

Perancangan Cryptoboard dengan Mengimplementasikan OpenSSL-0.9.7 pada Single Board Computer

Indra Antonius Simalango

Daftar Isi

Pengantar	5
Pendahuluan	6
Latar Belakang Masalah	6
Rumusan Masalah.....	6
Tujuan	7
Ruang Lingkup Penelitian.....	7
Embeddable - Single Board Computer (Embeddable - SBC) dan SystemOnChip (SOC)	8
Apa itu SBC	8
Pangsa pasar “Embedded-SBC“	8
Lahirnya PC-compatible SBC.....	9
Aspek Perubahan	9
Sebuah teori yang sedikit mengacaukan	10
Beberapa hal tentang Linux pada SBC embedded-PC	11
SBC mini untuk proyek berbasis embedded-Linux	11
Single Board Computer tipe SBC 456/E	13
DiskOnChip 2000	15
OpenSSL	17
Mekanisme Kerja SSL	19
OpenSSL-0.9.7.....	20
Struktur Umum Kode OpenSSL	21
Struktur Direktori crypto/aes	23
Sistem Operasi eCos-2.0.i386.linux	24
Tentang eCos	24
Spesifikasi Sistem	25
eCos-2.0.i386.linux.....	25
Desain Sistem	27
Rangkuman	29
Catatan Penulis	30
Bibliografi	31

Daftar Gambar

Gambar 1 Single Board Computer seri 456/E	13
Gambar 2 Fisik dan Teknologi Diagram DiskOnChip 2000	15
Gambar 3 Homepage OpenSSL.....	17
Gambar 4 SSL merupakan lapisan terpisah dalam susunan protokol Internet	18
Gambar 5 SSL juga dapat menangani keamanan aplikasi lain	18
Gambar 6 Letak SSL dalam model ISO Reference	20
Gambar 7 Ide perancangan sistem	27
Gambar 8 Rencana Tahapan Pengerjaan	28

Daftar Tabel

Tabel 1 Spesifikasi Lengkap SBC 456/E.....	14
Tabel 2 Spesifikasi Lengkap DiskOnChip 2000.....	16
Tabel 3 Subdirektori yang berada langsung di bawah direktori utama OpenSSL .	22
Tabel 4 File-file yang berada langsung dibawah direktori utama	22

Pengantar

Makalah ini dibuat dalam rangka memenuhi tugas akhir mata kuliah EC-5010
Keamanan Sistem Informasi

Indra Antonius Simalango
13200083
Departemen Teknik Elektro
Program Studi Teknik Elektronika
Institut Teknologi Bandung
indra at ic.vlsi.itb.ac.id

Pendahuluan

Latar Belakang Masalah

Aspek keamanan dalam proses pertukaran data adalah salah satu pendorong munculnya teknologi Kriptografi. Kriptografi berbasis pada algoritma pengkodean data informasi yang mendukung kebutuhan dari dua aspek kewanitaan informasi, yaitu secrecy (perlindungan terhadap kerahasiaan data informasi) dan authenticity (perlindungan terhadap pemalsuan dan perubahan informasi yang tidak diinginkan).

Pada proses transfer data antar Personal Computer (PC) dalam jaringan, modul Kriptografi akan mengkodekan data informasi sedemikian rupa dengan memanfaatkan resource yang ada. Pemakaian resource secara bersama-sama dengan aplikasi-aplikasi lainnya dalam PC menjadikan proses pengkodean dalam modul Kriptografi dilakukan dalam jangka waktu relatif lama.

Trend teknologi juga mengisyaratkan keamanan mutlak pada source-code modul Kriptografi. Hal ini untuk mencegah terjadinya pembajakan dan penyalahgunaan. Selama modul Kriptografi masih berbentuk perangkat lunak (software), peluang akan terjadi hal ini tetap terbuka.

Ide yang muncul adalah untuk mengintegrasikan modul Kriptografi pada suatu perangkat keras (hardware) tersendiri. Perangkat ini haruslah mempunyai resource sendiri yang dikhususkan untuk mendukung kinerja modul Kriptografi sepenuhnya. Karena akan diimplementasikan pada PC, perangkat ini hendaknya berupa board dengan slot standar pada PC.

Salah satu alternatif yang memungkinkan untuk mewujudkan hal ini adalah dengan menginstal modul Kriptografi pada sebuah Single Board Computer (SBC). Spesifikasi SBC secara teknis memungkinkan implementasi hal ini.

Rumusan Masalah

Permasalahan utama yang akan dibahas dalam tulisan ini adalah bagaimana memfungsikan Single Board Computer (SBC) 456/E menjadi Cryptoboard dan dapat dikendalikan / digunakan oleh Personal Computer (PC) sebagai peripheral tambahan.

Tujuan

Penelitian ini dilakukan untuk mengintegrasikan Sistem Operasi eCos-2.0.i386linux dan modul kriptografi OpenSSL-0.9.7d yang sudah dioptimasi, untuk diimplementasikan pada Single Board Computer sebagai Cryptoboard, yang akan menjadi peripheral tambahan pada Personal Computer (PC).

Ruang Lingkup Penelitian

Penelitian ini dibagi menjadi dua bagian besar, sesuai dengan tujuan penelitian tersebut yaitu integrasi Sistem Operasi eCos-2.0.i386linux dan modul kriptografi OpenSSL-0.9.7d, untuk diimplementasikan pada Single Board Computer kemudian memfungsikan Single Board Computer sebagai Cryptoboard, yang akan menjadi peripheral tambahan pada Personal Computer (PC).

Ruang lingkup penelitian yang akan dibahas oleh penulis adalah :

1. Menggunakan Single Board Computer tipe SBC 456/E dan DiskOnChip2000 untuk dijadikan Cryptoboard
2. Instalasi Sistem Operasi eCos-2.0.i386linux pada SBC 456/E
 1. Menggunakan OpenSSL-0.9.7 sebagai modul kriptografi pada Cryptoboard
 2. Mensimulasikan fungsi kriptografi pada SBC yang dikendalikan oleh PC

Embeddable - Single Board Computer (Embeddable - SBC) dan SystemOnChip (SOC)

Apa itu SBC

Mikrokomputer dahulu secara tipikal terdiri dari setengah lusin (atau lebih) circuit-board yang dicolokkan ke sebuah backplane yang diimplementasikan sebagai central processor unit (CPU), memory, disk controllers, dan port serial atau paralel. Mikrokomputer berbasis backplane ini digunakan untuk akuisisi data, kendali proses, dan untuk riset. Tapi secara umum ukurannya terlalu besar untuk digunakan sebagai intelligence-embedded dalam sebuah devais.

Di awal tahun 80-an, teknologi rangkaian terintegrasi (IC) membuktikan bahwa fungsi-fungsi yang sebelumnya diimplementasikan dengan circuit-boards dapat dijejalkan dalam sebuah chip logic “integrasi skala besar“ (LSI). Chip LSI untuk CPU, memory, media penyimpanan, dan port serial atau paralel sekarang menghasilkan implementasi sistem mikrokomputer secara lengkap pada sebuah “single board computer“ (SBC) yang mampu menjalankan operasi CP/M (Commercial Disk Operating System).

