locality or City*) : Bandung
   S (*State atau Province*) : West Java
   C (*Country Code*) : ID

4. Membuat *Secure HTTP listener*

   Setelah berhasil membuat sertifikat selanjutnya adalah membuat HTTP listener yang digunakan membuat komunikasi aman. Untuk melakukannya langkah pertama adalah login ke *administration console*.selanjutnya klik pada HTTP listener.

   Secara default glassfish sudah memiliki HTTP listener untuk komunikasi aman, terlihat pada file HTTP-listener2. Buka HTTP-listener2 lalu klik tab SSL. Pastikan mencek SSL3 dan TLS dan isikan certificate NickName dengan alias dari key yang dibuat sebelumnya. Pada bagian Cipher Suites, pilih SSL_RSA_WITH_RC4_128_SHA

5. Konfigurasi Security Pada Project

   Setelah http listener disiapkan untuk komunikasi aman menggunakan SSL. Tahapan selanjutnya adalah konfigurasi security pada project netbeans. Pertama buka web.xml dan pilih tab security. Pada security constraints ceklis checkbox enable *user* data constraint dan pilih confidential pada combobox transport guarantee. Konfigurasi tersebut yang menyebabkan aplikasi berjalan pada port 8181 ketika aplikasi diakses.

Pengujian

Pengujian dilakukan dengan menggunakan simulasi serangan. Berikut adalah simulasi penyerangan terhadap aplikasi e-restaurant.

Simulasi Serangan Man-in-the-middle

Pada pengujian keamanan sistem informasi e-restaurant berbasis web, akan dilakukan sebuah simulasi serangan *Man-in-the-middle*.Simulai dilakukan dengan aplikasi *Rawcap* yang mampu menangkap paket data yang dikirimkan dengan protokol HTTP. Berikut adalah tabel pengujian dari simulasi penyerangan Man-In-The-Middle :

Tabel 2 tabel Pengujian Simulasi Serangan Man-In-The-Middle

Nama Kasus Uji	Simulasi Serangan Man-In-The-Middle
Perangkat	Rawcap
Kondisi Awal	Halaman E-restaurant belum terbuka
Skenario	1. Hubungkan 2 buah komputer dalam jaringan LAN 2. Jalankan webserver pada komputer 1 3. Komputer 2 membuka aplikasi e-restaurant melalui browser dengan mengakses ip dari komputer yang menjalankan webserver 4. Komputer 2 menjalankan aplikasi rawcap 5. Komputer 2 melakukan login
Kondisi Yang Diharapkan	Komputer 2 tidak dapat membaca paket data yang dikirimkan ke komputer 1
Pengamatan	Komputer 2 dengan aplikasi rawcap tidak dapat membaca paket data yang telah dikirimkan.
Kesimpulan	Berhasil

Simulasi Serangan Denial of Service

Simulasi serangan DOS untuk menguji aspek *availability* . Simulasi DOS dilakukan dengan memanfaatkan aplikasi *PingFlood*. *PingFlood* adalah aplikasi yang dapat melakkukan ping dalam paket dengan jumlah yang sangat banyak serta sangat cepat pengirimannya.

Aplikasi pingFlood menyerang server, dalam kasus ini adalah server local dengan ip 127.0.0.1. Serangan tersebut mengakibatkan server tidak dapat diakses karena sibuk melayani ping dari pingFlood. Berikut adalah tabel pengujian dari simulai serangan DOS:

Tabel 3 tabel Pengujian Simulasi Serangan Denial Of Service

Nama Kasus Uji	Simulasi Serangan Denial of Service
Perangkat	PingFlood
Kondisi Awal	Halaman E-restaurant belum terbuka
Skenario	1. Aplikasi e-restaurant dijalankan pada localhost 2. Aplikasi pingFlood dijalankan dengan localhost sebagai tujuan serangan 3. Selagi aplikasi pingFlood dijalankan, lakukan komunikasi seperti biasa dengan e-restaurant
Kondisi Yang Diharapkan	E-restaurant berjalan seperti biasa
Pengamatan	E-restaurant tidak dapat di akses dan muncul pesan error "HTTP Status 403 – Access to the requested resource has been denied"
Kesimpulan	Gagal

KESIMPULAN

1. Berdasarkan hasil pengujian pada Tabel 2dengan implementasikan SSL sistem informasi e-restaurant berbasis web telah berhasil memenuhi aspek *privacy, integrity, authetication, access control,* dan *non-repudiation*dengan mengimplementasikan SSL.

E-restaurant gagal memenuhi aspek availability ketika server diserang oleh *Denial of Service (Dos)*. Server tidak dapat melayani klien karena sibuk melayani serangan Dos seperti pada hasil pengujian pada Tabel 3

DAFTAR PUSTAKA

1. Kuswono, Dodi.2010. *Solusi Bisnis Berbasis AJAX:Studi Kasus Sistem POS(Point Of Sale)*.Surabaya.Institut Teknologi Sepuluh Nopember
2. Pressman, Roger S.2005. *Software Engineering : a Practitioner's Approach*. Sixth edition. McGraw-Hill, New York.
3. Raharjo, Budi.2002.*Keamanan Sistem Informasi Berbasis Iternet*.Bandung
4. Simson, Garfinkel, 2005."*PGP: Pretty Good Privacy*," O'Reilly & Associates,Inc.,.