

ISSN 0853-6740

LAPORAN TEKNIS BERKALA

Vol. 11 No. 4 Desember 2003

TEKNIK INFORMATIKA

**Perangkat Ajar Bahasa Inggris Dasar dan Kosakata
Berbasiskan Multimedia**

Januar Wahjudi; Sally Cemerlang; Hadi Gunawan; Hartawan

**Authentikasi pada Linux Terminal Server Project
Menggunakan Smart Card**

Wiedjaja; Noprianto

**Analisis dan Perancangan Router Simulator
untuk Mensimulasikan IOS pada Cisco Router
dalam Mendukung Program Cisco Networking Academy**

Johan M.K.; Ferry Sumendap; Joshua; Lay Iwan Setiawan

**Penerapan Konsep Berorientasi Objek dengan Unified Process
dan Unified Modeling Language dalam Perancangan
Aplikasi Web Distance Learning**

Mark P. Eliasaputra; Henny Surya Ningsih; Adriani Halim

**Analisis dan Perancangan Sistem Knowledge Object Repository
Studi Kasus: UPT Perangkat Lunak Universitas Bina Nusantara**

Andreas Utomo Kuswara; Edo Budi; Oto Joseph Santoso; Ani

**Perancangan Aplikasi Mobile Information Service untuk Mahasiswa
Menggunakan Teknologi J2ME**

(Studi Kasus Universitas Bina Nusantara)

Hanny Santoso; Felix Oktavianus; Meiwin; Felix

Simulasi Sistem Penyusunan Barang Berbentuk Kotak pada Peti Kemas

Doddy Koeswandi; Budi Maryono; Daniel Haryanto; Mario Theodorus

Biro Penelitian dan Publikasi



LAPORAN
TEKNIS BERKALA
TEKNIK INFORMATIKA
Volume 11 Nomor 4, Des. 2003

Jurnal Ilmiah
Biro Penelitian dan Publikasi
Universitas Bina Nusantara

Pelindung:
Dr. Th. Widia Soerjaningsih

Redaktur Utama:
Prof. Dr. Gerardus Polla, M.App.Sc.

Redaktur Pelaksana:
Sablin Yusuf, M.Comp., M.Sc.
Januar Wahjudi, M.Sc.

Quality Assurance:
Dr. Ir. Haryono S, M.Sc.
Dr. Ir. Edi Abdurachman, M.Sc.
Once Kurniawan, S.Kom., M.M.

Editor:
Dra. Endang Emawati, M.Lib.
Titik Rahayu S, S.S.
Agung Pramanto, S.S.

Sekretariat:
Hery H.M., S.Kom.
Tri Septi Wulansari
Harkat Christian Zamasi

Alamat Redaksi:
Biro Penelitian dan Publikasi
Universitas Bina Nusantara
Jl. Kebon Jeruk Raya No. 27,
Kemanggisan - Jakarta, 11530
Telp. (021) 5345830, 5350660,
ext. 1706, Fax. (021) 5300244

ISSN 0853-6740

Redaksi menerima tulisan yang bersifat ilmiah, berisi ide baru yang asli, teori, konsep, praktik, maupun implementasi teknologi dalam kehidupan sehari-hari.

Tulisan ditik dengan spasi tunggal berisi abstrak, pendahuluan, isi, simpulan, dan daftar pustaka dengan panjang tulisan 10-15 halaman.

Tulisan diserahkan kepada redaksi Laporan Teknis Berkala dalam 2 bentuk, yaitu print-out dan file dalam disket dengan format Microsoft Word minimum versi 6.0.

DAFTAR ISI

Kata Pengantar.....	iii
Januar Wahjudi; Sally Cemerlang; Hadi Gunawan; Hartawan <i>Perangkat Ajar Bahasa Inggris Dasar dan Kosakata Berbasiskan Multimedia.....</i>	283-301
Wiedjaja; Noprianto <i>Authentikasi pada Linux Terminal Server Project Menggunakan Smart Card.....</i>	302-312
Johan M.K.; Ferry Sumendap; Joshua; Lay Iwan Setiawan <i>Analisis dan Perancangan Router Simulator untuk Mensimulasikan IOS pada Cisco Router dalam Mendukung Program Cisco Networking Academy.....</i>	313-330
Mark P. Eliasaputra; Henny Surya Ningsih; Adriani Halim <i>Penerapan Konsep Berorientasi Objek dengan Unified Process dan Unified Modeling Language dalam Perancangan Aplikasi Web Distance Learning..</i>	331-340
Andreas Utomo Kuswara; Edo Budi; Oto Joseph Santoso; Ani <i>Analisis dan Perancangan Sistem Knowledge Object Repository Studi Kasus: UPT Perangkat Lunak Universitas Bina Nusantara....</i>	341-353
Hanny Santoso; Felix Oktavianus; Meiwin; Felix <i>Perancangan Aplikasi Mobile Information Service untuk Mahasiswa Menggunakan Teknologi J2ME: Studi Kasus Universitas Bina Nusantara.....</i>	354-369
Doddy Koeswandi; Budi Mariyono; Daniel Haryanto; Mario Theodorus <i>Simulasi Sistem Penyusunan Barang Berbentuk Kotak pada Peti Kemas.....</i>	370-384