Pangsa pasar “Embedded-SBC“

Seperti halnya “Big Board” (sebutan untuk sistem dengan menggunakan board dalam ukuran besar) , “Little Board (sebutan untuk sistem mikrokomputer yang sudah diimplementasikan dengan LSI) menggunakan sebuah CPU Z80 dan dikhususkan untuk sistem operasi CP/M. Dan jelas, lebih kecil ukurannya, seukuran footprint sebuah floppy disk drive (5.75 x 8.9 inchi). Hal ini berkat kombinasi yang unik dari kekompakan, kesederhanaan, reliability, dan biaya murah, “Little Bird” memudahkan implementasi untuk sistem operasi commercial-disk, yang akan secara mudah juga diintegrasikan ke devais non-komputer.

Itulah awal kelahiran pangsa pasar embedded SBC , yang sekarang diramaikan oleh ratusan produsen manufaktur SBC yang menghasilkan ratusan jenis SBC dengan target aplikasi computer untuk embedded sistem.

Awalnya, tiap produk SBC dibentuk unik, baik secara arsitektur maupun fisik. Hal ini sesuai dengan kebutuhan embedded system yang diinginkan, dikombinasikan dengan pilihan processor dan piranti pengendali yang tersedia. Bahkan tidak ada standard yang menjadi pegangan bagi pengembang SBC dalam hal spesifikasi fungsional dan mekanikal.

Lahirnya PC-compatible SBC

Pada pertengahan tahun 80-an, IBM PC mengadakan riset untuk mengembangkan kompatibilitas embedded system untuk aplikasi non-desktop, dengan dua alasan :

- Peningkatan kinerja hardware ; chipset PC dan kompatibilitas pirantinya dapat membentuk sistem berbiaya murah, sederhana, dan mudah didukung pengembangannya.
- Peningkatan kinerja software ; kompatibilitas PC memungkinkan dihasilkannya keuntungan sistem operasi PC (mulai dari DOS sampai dengan Windows), bahasa, perangkat, dan software aplikasi.

Salah satu hasil mikrokomputer-PC-compatible adalah dengan dihasilkannya plug-in-card oleh IBM PC (menggunakan bus ISA). Hasil lainnya ada juga yang diimplementasikan sebagai standalone-system pada sebuah single-board. Ada juga yang diadaptasikan untuk bus dalam rangkain computer di industri (STD, VME).

Dalam hal SBC berkarakteristik embeddable-non-backplane, trend yang berkembang adalah menghasilkan devais yang memiliki kompatibilitas tinggi bila diintegrasikan dengan PC. Beberapa prasyarat yang menjadi acuan :

- Ukuran board kecil (5.75 x 8.0 inchi) ; sistem lengkap pada sebuah board, yang menyediakan fungsi modul plug-on.
- ISA “slot boards” (full-length, 13.8 x 4.8 inchi ; half-length, 7.1 x 4.8 inchi) ; seperti yang digunakan card plug-in IBM PC, meskipun backplane-oriented, tapi bisa juga difungsikan standalone (tanpa backplane)
- Modul PC/104 (3.6 x 3.8 inchi) ; compact, modul-modul tersusun dengan fitur pin-and-socket antar board yang sudah reliabel dan bus-expansion.

Dan dengan munculnya PCI, maka hadir juga :

- PC/104-Plus ; PCI ditambahkan ke PC/104
- EBX ; PC/104-Plus ditambahkan ke “Little Board“

Tidak semua SBC dibangun dengan model-model seperti ini. Bahkan, tidak semua yang PC-compatible (x86/DOS/Windows). Seiring dengan perkembangannya, makin banyak board non-standard yang dibangun sesuai dengan kebutuhan tanpa mempertimbangkan aspek ukuran board dan arsitektur prosesor. Bentuk seperti ini kemudian dikenal dengan sebutan arsitektur “Wintel” (PC-compatible).

Aspek Perubahan

Beberapa faktor penting yang menjadi tantangan bagi pengembang SBC adalah :

- Permintaan terhadap sistem dengan embedded-intelligence yang bertambah banyak ; bahkan produk paling kecil dan paling murah sekali pun dituntut memiliki dasar embedded-intelligence. Juga dituntut bersifat user-friendly dengan berbagai menu tampilan grafis atau antar muka suara.
- Multi-konektivitas ; terjadi peningkatan kebutuhan agar perangkat elektronik dapat saling terkoneksi, baik melalui kabel atau tanpa kabel.

Devais ini haruslah dilengkapi dengan standard protocol TCP/IP untuk koneksi Internet (TCP/IP, PPP, HTTP, FTP).

- Ketersediaan periferan dan bus interface yang dapat di-upgrade ; meskipun standard interkoneksi yang populer tersedia dan tahan lama (bandingkan antara Centronics dan RS232), kemampuan untuk menggantikan komponen lama haruslah tersedia. Hampir dua dekade setelah munculnya PC, bus ISA secara global mulai tergantikan oleh PCI. Sekarang, USB malah mendominasi, dengan menggantikan fungsi port serial, paralel, dan PS/2. Ethernet dan FireWire sekarang sudah mulai banyak digunakan. SCSI sekarang mulai menjadi pilihan utama dalam PC (selain Apple). Kecenderungan sekarang adalah membangun sistem backplane-free dengan mekanisme ekspansi melalui antar muka serial medium dan high-speed (USB, IrDA, FireWire, Ethernet,...).
- Prosesor dengan system-on-chip berorientasi pada aplikasi ; sejumlah modul integrasi ARM, MIPS, PowerPC, dan x86 berbasis sistem one-chip sedang dikembangkan untuk mendukung spesifikasi wide-array dari produk-produk dalam jumlah besar dan yang harga jualnya sangat dipengaruhi oleh aplikasi yang dimiliki. Saat ini, prosesor berbasis “application-on-chip” menghadirkan tantangan untuk menghadirkan generasi terbaru SBC dengan kualifikasi “high-integration” , “high-performance” , dan “highly cost-effective”. Sejumlah System-On-Chip (SOC) akhirnya melepaskan kompatibilitasnya terhadap arsitektur x86 demi keuntungan atas biaya, daya terpakai, dan integrasi.
- Embedded Linux ; dalam jangka waktu singkat, Linux langsung merambah pada segala aspek komputerisasi, menawarkan kelebihan dalam hal biaya, solusi open-source dengan dukungan yang kuat untuk jaringan, komunikasi, internet, grafis, dan lainnya. Meskipun awalnya adalah turunan dari Unix untuk PC, Linux sekarang sudah dapat dijadikan sebagai penggabungan berbagai prosesor untuk kepentingan sistem operasi yang embedded. Konsekuensinya, terjadi peningkatan riset dan penelitian pada Sistem Operasi yang mendukung penuh bermacam-macam arsitektur (diatas x86). Hal ini seiring dengan pesatnya perkembangan kapabilitas dan pengembangan sisi arsitektural bawaan Linux, menghasilkan kompetisi baru dalam hal arsitektur prosesor.

Atas dasar hal-hal tersebut, maka semakin jelaslah arah perubahan pada pangsa pasar embedded-SBC.

Sebuah teori yang sedikit mengacaukan

Merunut pada koalisi antara aspek arsitektur PC dan standard form-factor embedded-SBC, tidak dimungkinkan untuk meletakkan dua SBC pada suatu tempat secara bersamaan meskipun terdapat kemiripan. Arsitektur PC menyebabkan terjadinya penurunan orde (dalam beberapa bentuk dan ukuran) sehingga muncul kekacauan itu. Hal ini yang terjadi dalam dua dekade terakhir.