AUTHENTIKASI PADA *LINUX TERMINAL SERVER PROJECT* MENGGUNAKAN *SMART CARD*

Wiedjaja; Noprianto

ABSTRAK

Penelitian menjelaskan perancangan autentikasi pada Linux Terminal Server Project menggunakan smart card. Bahasan meliputi penerapan latar belakang, ruang lingkup, tujuan, metodologi penelitian, dan tinjauan pustaka yang mendasari artikel ini. Kemudian dilakukan perancangan, implementasi, dan evaluasi sistem autentikasi tersebut. Dengan sistem ini penggunaan komputer hanya perlu memasukkan dan mencabut smart card untuk mengakses sistem komputer.

Kata kunci: autentikasi, Linux Terminal Server Project (LTSP), smart card

PENDAHULUAN

Latar belakang

Saat ini, teknologi komputer dan infrastrukturnya telah begitu berkembang. Tidak dapat dipungkiri lagi, perkembangan tersebut memudahkan kehidupan manusia sampai ke tingkat yang setinggi-tingginya dan membantu meningkatkan produktifitas. Batasan dunia menjadi sangat sempit. Hal yang dulunya tidak mungkin, menjadi sangat mungkin pada saat ini. Harga komputer dan segenap komponennya pun menjadi semakin murah. Di sisi lain, mengambil contoh bangsa Indonesia, tidak semua orang dan tidak semua daerah, dapat menikmati perkembangan tersebut. Tidak semua orang dapat memiliki komputer. Pengertian harga murah pun menjadi bergeser. Di sini, terjadi *digital divide* yang sungguh besar. Makin besar, apabila setiap lembaga atau perorangan yang menggunakan komputer harus selalu membeli unit komputer baru dan lengkap hanya untuk mengerjakan hal sederhana.

Kantor, lembaga kursus, universitas, atau bahkan sekolah, pada umumnya memiliki komputer lengkap di meja kerja atau laboratorium komputer. Setiap kegiatan komputasi yang dilakukan akan menggunakan informasi yang tersimpan di dalam media penyimpanan tetap (fixed storage device). Setiap komputer memiliki sistem operasi sendiri, data disimpan di komputer tersebut (kalaupun menggunakan jaringan, tetap ada data yang disimpan di fixed storage device lokal), dan mandiri. Pada kenyataannya, terutama dari sisi pandang tepat guna, kegiatan komputasi tidak harus selalu menggunakan cara tersebut. Dengan terminal server sistem operasi GNU/Linux yang diterapkan dalam Linux Terminal Server Project (LTSP), kegiatan komputasi menjadi (sangat) murah, andal, dan terintegrasi. Proses autentikasi yang dilakukan pada umumnya,

inal Server Project
ng lingkup, tujuan,
emudian dilakukan
Dengan sistem ini
1 untuk mengkses

nbang. Tidak dapat
pai ke tingkat yang
ria menjadi sangat
ni. Harga komputer
nbil contoh bangsa
embangan tersebut.
enjadi bergeser. Di
tiap lembaga atau
r baru dan lengkap

umunya memiliki
an komputasi yang
npanan tetap (fixed
impan di komputer
ixed storage device
kegiatan komputasi
operasi GNU/Linux
komputasi menjadi
in pada umumnya,

ber 2003: 302-312

bahkan pada LTSP sendiri, dilakukan dengan metode konvensional, misalnya mengetikkan pasangan username dan password. Dengan perkembangan teknologi smart card, metode autentikasi dapat dilakukan dengan smart card. Pengguna hanya perlu memasukkan smart card untuk memasuki sistem komputer dan mencabut smart card tersebut apabila ingin keluar dari sistem komputer tersebut.

Menyeberang ke sisi lain perkembangan teknologi, muncullah *smart card* ke permukaan. Segala hal yang berhubungan dengan kartu, dapat dibuat lebih sederhana dan kompak ketika menggunakan *smart card*. Salah satu kegunaan *smart card* adalah penerapannya dalam proses autentikasi. Untuk mengantisipasi permasalahan *digital divide* dan kegiatan komputasi mahal (untuk pekerjaan sederhana), dapat digunakan sistem *terminal server* yang diterapkan menggunakan sistem operasi GNU/Linux. Untuk perangkat lunak, tidak ada sepeser pun uang yang harus dibayarkan kepada perusahaan tertentu. Satu langkah penghematan besar telah dilakukan. Untuk perangkat keras, terminal tidak perlu menggunakan komputer canggih karena pada umumnya, kegiatan komputasi dilakukan di *server*, walaupun memungkinkan jika *client* juga bekerja. Langkah penghematan, termasuk menggunakan perangkat keras lama yang telah disimpan di gudang, telah dilakukan. Dua langkah tersebut akan menekan pengeluaran. Dengan demikian, apabila kegiatan komputasi masal tidak terlalu mahal, diharapkan *digital divide* dapat diperkecil. Salah satu proyek yang didedikasikan untuk itu adalah *Linux Terminal Server Project* (LTSP). Di sisi lain, murah tidak harus selalu tidak bagus. *Smart card* dapat diterapkan, sebagai contoh, digunakan untuk proses autentikasi.

Penelitian ini menerapkan penggunaan *smart card* pada proses autentikasi untuk *Linux Terminal Server Project* (LTSP) sehingga pada akhirnya, kegiatan komputasi dapat menggunakan *Linux Terminal Server Project* yang proses autentikasinya dilakukan dengan bantuan *smart card*. Seorang pengguna yang memiliki hak akses hanya menancapkan sebuah *smart card* ke suatu terminal kemudian proses autentikasi akan dilakukan dan terminal tersebut siap dipakai. Jika sesi penggunaan telah selesai, *smart card* tersebut tinggal dicabut dan terminal tersebut siap digunakan kembali untuk sesi berikutnya.

Ruang lingkup

Bahasan hanya pada proses autentikasi pada *Linux Terminal Server Project* menggunakan *smart card* jenis *memory card*. Pada akhirnya, kegiatan komputasi dapat menggunakan *Linux Terminal Server Project* yang proses autentikasinya dilakukan dengan bantuan *smart card*. Segala sistem, dari sistem *terminal server*, pemrograman, sampai penggunaan *smart card*, dijalankan pada sistem operasi GNU/Linux dan perangkat lunak bebas lainnya.

Tujuan dan Manfaat

Tujuan bahasan adalah menerapkan penggunaan *smart card* pada proses autentikasi dalam *Linux Terminal Server Project* (LTSP).

Manfaat penelitian sebagai berikut.

1. Sebagai salah satu alternatif untuk menggantikan sistem komputasi mahal (untuk pekerjaan sederhana) dan tidak tepat guna dengan menerapkan komputasi hemat, tepat guna, dan modern dengan *Linux Terminal Server Project* yang dilengkapi dengan *smart card*.

2. Sebagai salah satu alternatif untuk penggunaan sistem komputasi yang sepenuhnya dijalankan menggunakan perangkat lunak bebas.

Metode Penelitian

Penelitian ini menggunakan metode perancangan yang dilakukan dalam berbagai tahap berikut.

1. Pembuatan bagan alir.
2. Perancangan program yang berjalan di *client*.
3. Perancangan program yang berjalan di *server*.
4. Pengimplementasian dan pengevaluasian sistem.

TINJAUAN PUSTAKA

GNU/Linux

GNU/Linux adalah sistem operasi yang digunakan dalam penelitian ini. Nama GNU dimasukkan dan harus digunakan karena sistem operasi Linux menggunakan perangkat lunak GNU.

Terminal server

Definisi umum *terminal server* adalah suatu perangkat keras atau *server* yang menyediakan layanan bagi terminal (dapat berupa komputer, *printer*, dan alat lainnya) dengan koneksi pada jaringan lokal ataupun jaringan yang lebih luas. Dalam skala yang lebih sempit, suatu terminal dapat berupa komputer. Dalam hal ini, komputer tersebut hanyalah berupa sarana untuk berkomunikasi dengan pengguna. Segala kegiatan komputasi dan penyimpanan secara umum dilakukan pada *server*.

Linux Terminal Server Project

Salah satu penerapan sistem komputasi *client* tanpa media penyimpanan tetap (*hard disk*) adalah *Linux Terminal Server Project (LTSP)*. Proyek yang memiliki *website* di <http://www.ltsp.org> itu dimulai oleh James A. McQuillan dan dikembangkan oleh para pemrogram di seluruh dunia. LTSP menyediakan cara untuk memanfaatkan komputer yang tidak terpakai oleh perkembangan komputasi saat ini menjadi terminal yang dapat menjalankan aplikasi berbasis teks ataupun grafikal dengan *server* yang berbasiskan sistem operasi GNU/Linux.