Kini, dengan adanya norma platform yang diramaikan dengan kehadiran jenis antar muka baru (USB, FireWire, Bluetooth), arsitektur (MIPS, PowerPC, ARM), dan sistem operasi (Linux), pasar embedded-SBC memasuki tahap baru perkembangannya. Akan ditandai dengan karakteristik keberagaman sistem operasi yang digunakan, arsitektur prosesor, antar muka periferal, dan faktor pembentuk secara fisik.

Beberapa hal tentang Linux pada SBC embedded-PC

Saat ini, hampir seluruh vendor SBC yang kompatibel dengan PC mengklaim bahwa produk mereka mendukung embedded Linux dan pengadaan perangkat lunak pendukungnya, baik memproduksinya secara langsung maupun dengan vendor pihak ketiga.

Secara umum, Linux mendukung embedded-SBC yang kompatibel dengan PC, dengan menyediakan jenis chipset yang digunakan secara umum dan yang digunakan oleh vendor tertentu. Vendor SBC selalu memaparkan spesifikasi versi Linux yang telah diujicoba diimplantkan pada SBC tersebut, bagaimana ujicoba tersebut dilakukan, antar muka apa saja yang digunakan, dan fungsi-fungsi yang belum diujicoba atau belum dapat didukung.

Beberapa fakta yang menjadi pembandingan dalam memilih jenis SBC :

- Mode pengendali tampilan di atas VGA
- Panel pengendali sinyal LCD
- SCSI
- PCMCIA
- Disk solid-state onboard
- Fungsi-fungsi nonstandard seperti timer watchdog, Input/Output digital, dan Input/Output analog.
- Ethernet (pada beberapa kasus)

Dengan pengetahuan bahwa drive Linux atau adanya dukungan dari kernel untuk chip yang digunakan cukup memuaskan, tapi itu tidaklah cukup. Vendor SBC terkadang mengambil jalan pintas untuk menghemat biaya produksi. Dan yang di-bypass adalah proses ujicoba tersebut. Konsekuensinya, kompatibilitasnya menjadi dipertanyakan.

Untungnya, pengguna Linux pada embedded-SBC dapat membaca dan mempelajari source-code driver yang tersedia. Banyak programmer yang menguasai hal ini.

SBC mini untuk proyek berbasis embedded-Linux

Sebagaimana SBC non-standard yang terkadang ukurannya lebih besar daripada modul PC/104, ada juga yang ukurannya lebih kecil. Hal ini makin memperluas kemungkinan penggunaan embedded-Linux untuk berbagai aplikasi, yang pada tulisan ini akan digunakan untuk mendukung sistem keamanan komunikasi dan tukar menukar data. Fungsi-fungsi yang ada pada komputer secara umum, termasuk CPU, memory

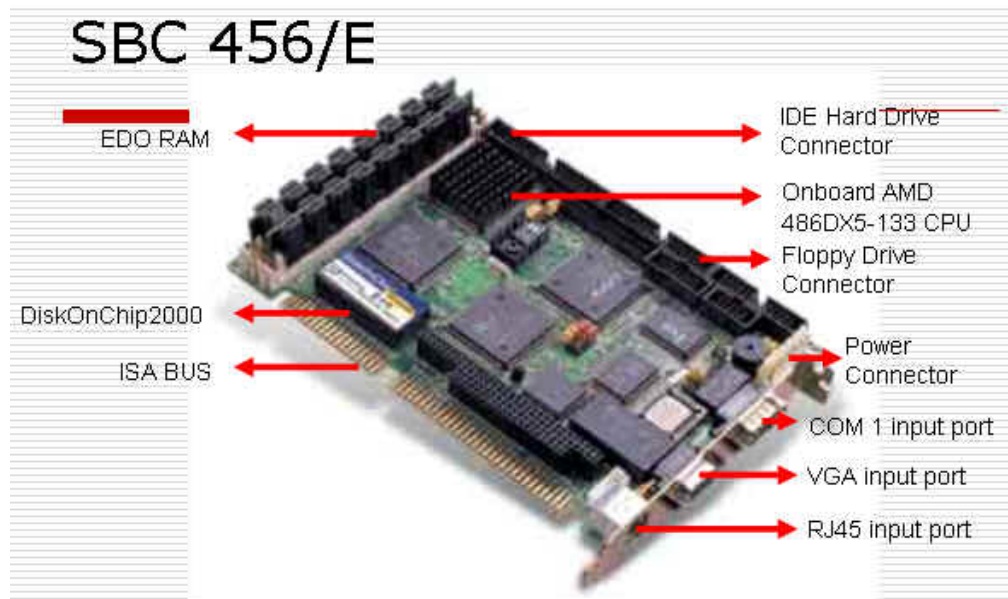
program, “solid-state-disk”, port serial dan paralel, antar muka tampilan, dan antar muka jaringan, terangkum dalam ruang berukuran belasan inchi.

Beberapa produk juga menyediakan slot CompactFlash atau PCMCIA, yang mampu memfasilitasi baik ekspansi memory maupun antar muka peripheral. Banyak juga yang mendukung operasi menggunakan tenaga baterai, sehingga devais yang dihasilkan bisa portable.

Hal yang patut dicatat juga adalah adanya kemungkinan minimnya kemampuan fungsi plug-and-play di antara SBC ini mengingat bentuk ini adalah miniature PC/104 yang berbeda form-factor. Meskipun ada trend untuk mencocokkan antara ukuran dan tipe konektor modul memory DIMM atau SIMM, belum ada konsistensi tentang pengolahan sinyal yang didapat oleh konektor modul. Ditambah lagi, trend produk baru yang mengintegrasikan prosesor high-integration system-on-chip (StrongARM, Elan, Etrax, dsb) yang tidak kompatibel dengan x86.

Bagaimana pun juga, disamping minimnya kemampuan modul plug-and-play, adanya port embedded-Linux pada SBC mini ini, yang dikombinasikan dengan rasio integrasi atau ukuran yang mengagumkan, menjadikan produk ini memiliki daya tarik tinggi untuk diterapkan pada aplikasi yang tidak memiliki toleransi terhadap ukuran standard SBC. Harapannya, dengan menggunakan salah satu SBC mini ini, akan mengurangi biaya, resiko, dan pemborosan waktu proses pada pengembangan suatu embedded-computer.

Single Board Computer tipe SBC 456/E



Gambar 1 Single Board Computer seri 456/E

Single Board Computer (SBC) seri 456/E yang digunakan pada penelitian ini diproduksi oleh AAEON® yang didistribusikan oleh Lely Tempustech,LLC.