Etherboot

Etherboot adalah paket perangkat lunak yang berguna untuk membuat image ROM (*Read Only Memory*) yang mampu untuk mengambil kode dari jaringan ethernet untuk dieksekusi di komputer dengan arsitektur x86. *Etherboot* sendiri dapat diartikan sebagai kode untuk dimasukkan ke dalam ROM yang dimiliki oleh kartu jaringan. Tidak hanya itu, *Etherboot* juga dapat

dimasukkan ke dalam *floppy* dan *hard disk*, terutama bagi yang ingin membuat percobaan terlebih dahulu atau bagi yang tidak memiliki kartu jaringan dengan dukungan *boot ROM*.

DHCP

Dalam bekerja dengan LTSP, akan menghadapi komputer tanpa media penyimpanan tetap, seperti *hard disk* dan tanpa sistem operasi. Tanpa sistem operasi, secara otomatis membuat komputer tersebut tidak memiliki konfigurasi jaringan, seperti alamat *Internet Protocol* (IP) ketika bekerja dengan protokol TCP/IP. Untuk itu, komputer tersebut memerlukan sebuah *server* yang akan memberikan parameter konfigurasi. Semua hal tersebut dapat terjadi karena terdapatnya *DHCP Server*.

TFTP

TFTP merupakan singkatan dari *Trivial File Transfer Protocol*. TFTP didefinisikan di dalam *Request For Comments* (RFC) 1350 (yang memperbaharui RFC 783). TFTP adalah protokol sederhana yang berguna untuk melakukan *transfer file*. Protokol itu diimplementasikan menggunakan *User Datagram Protokol* (UDP). Fokus utama TFTP adalah membaca *file* dari *server* atau menulis *file* ke *server*. Karena diimplementasikan menggunakan UDP, TFTP mewarisi sifat UDP tersebut.

Network File System (NFS)

NFS adalah singkatan dari *Network File System*, yang pertama kali dikembangkan oleh Sun Microsystems pada tahun 1980 sebagai sarana untuk membuat *file* sistem bagi sistem komputasi *client* tanpa media penyimpanan tetap. NFS menyediakan akses jarak jauh untuk membagi *file* sistem pada jaringan. Hal itu menyebabkan sebuah *client* yang tidak memiliki media penyimpanan tetap dapat mengakses *file* sistem pada *server* yang telah di bagi lewat NFS.

Window System

Secara singkat, sesuai dengan manual penggunaannya, X(1), *X Window System* adalah sistem window yang portabel dan transparan terhadap jaringan. Perbedaan utamanya dengan sistem window lainnya, seperti sistem window Microsoft dalam sistem operasi Microsoft Windows dan sistem window Apple dalam sistem operasi MacOS, adalah *X Window System* didesain dalam konsep kemampuan penggunaan pada jaringan lokal (LAN). Program dapat dijalankan pada satu komputer dan ditampilkan pada komputer lainnya.

Smart card

Smart card merupakan salah satu tipe *chip card*. *Smart card* sendiri adalah sebuah kartu plastik yang memiliki sebuah *chip* komputer yang menyimpan dan melakukan transaksi data penggunaannya. Data diasosiasikan pada nilai dan informasi atau keduanya yang disimpan dan diproses dalam *chip* tersebut, baik pada kartu memori maupun kartu mikroprosesor. Transaksi pada *smart card* dilakukan melalui *smart card reader*.

Sistem yang dilengkapi dengan penerapan *smart card* kini semakin populer. Sistem tersebut termasuk kesehatan, perbankan, *entertainment*, dan transportasi. Sistem dapat

memanfaatkan fitur dan keamanan yang datang bersama *smart card*. Menurut Dataquest, pasar *smart card* akan meningkat sebesar 4,7 miliar unit dan 6,8 miliar dollar Amerika Serikat pada tahun 2002.

Smart card digunakan karena penggunaan *smart card* secara umum meningkatkan kenyamanan dan keamanan dalam berbagai transaksi. *Smart card* menyediakan permukaan yang tahan terhadap pemakaian normal (lebih detail di ISO 7816) dan identitas per pengguna. *Smart card* juga menyediakan sistem keamanan untuk transaksi. Sistem keamanannya meliputi kecerobohan pengguna sampai sistem perusakan tingkat tinggi. Seperti yang diterapkan dalam penelitian ini, *smart card* juga dapat digunakan dalam jaringan.

PERANCANGAN SISTEM

Analisis Permasalahan

Linux Terminal Server Project adalah suatu proyek sistem terminal *server* yang berjalan di sistem operasi GNU/Linux. LTSP sendiri dapat mengoptimalkan penggunaan yang sudah tidak terpakai lagi. Komputer yang selama ini telah disimpan di gudang dapat didayakan kembali. LTSP pun dapat mendayakan komputer yang sudah relatif tua. Tidak ada sepeser pun uang yang harus dibayarkan kepada pihak tertentu. Dengan demikian, dapat dikatakan LTSP merupakan solusi tepat guna untuk komputasi.

Di lain sisi perkembangan teknologi, *smart card* hadir sebagai solusi kartu yang kompak dan andal. Kartu konvensional dengan berbagai fungsi dapat disatukan dengan satu buah *smart card*. Permasalahannya, LTSP sendiri tidaklah dirancang untuk bekerja dengan *smart card*. Hal itu merupakan sesuatu yang cukup disayangkan. Apabila suatu perusahaan menerapkan penggunaan *smart card* secara meluas maka ada baiknya apabila sistem komputasi mereka pun dilengkapi dengan *smart card*. Untuk itulah penelitian ini dibuat agar LTSP dapat bekerja dengan *smart card* dan menjadi suatu solusi komputasi yang murah, tepat guna, dan modern.

Cara Kerja Sistem

Pertama kali komputer *client* melakukan proses *boot*, langkah proses *boot* LTSP akan dikerjakan. Asumsikan *client* LTSP menjalankan dalam *runlevel* 5 kemudian *script* /tmp/start_ws akan dijalankan. Pada sistem LTSP yang normal tanpa *smart card*, *script* tersebut akan menjalankan X *server* (*server* X) dan melakukan *query* ke *server* LTSP. Akan tetapi, modifikasi pada *script* rc.local LTSP membuat *script* rc.local tersebut membuat *script* /tmp/start_ws yang akan menjalankan program *smartcardLTSP*.

Program *smartcardLTSP* tersebut akan mengatur nilai *default* untuk beberapa variabel tertentu yang dapat diatur sendiri oleh *user*. Pengaturan nilai *default* itu dilakukan sebagai tindakan antisipatif, apabila *file* konfigurasi tidak terdapat di sistem. Nilai *default* tersebut dapat digantikan dengan nilai yang tersimpan di dalam *file* konfigurasi, pada saat pembacaan *file* konfigurasi dilakukan. Lalu, inisialisasi jaringan akan dilakukan dengan membuka koneksi UDP pada port 2728.

ut Dataquest, pasar
erika Serikat pada

um meningkatkan
an permukaan yang
er pengguna. *Smart*
nanannya meliputi
3 diterapkan dalam

server yang berjalan
ian yang sudah tidak
akan kembali. LTSP
in uang yang harus
P merupakan solusi

kartu yang kompak
gan satu buah *smart*
1 *smart card*. Hal itu
erapkan penggunaan
reka pun dilengkapi
a dengan *smart card*

ses boot LTSP akan
script /tmp/start_ws
ript tersebut akan
an tetapi, modifikasi
/tmp/start_ws yang

ik beberapa variabel
kan sebagai tindakan
:but dapat digantikan
:aan *file* konfigurasi
eksi UDP pada port

Langkah berikutnya adalah melakukan perulangan terus menerus sampai program diinterupsi. Di dalam perulangan tersebut, sistem mengenal dua macam status: *LOGIN* dan *LOGOUT*. Status *LOGIN* dimaksudkan sedang berada di dalam sesi sistem. Status *LOGOUT* dimaksudkan sedang tidak berada di dalam sesi sistem. Selanjutnya, masih berada di dalam perulangan terus menerus, proses pengecekan keberadaan *smart card* di dalam *smart card reader* akan dilakukan. Jika terdapat *smart card* di dalam *smart card reader* maka pengecekan status akan dilakukan. Apabila terdapat *smart card* di dalam *smart card reader* dan status saat itu adalah *LOGIN* maka untuk sementara, tidak ada aksi yang dilakukan mengingat itu adalah kondisi normal: sedang berada di sesi sistem (sedang *LOGIN*) dan *smart card* masih terpasang. Untuk pengembangan di masa depan, pengecekan sekali lagi akan dilakukan untuk menghindari tindakan kejahatan komputer dengan memasukkan *smart card* lain. Apabila terdapat *smart card* di dalam *smart card reader* dan status saat itu adalah *LOGOUT* maka anak proses baru akan dibuat untuk menjalankan *X server (server X)* dan melakukan query ke *server LTSP*.

Skenario yang seharusnya terjadi adalah pada saat *X server (server X)* dinyalakan maka seharusnya, program *smartcardLTSPd* telah menunggu di *server*. Program itu siap berkomunikasi dengan program *smartcardLTSP* dan menunggu pesan setiap saat. Untuk memastikan koneksi, program *smartcardLTSP* di *client* akan terus menerus mencoba membuat koneksi dan mengirimkan pesan *CLIENT_CONNECT* ke program *smartcardLTSPd* di *server* sampai menerima kembali pesan *SERVER_CONNECT* dari *smartcardLTSPd*. Apabila dalam rentang waktu tertentu, program *smartcardLTSPd* tidak memberikan respons maka pengulangan akan dilakukan dengan mengasumsikan tidak terjadi apa-apa sebelumnya. Apabila pesan *SERVER_CONNECT* telah diterima dan diyakini *smartcardLTSPd* telah berjalan maka pesan *CLIENT_READY* akan dikirimkan ke program *smartcardLTSPd* yang berjalan di *server LTSP*.