Fitur yang disediakan :

- Prosesor AMD 486DX5-133 CPU (SQFP) onboard
- C&T 65550 LCD controller dengan memory EDO maksimalam 128 MB
- 10Base-T Ethernet
- Mendukung DiskOnChip samai dengan 288MB
- PC/104 expansion connector

Spesifikasi lengkapnya dapat dilihat dalam tabel berikut :

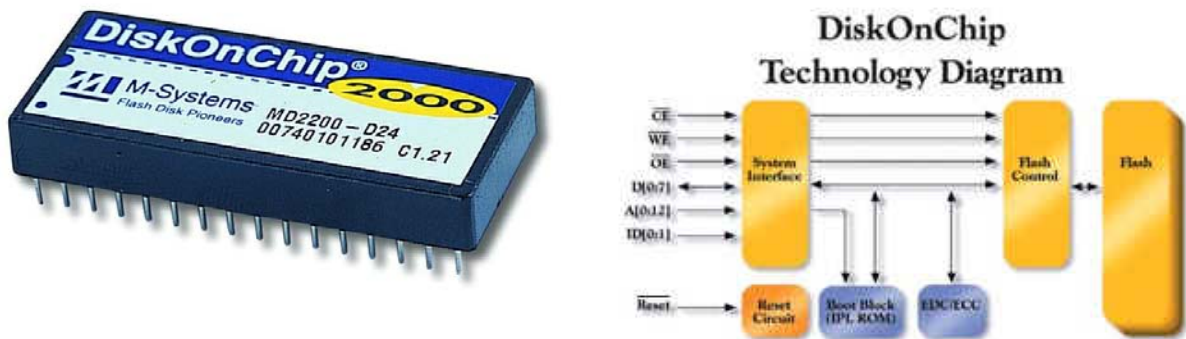
• Sistem :	
CPU	AMD DX5-133 (SQFP)
System Memory	FPM/EDO SIMM x 2, Max. 128MB
2nd Cache Memory	Onboard 128KB L2 Cache
Chipset	Ali 1487/1489
I/O Chipset	ITE IT8661F, Fully 16-bit I/O decoded
BIOS	AMI 128KB Flash BIOS
SSD	Supports DiskOnChip up to 288MB
Watchdog timer	Can generate a system reset, IRQ15 or NMI
Expansion Interface	ISA interface, PC/104 + connector x 1
Battery	Lithium battery
DMA	7 DMA channels (8237 equivalent)
Interrupt	15 interrupt levels (8259 equivalent)
Power Supply voltage	+5V (4.75V to 5.25V), -12V, +3.3V, AT.

Size/Weight	7.3" (L) x 4.8" (W) (185 mm x 122 mm), 0.66lb. (0.3Kg)
Operating temperature	32 F to 140 F (0 C to 60 C)
• Input/Output	
MIO	IDE x 1, FDD x 1, KB x 1, Mouse x 1, RS-232 x 1, RS-232/422/485 x 1, Parallel x1
Ethernet	Realtek RTL8029AS, 10Base-T RJ-45 connector x 1
• Display	
Chipset	C&T 65550
Memory size	1MB onboard, 2MB SDRAM optional
Resolution	1024 x 768@ 64K colors
LCD Interface	36 bit TFT/DSTN/MONO LCD panel
• Packing List	
IDE cable	9657456000
FDD cable	9657456000
COM port cable kit (1 parallel, 1 serial)	9657456000
Ethernet Cable	1701100203
KB/Mouse cable	1700060191
Quick Installation Guide	
Utility CD	
• Optional	
PCM-3524	LVDS Transmitter/Receiver Module

Tabel 1 Spesifikasi Lengkap SBC 456/E

DiskOnChip 2000

DiskOnChip dikenal luas sebagai modul flash-memory yang diproduksi oleh M-systems®. Dengan chip tunggal dari M-systems dan teknologi TrueFFS, DiskOnChip menyediakan fitur dengan performa tinggi, daya tahan tinggi dan menjadi solusi berharga relative murah untuk aplikasi menggunakan flash-disk.



Gambar 2 Fisik dan Diagram Teknologi DiskOnChip 2000

Fitur yang disediakan :

- Disk solid-state untuk kepentingan industri yang bersifat embedded, bootable, padat dan murah
- Error detection and Correction (EDC/ECC)
- EPROM Compatible Electrical Interface
- Performa tinggi dengan kemampuan baca/tulis 1.4/0.5 MB/sec
- Built-in TrueFFS untuk Full Hard Disk Emulation
- Memory Window 8KB

Spesifikasi lengkapnya dapat dilihat pada tabel berikut :

Memory Capacity (DiskOnChip2000)		16 ~ 576 MB
Memory Capacity (DiskOnChip2000 Millennium)		8 MB
Package Type		32-pin DIP
Operating Voltage		3.3V or 5V
Power Consumption (5V)	Active	40 mA
	Stand by (Low Profile)	60 μ A
	Stand by (High Profile)	240 μ A
Power Consumption (3.3V)	Active	30 mA
	Stand by (Low Profile)	40 μ A
	Stand by (High Profile)	120 μ A
Sustained Transfer Rate	Read	1.4 MB/sec
	Write	0.5 MB/sec
Burst Transfer Rate	Read	5 MB/sec
	Write	5 MB/sec
Temperature (Standard)	Operating	0 ~ 70 °C
	Storage	-55 ~ 100 °C
Temperature (Extended)	Operating	-40 ~ 85 °C
	Storage	-55 ~ 100 °C
Humidity		10 ~ 90% (Non-condensation)
EDC/ECC	For each 512-byte block of data: - Corrects up to two 10-bit symbols, including two random bit errors. - Corrects single bursts up to 11 bits. - Detects single bursts up to 31 bits and double bursts up to 11 bits. - Detects up to 4 random bit errors.	

Tabel 2 Spesifikasi Lengkap DiskOnChip 2000

OpenSSL

Salah satu objek dalam cakupan keamanan adalah keamanan protokol yang dipergunakan dalam komunikasi dan transaksi. Protokol, dalam konteks ini, merupakan suatu set aturan yang dipergunakan oleh komputer-komputer (yang terhubung dalam suatu jaringan) untuk saling berkomunikasi. Protokol yang paling banyak dipergunakan dalam Web merupakan set protokol TCP/IP. Sayangnya, protokol ini didesain tanpa memperhatikan faktor keamanan data. Oleh karena itu dibutuhkan suatu mekanisme tambahan untuk memperkuat keamanan protokol ini tanpa harus menggantinya dengan yang lain. Dengan protokol (beserta mekanismenya) yang aman serta implementasi yang benar, tingkat keamanan komunikasi dan transaksi Web dapat ditingkatkan. Salah satu metode penerapan keamanan protokol ini adalah dengan menerapkan Secure Socket Layer (SSL).

Mekanisme SSL ini terdiri dari beberapa komponen. Salah satu dari komponen yang berkaitan dengan sistem keamanan informasi dan dibahas dalam tulisan ini adalah komponen kriptografi yang terdapat di dalamnya. Komponen ini berfungsi melakukan pengolahan khusus terhadap data yang akan dikomunikasikan agar data tersebut tidak dapat diganggu oleh pihak luar. Komponen kriptografi dalam SSL tersusun dari beberapa algoritma matematis.

Dengan suatu algoritma yang baik serta implementasi yang benar, diharapkan SSL dapat berfungsi sesuai spesifikasi yang diharapkan: cepat, efisien, dan aman. Ketiga faktor inilah yang akan sangat menentukan nilai keamanan dan keterandalan dalam penggunaan SSL sebagai mekanisme pengamanan di tingkat protokol.

OpenSSL merupakan komunitas yang didasarkan pada model inisiatif OpenSource. Bedanya, komunitas OpenSSL lebih memfokuskan diri pada apa yang disebut pengembangan SSL secara terbuka. Komunitas inilah yang banyak memberi dukungan teknis bagi terlaksananya tulisan ini.

OpenSSL

Title
FAQ
About
News
Documents
Source
Contribution
Support
Related

Welcome to the OpenSSL Project

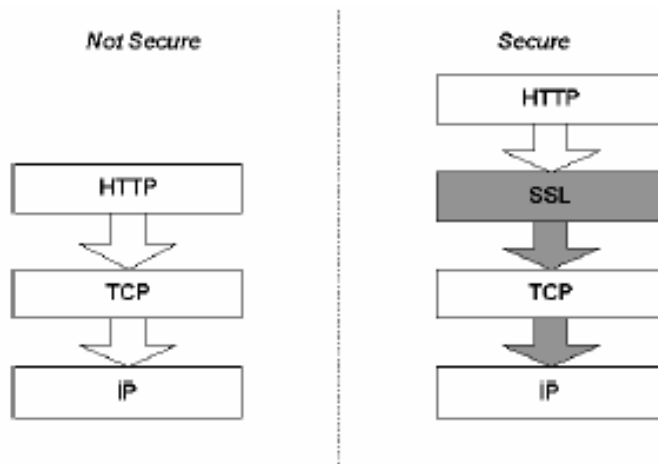
The OpenSSL Project is a collaborative effort to develop a robust, commercial-grade, full-featured, and **Open Source** toolkit implementing the **Secure Sockets Layer (SSL v2/v3)** and **Transport Layer Security (TLS v1)** protocols as well as a full-strength general purpose cryptography library. The project is managed by a worldwide community of volunteers that use the Internet to communicate, plan, and develop the OpenSSL toolkit and its related documentation.

Why buy an SSL toolkit as a black-box when you can get an open one for free?

OpenSSL is based on the excellent SSlEay library developed by Eric A. Young and Tim J. Hudson. The OpenSSL toolkit is licensed under an Apache-style licence, which basically means that you are free to get and use it for commercial and non-commercial purposes subject to some

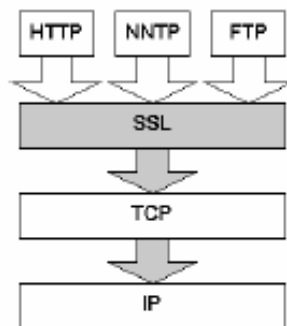
Gambar 3 Homepage OpenSSL

Para desainer SSL memutuskan untuk membuat protokol terpisah untuk menangani masalah keamanan. Mereka menambahkan lapisan (layer) tambahan pada arsitektur protokol Internet. Gambar 2.8 sebelah kiri menunjukkan susunan lapisan protokol dalam komunikasi Web. Gambar sebelah kanan menunjukkan penambahan lapisan SSL untuk memperkokoh keamanan. Dengan bertindak sebagai lapisan baru, SSL tidak banyak mengubah lapisan di atasnya dan dibawahnya. Interface aplikasi HTTP akan melihat SSL sebagai suatu lapisan yang hampir sama dengan lapisan TCP. Demikian juga halnya dengan lapisan TCP; ia akan melihat SSL sebagai suatu aplikasi lain yang menggunakan layanannya. Gambar 2.8 SSL merupakan lapisan terpisah dalam susunan protokol Internet



Gambar 4 SSL merupakan lapisan terpisah dalam susunan protokol Internet

Selain hanya membutuhkan sedikit perubahan pada implementasi yang sudah ada, pendekatan ini memiliki keuntungan lain: SSL mampu mendukung aplikasi lain selain HTTP. Motivasi utama perancangan SSL adalah untuk meningkatkan keamanan Web, namun pada Gambar 2.9, SSL dapat juga dipergunakan untuk menambahkan keamanan pada aplikasi Internet lainnya seperti Net News Transfer Protocol (NNTP) dan FileTransfer Protocol (FTP).



Gambar 5 SSL juga dapat menangani keamanan aplikasi lain

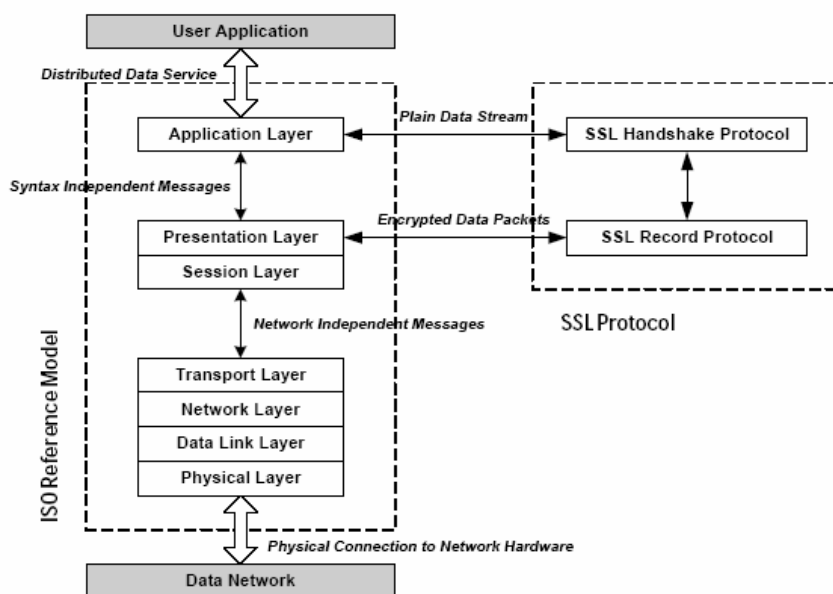
Mekanisme Kerja SSL

Transmission Control Protocol/Internet Protocol (TCP/IP) mengatur transportasi dan pembentukan rute data di Internet. Protokol lain, seperti HyperText Transport Protocol (HTTP), Lightweight Directory Access Protocol (LDAP), atau Internet Messaging Access Protocol (IMAP), berjalan 'diatas' TCP/IP. Seluruh protokol tersebut menggunakan TCP/IP untuk mendukung aplikasi tipikal seperti menampilkan halaman web atau menjalankan server e-mail.

Protokol SSL berjalan diatas TCP/IP dan dibawah protokol-level-atas seperti HTTP atau IMAP. Ia menggunakan TCP/IP atas nama protokol-level-atas, dan dalam prosesnya memungkinkan sebuah SSL-enabled server untuk mengautentikasi dirinya sendiri kepada SSL-enabled client, serta memungkinkan client untuk mengautentikasi dirinya sendiri kepada server, dan memungkinkan kedua mesin untuk membangun sebuah koneksi terenkripsi.

Hal diatas berimplikasi terhadap munculnya komponen fundamental dalam komunikasi di Internet dan di jaringan yang menggunakan TCP/IP :

- **Autentikasi server SSL** memungkinkan seorang pengguna untuk memastikan identitas server. Software SSL-enabled client dapat menggunakan teknik standar public-key cryptography untuk memeriksa keabsahan sertifikat server dan public ID yang dikeluarkan oleh Certificate Authority (CA). CA ini harus ada dalam daftar CA yang dipercayai oleh client. Konfirmasi ini bisa menjadi sangat penting jika, misalnya, mengirimkan sebuah nomor kartu kredit melalui jaringan dan ingin memastikan identitas server penerima nomor tersebut.
- **Autentikasi client SSL** memungkinkan sebuah server untuk memastikan identitas pengguna. Dengan menggunakan teknik yang sama seperti yang digunakan pada autentikasi server, software SSL pada server dapat memeriksa keabsahan sertifikat client dan public ID. Peran CA dalam hal ini sama seperti yang telah disebutkan sebelumnya. Konfirmasi ini bisa menjadi sangat penting bagi server jika, misalnya, server tersebut adalah sebuah bank yang akan mengirimkan informasi finansial rahasia kepada pelanggannya. Pemeriksaan identitas penerima, dalam hal ini, sangat penting.
- **Enkripsi koneksi SSL** membutuhkan kondisi dimana semua informasi yang dikirimkan antara client dan server dienkripsikan/didekripsikan oleh kedua belah pihak. Hal ini menyediakan tingkat kerahasiaan yang tinggi. Kerahasiaan merupakan hal penting bagi kedua pihak yang akan melakukan komunikasi pribadi. Sebagai tambahan, semua data yang dikirimkan melalui koneksi SSL dilindungi oleh mekanisme yang akan mendeteksi tampering – yaitu kemampuan untuk mendeteksi apakah data tersebut diubah selama dalam perjalanan.