Masih dalam perulangan, jika proses pengecekan keberadaan *smart card* di dalam *smart card reader* mendapatkan hasil negatif, *smart card* tidak dimasukkan ke dalam *smart card reader*, maka pengecekan status akan dilakukan kembali. Jika status saat itu adalah *LOGOUT* maka untuk sementara, tidak ada aksi yang dilakukan, mengingat itu adalah kondisi normal: tidak berada di sesi sistem tetapi *smart card* juga tidak dimasukkan. Apabila *smart card* tidak dimasukkan ke dalam *smart card reader* dan status saat itu adalah *LOGIN* maka untuk anak proses yang menjalankan *X server (server X)* akan diterminasi. Dengan demikian, segala tampilan dan koneksi ke *server* akan terputus. Untuk pengembangan di masa depan, rutin *LOGOUT* (keluar dari sistem) akan dilakukan dengan cara yang lebih prosedural untuk setiap desktop tertentu.

Masih di dalam perulangan terus menerus, apabila diyakini *smartcardLTSPd* telah berjalan, program *smartcardLTSP* akan mencoba mendapatkan pesan yang dikirimkan oleh program *smartcardLTSPd*. Pesan tersebut akan dikirimkan ke modul *smartcardLTSP_parse_msg()* yang akan mengartikan pesan tersebut dan mengambil tindakan yang bersesuaian. Penundaan selama waktu tertentu yang dapat diatur oleh pengguna (nilai *default* adalah 1 detik) akan dilakukan dalam perulangan terus menerus dalam program *smartcardLTSP*.

Dari sisi program *smartcardLTSPd*, pada saat dijalankan, program akan membuka koneksi pada port 2728. Pada saat itu juga, percobaan koneksi dari program *smartcardLTSP* yang berjalan di *client* seharusnya berhasil dan program *smartcardLTSPd* akan menerima pesan *CLIENT_CONNECT* yang akan dibalas dengan pesan *SERVER_CONNECT*. Dalam program, *smartcardLTSPd* akan menunggu pesan, mengartikannya, dan melakukan aksi yang bersesuaian. Penelitian ini menghasilkan dua program, yaitu *smartcardLTSP* yang berjalan di sisi *client* dan *smartcardLTSPd* yang berjalan di sisi *server*.

IMPLEMENTASI DAN EVALUASI

Implementasi

Dalam penelitian ini, dibahas penggunaan hasil penelitian dari kaca mata pandang pengguna akhir. Dengan demikian, seorang pengguna dapat memasukkan *smart card* untuk melakukan proses autentikasi dirinya dan dapat mencabut *smart card* untuk keluar dari sesi sistem.

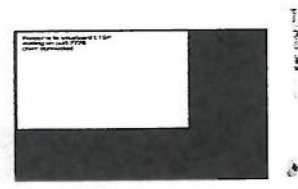
Langkah Autentikasi

1. Komputer *client* dinyalakan dan *boot device* diarahkan kepada *floppy* dengan asumsi *floppy* adalah *boot device*. Pengaturan *boot device* dapat dilakukan di BIOS. Untuk memasuki pengaturan BIOS, umumnya dilakukan penekanan tombol DEL pada saat komputer dinyalakan.
2. Komputer *client* akan meminta informasi dari *server*.
3. Setelah semua informasi diterima, program *smartcardLTSP* akan berjalan dan mengambil kontrol atas komputer *client*. Pengguna akan melihat tampilan berikut ini, apabila *smart card* tidak dimasukkan selama jumlah waktu tertentu.



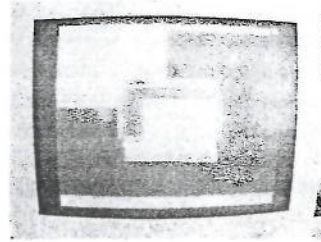
Gambar 1 Program *SmartcardLTSP* tanpa *Smart card*

4. *Smart card* dimasukkan untuk melakukan autentikasi ke *server*. *X server (server X)* akan dinyalakan dan sekaligus meminta informasi ke *server*. Program *smartcardLTSPd* telah menunggu dan pengguna akan melihat tampilan berikut ini.



Gambar 2 Program *SmartcardLTSPd*

5. Apabila autentikasi gagal maka pesan kesalahan akan ditampilkan. Apabila autentikasi berhasil, pengguna akan memasuki sesi sistem, seperti ditunjukkan gambar berikut ini.



Gambar 3 Autentikasi Berhasil dan Memasuki Sesi Sistem

6. Untuk keluar dari sesi sistem, *smart card* dicabut dari *smart card reader*. Proses yang menjalankan *X server (server X)* akan diterminasi. Pengguna kembali akan melihat program *smartcardLTSP*.