Gambar 6 Letak SSL dalam model ISO Reference

OpenSSL-0.9.7

Source code OpenSSL yang digunakan memiliki nomor versi openssl-0.9.7d. Hingga saat penulisan, versi OpenSSL tersebut belum berada dalam status stable. Ini berarti bahwa versi OpenSSL masih akan mengalami perbaikan dan penyempurnaan; tidak ada yang menjamin bahwa versi ini dapat berfungsi secara penuh. Versi ini dipilih sebagai daerah implementasi dengan pertimbangan bahwa dalam peluncurannya, telah disertai dengan bugfixes.

OpenSSL, seperti yang telah dijelaskan di atas, merupakan usaha kolaboratif untuk membangun suatu toolkit OpenSource yang dapat memfungsikan Secure Socket Layer (SSL v2/v3) dan Transport Layer Secure (TLS v1) dan juga untuk membangun suatu cryptography library umum yang kuat. Dengan demikian, OpenSSL dituntut untuk dapat difungsikan sepenuhnya seperti implementasi SSL/TLS lainnya dan memiliki keamanan yang kuat, yang memenuhi berbagai macam tingkat kebutuhan akan komunikasi data yang aman.

Source code OpenSSL ini menyediakan bukan hanya fasilitas SSL/TLS saja, tetapi juga sebuah toolkit kriptografi yang dapat digunakan untuk berbagai macam keperluan yang menyangkut pengamanan data dan komunikasi. Toolkit OpenSSL ini berisi :

1. **libssl.a** : merupakan implementasi SSLv2, SSLv3, TLSv1 dan kode-kode yang dibutuhkan untuk mendukung SSLv2, SSLv3, dan TLSv1 pada sebuah server dan client.
2. **libcrypto.a** : enkripsi umum dan X.509 v1/v3 yang dibutuhkan oleh SSL/TLS namun bukan merupakan bagian utama dari SSL/TLS tersebut. Bagian ini berisi rutin untuk yang menangani :

- a. Ciphers**
 - i. libdes, library enkripsi DES termasuk 15 mode variannya dan rutin untuk membaca password dari keyboard.
 - ii. enkripsi RC4.
 - iii. enkripsi RC2, dengan 4 mode: ECB, CBC, CFB, dan OFB.
 - iv. enkripsi Blowfish, dengan 4 mode: ECB, CBC, CFB, dan OFB.
 - v. enkripsi IDEA, dengan 4 mode: ECB, CBC, CFB, dan OFB
 - vi. enkripsi AES 128, 192, dan 256 bit, dengan 2 mode: ECB, dan CBC.
- b. Digest**
 - i. MD5 dan MD2 message digest algorithm.
 - ii. SHA dan SHA-1 message digest algorithm.
 - iii. MDC2 message digest.
- c. Public Key**
 - i. RSA: enkripsi/dekripsi/pembangkitan
 - ii. DSA: enkripsi/dekripsi/pembangkitan
 - iii. Diffie-Hellman: pertukaran kunci/pembangkitan kunci
- d. Sertifikat X.509.3**
- e. Sistem**
- f. Struktur Data**
- 3. **openssl** : command-line tool yang dapat digunakan untuk :
 - a. Pembuatan parameter kunci RSA, DH, dan DSA
 - b. Pembuatan sertifikat X.509, CSR, dan CRL
 - c. Perhitungan message digest
 - d. Enkripsi dan dekripsi dengan cipher
 - e. Tes SSL/TLS pada client dan server
 - f. Penanganan S/MIME pada enkripsi mail

Struktur Umum Kode OpenSSL

Source code OpenSSL-0.9.7, yang digunakan dalam implementasi ini, datang dalam bentuk tarballs. Format ini merupakan format file source code terkompresi yang lazim digunakan dalam lingkungan Linux/UNIX. Source code ini dapat diambil di <ftp://ftp.openssl.org/source> dengan nama `openssl-0.9.7d.tar.gz`. File ini kemudian diekstrak ke direktori bernama `openssl-0.9.7d/` dengan menggunakan perintah (di Linux) :

```
# tar -xzvf openssl-0.9.7d.tar.gz
```

Setelah diekstraksi, barulah source code OpenSSL dapat dibaca dan dianalisis. Source code ini ditulis dalam bahasa pemrograman C (ANSI C). Di dalam direktori ini terdapat beberapa subdirektori dan beberapa file. Ukuran total source code ini setelah diekstrak adalah sekitar 17 MB.

apps	bugs	certs
crypto	demos	doc
include	MacOS	ms
os2	perl	shlib
ssl	tests	times
tools	util	VMS

Tabel 3 Subdirektori yang berada langsung dibawah direktori utama OpenSSL

CHANGES	CHANGES.SSLeay	config
Configure	e_os2.h	e_os.h
FAQ	INSTALL	install.com
INSTALL.DJGPP	INSTALL.MacOS	INSTALL.OS2
INSTALL.VMS	INSTALL.W32	INSTALL.WCE
LICENSE	Makefile	Makefile.org
Makefile.ssl	makevms.com	NEWS
openssl.doxy	openssl.spec	PROBLEMS
README	README.ASN1	README.ENGINE

Tabel 4 File-file yang berada langsung dibawah direktori utama

Secara sederhana, seluruh *source code* dalam direktori utama ini dapat dikompilasi (*compile*) dengan menggunakan *compiler C* untuk menghasilkan file-file program yang akan digunakan untuk menjalankan OpenSSL secara keseluruhan. File-file yang ada dalam direktori utama ini tidak hanya berupa file *source code C* (*.o *.c *.h), tetapi juga termasuk didalamnya file-file konfigurasi kompilasi (terutama untuk kompilasi di system Linux) dan file-file teks dokumentasi untuk membantu pemasangan dan konfigurasi OpenSSL.

Karena *source code* ini merupakan suatu kode yang sangat kompleks, maka implementasi dilakukan hanya pada file (dan direktori) yang memang benar-benar berpengaruh. File-file pada direktori utama yang dianggap cukup penting dalam implementasi ini adalah :

1. **config** : file *script* ini mengatur mekanisme pendeteksian sistem operasi yang digunakan dan juga menjalankan file *script* instalasi utama, Configure.
2. **Configure** : file *script* utama yang akan mempersiapkan sistem dan *compiler* dalam pemasangan OpenSSL.
3. **INSTALL** : file teks yang berisi panduan instalasi OpenSSL.
4. **install** : file yang mengatur penempatan file-file program pada saat instalasi berlangsung.
5. **makefile.org** : file yang mengatur parameter-parameter dan mekanisme kompilasi secara detil pada platform Linux.
6. **README** : file teks yang berisi deskripsi singkat mengenai OpenSSL dan tinjauan singkat mengenai *toolkit* OpenSSL.

Struktur Direktori crypto/aes

Direktori **crypto** merupakan direktori yang berada di bawah direktori utama OpenSSL. Direktori ini berisi seluruh *source code* dari *cipher* yang digunakan dalam OpenSSL. Karena implementasi ini hanya menangani masalah *cipher*, maka direktori inilah yang paling penting dalam implementasi ini. Direktori ini juga merupakan *cryptographic library* yang dapat dipergunakan untuk berbagai keperluan diluar SSL itu sendiri. Setiap *cipher* berada di dalam direktori tersendiri dan memiliki struktur yang hampir sama (standar). Untuk implementasi ini, perhatian akan lebih difokuskan pada direktori **aes**, direktori tempat *source code* algoritma AES (Rijndael) berada.

Direktori crypto/aes ini berisi 11 file. File-file tersebut adalah :

1. **aes.h** : merupakan file *header* utama yang hanya menangani pemanggilan fungsifungsi utama pada algoritma ini. File ini juga merupakan *header* perantara antara modul-modul *cipher* pada direktori aes ini dengan sistem OpenSSL secara keseluruhan.
2. **aes_cbc.c** : merupakan file yang menangani pengolahan data pada mode CBC bersama-sama dengan fungsi enkripsi/dekripsi utama.
3. **aes_cfb.c** : merupakan file yang menangani pengolahan data pada mode CFB bersama-sama dengan fungsi enkripsi/dekripsi utama.
4. **aes_core.c** : merupakan file enkripsi/dekripsi utama yang memuat *ciphe source code* yang sebenarnya. File ini memuat versi optimasi Rijndael yang dikompertisikan oleh NIST, dengan sedikit perubahan untuk *error control* dan kompatibilitas dengan API OpenSSL.
5. **aes_ctr.c** : merupakan file yang menangani pengolahan data pada mode CTR bersama-sama dengan fungsi enkripsi/dekripsi utama.
6. **aes_ecb.c** : merupakan file yang menangani pengolahan data pada mode ECB bersama-sama dengan fungsi enkripsi/dekripsi utama.
7. **aes_locl.h** : merupakan file *header* lokal, *header* yang sebenarnya untuk *sourcecode* algoritma ini.
8. **aes_misc.c** : merupakan file yang merupakan kode-kode tambahan untuk keperluan kompatibilitas.
9. **aes_ofb.c** : merupakan file yang menangani pengolahan data pada mode OFB bersama-sama dengan fungsi enkripsi/dekripsi utama.
10. **Makefile.ssl** : merupakan file konfigurasi yang mengatur mekanisme dan metode kompilasi terhadap kode-kode yang ada di direktori ini.
11. **README** : merupakan deskripsi singkat mengenai versi AES yang digunakan dalam OpenSSL ini.

Sistem Operasi eCos-2.0.i386.linux

Tentang eCos

eCos disediakan sebagai system open source yang didukung oleh GNU. Para pengembang telah melengkapi dan membuka akses pada semua aspek sistem runtime-nya. Tak ada bagian yang disembunyikan, dan pengembang memiliki kebebasan untuk menguji, menambahkan, atau memodifikasi source code seperti yang diinginkan. Hak ini dilindungi oleh lisensi eCos. Lisensi ini juga menjamin para pengembang untuk mengembangkan secara bebas dan mendistribusikan aplikasi-aplikasi berbasis eCos. Komunitas pengembang eCos muncul sebagai wadah yang menampung semua inovasi semacam board ports, antar muka devais, dan komponen-komponen lainnya.

Salah satu kelebihan teknologi eCos adalah sistem konfigurasinya. Sistem konfigurasinya memperbolehkan pengembang untuk menentukan sendiri kebutuhan terhadap komponen run-time, baik secara fungsional maupun dari segi implementasi, yang secara tradisional sistem operasi telah memiliki fungsi dan implementasi bawaan. Pengembang eCos dapat membentuk sistem operasi sesuai spesifikasi aplikasi yang ingin dibangun dan dapat digunakan dalam berbagai sistem embedded. Pengembang juga dapat mengkonfigurasi source-code eCos seminimal mungkin dengan menghapus fungsi dan fitur yang tidak diinginkan. Konfigurasi sistem juga menjadikan eCos sebagai sebuah arsitektur komponen. Komponen dapat diperoleh dari berbagai sumber yang sesuai standard rilis eCos, pengembang pihak ketiga, dan kontributor open source.

eCos didesain untuk dapat berfungsi dalam berbagai tipe arsitektur dan platform, termasuk arsitektur 16, 32, dan 64 bit, MPU, MCU, dan DSP. Kernel, library, dan komponen eCos disusun pada Hardware Abstraction Layer (HAL), dan sekali HAL dan antar muka devais yang bersesuaian dipasangkan pada arsitektur prosesor dan board. Versi terakhir eCos mendukung sepuluh jenis arsitektur yang berbeda (ARM, Hitachi H8300, Intel x86, MIPS, Matsushita AM3x, Motorola 68k, PowerPC, SuperH, SPARC, dan NEC V8xx) termasuk varian dari masing-masing arsitektur dan evaluation board-nya. Beberapa port baru juga sedang dikembangkan dan akan dirilis.

eCos juga telah didesain untuk mendukung aplikasi real-time dengan fitur full preemptability, minimal interrupt latency, dan semua sinkronisasi yang mungkin, aturan penjadwalan, dan mekanisme interrupt yang dibutuhkan oleh tipe aplikasi ini. eCos juga menyediakan semua kebutuhan fungsional untuk aplikasi embedded secara umum termasuk devais antar muka, manajemen memory, exception handling, C, library matematis, dan sebagainya. Sebagai tambahan untuk dukungan runtime, sistem eCos memuat semua piranti pendukung aplikasi embedded, termasuk konfigurasi perangkat lunak eCos dan piranti pembangun, compiler berbasis GNU, assembler, linker, debugger, dan simulator.

Secara fungsional, bagian-bagian berikut biasanya telah disediakan :

- Hardware Abstraction Layer (HAL)
- Real-time kernel
 - Interrupt handling

- Exception handling
- Pilihan skedul
- Dukungan thread
- Berbagai tipe sinkronisasi
- Timer, counter, dan alarm
- Pilihan pengalokasian memory
- Dukungan debug dan instrumentasi
- μ ITRON 3.0 kompatibel API
- POSIX kompatibel API
- ISO C dan library matematis
- Serial, ethernet, antar muka devais wallclock dan watchdog
- Dukungan USB
- Jaringan TCP/IP
- Dukungan debug GDB

Spesifikasi Sistem

Distribusi eCos tersedia untuk versi Linux maupun Windows. Versi Linux telah diuji coba dalam distribusi Red Hat, SuSE, dan Debian untuk arsitektur x86 dan seharusnya dapat bekerja di semua varian Linux. Versi Windows telah diuji coba pada Microsoft Windows 2000 Professional dan seharusnya juga bekerja pada Windows 95, Windows 98, Windows ME, dan Windows XP. Bagaimanapun, membangun cross toolchains GNU eCos pada Windows 95/98/ME tidak dapat dipercaya. Direkomendasikan untuk menghindari membangun development tools pada platform ini.