Evaluasi

Penelitian ini menerapkan penggunaan *smart card* untuk proses autentikasi pada *Linux Terminal Server Project*. Evaluasi terutama dilakukan pada penerapan *smart card* untuk proses autentikasi pada LTSP.

Evaluasi Kemudahan Proses Autentikasi

Kemudahan pada saat autentikasi adalah salah satu kegunaan penelitian ini. Apabila pada sistem LTSP *default*, untuk melakukan autentikasi, pengguna sistem harus mengetikkan nama dan *password*. Penerapan penelitian ini menjadikan seorang pengguna cukup memasukkan *smart card* miliknya ke dalam *smart card reader* dan proses autentikasi akan dilakukan. Apabila proses autentikasi berhasil maka pengguna akan memasuki sesi sistem. Dari percobaan, waktu yang diperlukan untuk proses autentikasi sampai memasuki sesi sistem adalah satu sampai lima detik. Dengan cara konvensional, dengan mengetikkan pasangan *username* dan *password* dari percobaan, didapat minimal waktu adalah tiga sampai sepuluh detik. Hal itu tentu saja paling tidak bergantung kepada kecepatan mengetik dan panjang teks yang harus dimasukkan. Namun, dengan hanya perlu satu kartu untuk masuk ke sistem, keamanan dapat menjadi rentan. Terutama karena penelitian ini menggunakan *smart card* jenis *memory card* yang tidak memiliki proteksi ganda. Dalam beberapa hal, misalnya ketika *smart card* hilang maka yang mendapatkan kartu tersebut dapat segera memasuki sistem dengan mulus.

Evaluasi Penggunaan Smart Card

Evaluasi dilakukan dengan mengukur waktu yang diperlukan oleh seorang pengguna untuk memasuki dan keluar dari sesi sistem. Rata-rata waktu yang diperlukan dari menancapkan *smart card* ke dalam *smart card reader* sampai memasuki sesi sistem adalah 3-7 detik. Waktu yang diperlukan dari mencabut *smart card* sampai kembali ke tampilan program *smartcardLTSP* di *client* adalah 1-3 detik. Akan tetapi, penggunaan *smart card* disini tidak memberikan fasilitas

cacamata pandang
smart card untuk
ik keluar dari sesi

ngan asumsi *floppy*
Untuk memasuki
da saat komputer

an dan mengambil
apabila *smart card*

er (*server X*) akan
rtcardLTSPd telah

kepada pengguna untuk memasukkan kembali *smart card* apabila dicabut dalam jangka waktu tertentu. Hal itu diperlukan pada lingkungan kerja produktif dan pengguna dapat mencabut *smart card* kapan saja dan memasukkannya kembali kapan saja dan di mana saja, dan pekerjaan pengguna tersebut akan tetap terbuka. Dalam penelitian ini, begitu *smart card* dicabut maka X server (*server X*) dan semua aplikasi yang berjalan di bawahnya akan diterminasi. Sebagai konsekuensi penggunaan *smart card*, setiap *client* harus memiliki *smart card reader* yang untuk saat ini dapat dibeli dengan harga kurang lebih 30 Dollar AS.

PENUTUP

Simpulan

1. Program *smartcardLTSP* dan *smartcardLTSPd* yang bertanggung jawab untuk melakukan autentikasi pada LTSP menggunakan *smart card*, telah dapat digunakan pada lingkungan pengembangan dan belum teruji dapat diterapkan pada lingkungan produktif. Hal itu disebabkan minimal tiga hal berikut.
 - a. Belum diterapkan pada lingkungan kerja nyata. Sebelum itu diterapkan, status kedua program tersebut adalah *unstable/testing*.
 - b. Tidak memberikan kesempatan kepada pengguna untuk memasukkan *smart card* kembali ketika *smart card* dicabut untuk beberapa waktu tertentu. Setiap pencabutan *smart card* akan berakibat anak proses yang menjalankan X server (*server X*) diterminasi. Akibatnya, aplikasi yang sedang terbuka akan tertutup sehingga begitu pengguna memasukkan *smart card* lagi, sebuah sesi baru akan dibuka dari awal. Metode membuat anak proses dan memterminasinya untuk saat ini diyakini bukan merupakan metode yang optimal dan berisiko kecil.
 - c. Metode autentikasi yang digunakan dengan membaca langsung *file /etc/passwd* dan */etc/shadow* tanpa perantara yang aman, seperti pustaka PAM, dapat menyebabkan celah keamanan yang tidak dapat diprediksi.
2. Program *smartcardLTSP* dan *smartcardLTSPd* membaca *file* konfigurasi yang akan membuat penelitian ini tampil lebih personal dengan mengubah beberapa parameter.