Distribusi eCos disuplai dengan dukungan penuh untuk konfigurasi eCos pada berbagai platform melalui perangkat konfigurasi grafis maupun command-line. Diinginkan, sistem ini dapat terhubung dengan tools pengembangan GNU yang tersedia secara bebas pada internet. Minimal tersedia gcc compiler, gdb debugger, dan binutils tools yang dibutuhkan untuk membangun eCos, link ke kode aplikasi dan undertake debugging.

eCos-2.0.i386.linux

Sistem operasi eCos-2.0.i386 dipaket dalam bentuk tarball eCos-2.0.i386.linux.tar.bz2 berukuran 15.084.304 bytes (sekitar 15MB). Setelah ekstraksi, diperoleh ukuran paket lengkap sebesar 136MB.

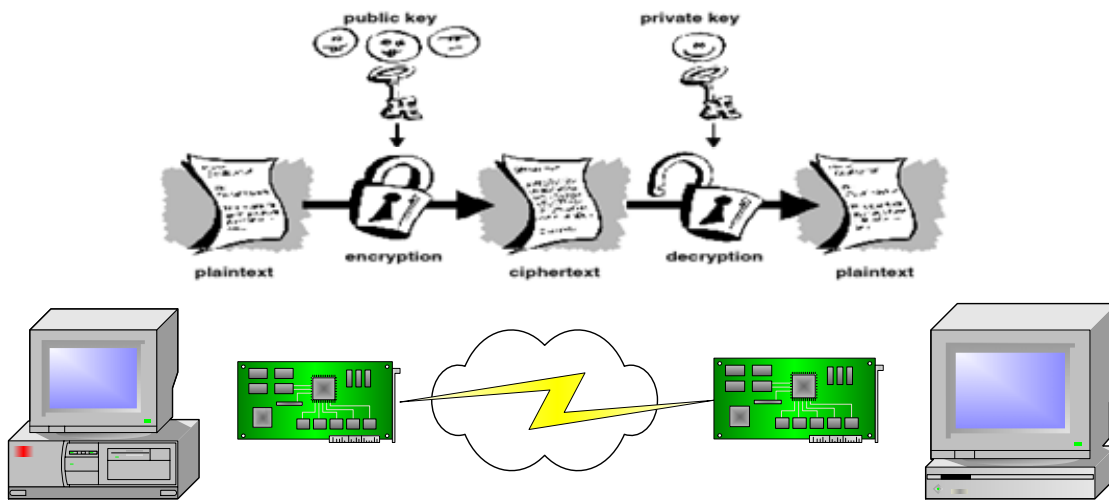
```
# tar -xjvf ecos-2.0.i386.linux.tar.bz2
# cd ecos-2.0/
# du -h ecos-2.0/
102M
```

Dengan sistem embedded berukuran kecil, tidak semua aplikasi dalam paket sistem operasi ini digunakan. Bagian terpenting yang akan digunakan adalah kernel dan beberapa aplikasi pendukung lainnya seperti library dan compiler engine, antar muka untuk ethernet, tampilan grafis, dan keyboard. Masih terdapat kemungkinan pemakaian aplikasi tambahan sesuai dengan kebutuhan dan spesifikasi sistem.

Kernel yang digunakan merupakan bawaan sistem, versi 2.0. Kernel dasar yang disediakan berukuran 1,8 MB. Dengan tambahan beberapa aplikasi tambahan, ukuran sistem operasi tidak akan lebih besar daripada 3 MB.

Desain Sistem

Ide desain sistem yang hendak dibangun diilustrasikan seperti pada gambar di bawah ini. Dengan penggunaan cryptoboard, seluruh proses enkripsi dan dekripsi dari data yang dikirim dan diterima oleh PC, dilakukan dengan aplikasi openssl yang ada pada cryptoboard. Diharapkan seluruh kunci yang digunakan dalam enkripsi dan dekripsi akan disimpan dalam memory cryptoboard. Hal ini menjadi aspek keamanan utama menggantikan sistem yang menggunakan kunci-kunci enkripsi dan dekripsi yang disimpan pada hardisk PC.



Gambar 7 Ide perancangan sistem

Bagian penting lainnya yang akan dilakukan adalah kustomisasi cryptoboard. Hal-hal penting yang menjadi perhatian adalah

- Mengatur jalur komunikasi dan interkoneksi antara PC dan cryptoboard. Hal ini dilakukan untuk menjamin keamanan data yang dikirim atau diterima dan kunci yang digunakan. Hal ini dilakukan dengan membangun antar muka yang sesuai untuk interkoneksi PC dan cryptoboard.
- Jaminan keamanan fisik terhadap cryptoboard. Diharapkan, bila cryptoboard dilepas dari PC, segala data dan kunci yang ada akan otomatis hilang atau terkunci. Dengan demikian, kejahatan dengan mencuri cryptoboard untuk membongkar data dan kunci yang tersimpan di dalamnya, dapat diatasi.

Untuk menghasilkan sistem tersebut, disusun garis besar pengerjaan sistem seperti ditunjukkan pada gambar berikut :



Gambar 8 Rencana Tahapan Pengerjaan

Rangkuman

Pada tulisan ini telah dibahas mengenai empat komponen utama penyusun cryptoboard ini yaitu : Single Board Computer (SBC) 456/E, DiskOnChip (DOC) 2000, OpenSSL 0.9.7d, dan Sistem Operasi eCos-2.0.i386.linux. Langkah penting berikutnya yang akan dilakukan adalah membuat antar muka antara cryptoboard dan PC.

Permasalahan yang diperkirakan akan muncul adalah memfungsikan cryptoboard agar menjadi peripheral tambahan PC. Termasuk pemilihan sistem operasi PC yang akan mengendalikan cryptoboard. Dari yang sudah dibahas sebelumnya, penulis memilih untuk menggunakan sistem operasi Linux untuk PC. Dengan ini, diharapkan kustomisasi komunikasi dan interkoneksi antara PC dan cryptoboard dapat dilakukan lebih mudah. Contoh lain adalah spesifikasi fisik SBC 456/E yang masih menggunakan slot ISA bus pada PC, sementara PC modern saat ini sudah mulai meninggalkan bus mode ini.

Penulis juga memperhitungkan kemungkinan-kemungkinan lain yang akan terjadi. Misalnya, penggunaan non-cryptoboard pada PC lain untuk menerima atau mengirim data terenkripsi dari dan oleh cryptoboard. Juga kemungkinan penggunaan cryptoboard pada sistem operasi selain Linux. Termasuk juga aplikasi-aplikasi keamanan lain yang dapat dimasukkan ke cryptoboard seperti IPsec dan sejenisnya.

Catatan Penulis

Saat tulisan ini dibuat, penulis sedang menyelesaikan bagian Instalasi dan Konfigurasi Paket mini eCos-2.0 pada SBC 456/E sebagai bagian dari Tugas Akhir pada Laboratorium IC-VLSI , Departemen Teknik Elektro - Institut Teknologi Bandung.

Bibliografi

Buku

Applied Cryptography - 2nd ed., Bruce Schneier, John Wiley & Sons Inc., 1996
SSL and TLS Essentials - Securing the Web, Stephen Thomas, John Wiley & Sons, 2000

Paper

AES Proposal : Rijndael, Joan Daemen and Vincent Rijmen, 1997
Announcing the Advanced Encryption Standard (AES), NIST, 2001

Web Resources

OpenSSL : www.openssl.org
eCos : <http://sources.redhat.com/ecos/about.html> , www.ecoscentric.com
DiskOnChip : www.m-sys.com
Opensource Initiative : www.opensource.org