Saran

Penelitian ini memerlukan banyak saran dan pengembangan untuk peningkatan kinerja dan pengurangan risiko agar dapat diuji dan digunakan di lingkungan kerja produktif. Berikut ini adalah saran tersebut.

1. Penggunaan pustaka umum untuk mengakses *smart card* di sistem operasi GNU/Linux. Pengembangan besar-besaran telah dilakukan dan telah berada dalam tahap stabil ketika penelitian ini selesai dikerjakan. Untuk pengembangan selanjutnya, pustaka tersebut akan dipakai agar dapat mendukung berbagai *smart card reader* yang beredar di pasaran.
2. Penggunaan pustaka PAM untuk melakukan autentikasi dalam mekanisme yang lebih aman.
3. Penanganan koneksi antara program *client* dan *server* serta mekanisme untuk memasuki sesi sistem dan mekanisme untuk keluar dari sesi sistem.

lam jangka waktu
at mencabut *smart*
ja, dan pekerjaan
d dicabut maka X
erminasi. Sebagai
reader yang untuk

untuk melakukan
n pada lingkungan
produktif. Hal itu

pkan, status kedua

smart card kembali
cabutan *smart card*
r X) diterminasi.
begitu pengguna
Metode membuat
pakan metode yang

le /etc/passwd dan
menyebabkan celah

yang akan membuat

peningkatan kinerja
produktif. Berikut ini

operasi GNU/Linux.
tahap stabil ketika
staka tersebut akan
di pasaran.
ne yang lebih aman.
ntuk memasuki sesi

ber 2003: 302-312

4. Untuk pengembangan selanjutnya, pengguna sistem harus diberikan kesempatan untuk memasukkan kembali *smart card* ketika *smart card* dicabut, tentunya dalam selang waktu tertentu.
5. Pengguna seharusnya dapat memasuki sesi sistem dalam selang waktu yang lebih singkat. Untuk saat ini, pengguna dihadapkan pada eksekusi program DESKTOP. Untuk pengembangan di masa depan, diharapkan waktu eksekusi dapat dipersingkat.
6. Pengguna seharusnya dapat mencabut *smart card* di satu terminal kemudian memasukkan kembali *smart card*nya di terminal lain dan segala pekerjaannya akan dibuka kembali dalam waktu kurang dari 5 detik. Hal itu belum dapat dilakukan pada saat ini.
7. Penerapan *smart card* jenis *CPU card* sangat dianjurkan. Hal itu karena masalah keamanan. Pada *smart card* jenis *CPU card*, pembacaan, dan penulisan data dilakukan dalam protokol yang aman, dilengkapi dengan enkripsi. *CPU card* dapat diartikan sebagai komputer kecil di dalam *smart card*. Untuk membaca isi suatu *CPU card* misalnya, diberikan serangkaian perintah dan harus melalui proses enkripsi.
8. Penggunaan *biometric smart card* untuk meningkatkan keamanan karena sistem biometrik menjadikan *smart card* hanya dapat digunakan oleh orang dengan nilai biometrik tertentu sehingga seseorang pengguna tidak dapat menggunakan *smart card* yang bukan menjadi milik pengguna tersebut.

DAFTAR PUSTAKA

- Droms, Richard. 1997. "*Dynamic Host Configuration Protocol*."
<http://www.ietf.org/rfc/rfc2131.txt>
- Jacquinet Consulting, Inc. 2003. "*The ISO 7816 Smart Card Standard: Overview*."
http://www.cardwerk.com/smartcards/smartcard_standard_ISO7816.aspx
- McQuillan, James A. 2002. "*Linux Terminal Server Project*."
<http://www.ltsp.org/documentation/ltsp-3.0-4-en.html>
- Noprianto. 2003. *Python dan Pemrograman Linux*. Yogyakarta: Andi Offset.
- Smartcardbasics.com. 1999. "*Smart Cards and Security Overview*."
<http://www.smartcardbasics.com/index.html>
- Sollins, Karen R. 1992. "*The TFTP Protocol (Revision 2)*." <http://www.ietf.org/rfc/rfc1350.txt>
- TechTarget. 2002. "*Terminal server*."
http://whatis.techtarget.com/definition/0,,sid9_gci213123,00.html
- TLDP.org. 2003. "*The Linux Documentation Project*."
<http://www.tldp.org/docs.html>

The Trustees of Indiana University. 1996. "*Network File System.*"
<http://www.uwsg.iu.edu/usail/network/nfs/overview.html>

Wall, Kurt. 1999. *Linux Programming by Example*. Que, Indianapolis.

X.org. 2003. "*Introduction: About The X Window System.*"
<http://www.x.org/X11.html>

Yap, Ken and Markus Gutschke. 2002. "*Etherboot User Manual.*"
<http://Etherboot.sourceforge.net/doc/html/userman.html>