IMPLEMENTASI LTSP *(LINUX TERMINAL SERVER)* SEBAGAI SOLUSI OPTIMALISASI PENYELENGGARAAN INFRASTRUKTUR LABORATORIUM KOMPUTER DI PERGURUAN TINGGI (STUDI KASUS: LABORATORIUM TUK (TEMPAT UJI KOMPETENSI), TEKNIK INFORMATIKA, STT TELKOM)

Dodi Wisaksono Sudiharto
Jurusan Teknik Informatika, Sekolah Tinggi Teknologi Telkom, Bandung
e-mail: dws@stttelkom.ac.id

ABSTRAKSI

Linux Terminal Server Project (LTSP) adalah suatu proyek yang mengeksplorasi kemampuan Linux untuk aplikasi diskless Xterminal di sisi klien. Teknologi yang mirip sistem mainframe ini semakin populer karena dapat menghemat sumber daya perangkat keras tanpa perlu mengurangi performansi. Seperti yang diketahui bahwa penyelenggaraan infrastruktur laboratorium komputer di perguruan tinggi membutuhkan biaya yang tidak sedikit. Masalah dapat timbul mengingat perkembangan teknologi yang demikian cepat namun belum tentu mampu diadopsi oleh lingkungan perguruan tinggi yang terkadang terbentur dengan faktor-faktor seperti pembiayaan penyelenggaraan infrastruktur maupun kurangnya optimalisasi laboratorium seperti dalam penggunaan perangkatnya. Dengan LTSP ini diharapkan penyelenggaraan infrastruktur laboratorium komputer di perguruan tinggi dapat semakin dioptimalkan.

***Kata kunci**: LTSP, thin-client, teknologi terminal, performansi.*

1. PENDAHULUAN

1.1 Latar Belakang

Penyelenggaraan infrastruktur seperti laboratorium komputer di perguruan tinggi membutuhkan biaya yang tidak sedikit. Perkembangan teknologi yang ada, seperti sistem operasi dan perangkat lunak juga membuat perlu ditingkatkannya kompatibilitas dari infrastruktur pendukung dalam hal ini perangkat keras dari komputer yang digunakan, sehingga dapat mendukung implementasi dari sistem operasi dan perangkat lunak yang ada. Masalah dapat timbul mengingat perkembangan teknologi yang demikian cepat namun belum tentu mampu diadopsi oleh lingkungan perguruan tinggi yang terkadang terbentur dengan faktor-faktor seperti pembiayaan penyelenggaraan infrastruktur maupun kurangnya optimalisasi laboratorium seperti dalam penggunaan perangkatnya. Dengan LTSP ini diharapkan penyelenggaraan infrastruktur laboratorium komputer di perguruan tinggi dapat semakin dioptimalkan.

1.2 Tujuan

Tujuan dari penulisan makalah ini adalah:

1. Mengetahui arsitektur yang diadopsi oleh LTSP serta memahami prinsip-prinsipnya penggunaannya.
2. Memahami keuntungan dan kerugian dari implementasi LTSP.
3. Memahami bagaimana LTSP diterapkan dalam optimalisasi penggunaan perangkat laboratorium komputer melalui kasus laboratorium TUK (Tempat Uji Kompetensi) di Teknik Informatika, STT Telkom.

1.3 Batasan Masalah

Tidak membahas masalah *capacity processing*, namun lebih pada implementasi LTSP pada laboratorium komputer untuk optimalisasi sumber daya yang ada.

2. LANDASAN TEORI

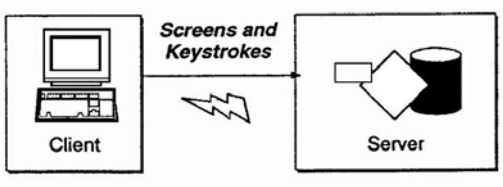
2.1 Arsitektur LTSP

Linux Terminal Server Project (LTSP) yang dalam bahasa pasar sering disebut sebagai teknologi *PC cloning* mengadopsi arsitektur *thin client* di mana sebuah *PC server* yang besar diakses oleh banyak *PC workstation.* Disebut mengadopsi arsitektur *thin-client* karena pada sisi *end-user* sebagai klien hanya berfungsi sebagai terminal saja, meski terminal tersebut dapat berupa PC yang memanfaatkan jaringan komputer yang ada.

Pada lingkungan LTSP, klien di-*boot* menggunakan *BootROM* (umumnya berupa *disket*) yang sudah terpasang pada *kernel Linux.* Setelah *kernel* di-*load* dalam memori, ia mulai bekerja untuk mencari *server* yang memiliki DHCP atau *Boot Protocol* (BOOTP). *Server* yang menangkap permintaan klien memeriksa terlebih dahulu apakah klien tersebut sudah terdaftar sebagai komputer yang boleh masuk atau tidak. Bila klien tersebut sudah terdaftar, maka *server* memberikan *IP Address* kepada klien dan selanjutnya menjalankan *XWindow* dimana prosesnya terjadi di *server* namun hasilnya yang berupa *Graphical User Interface* (GUI) akan tampak pada komputer klien.

PC cloning yang sebenarnya merupakan *cloning* dari sistem operasi dapat dilakukan dengan cara menjalankan profil dari *user* yang terdaftar pada direktori *server* yang kemudian diakses menggunakan teknologi *remote*. Sehingga segala

proses yang berjalan sesungguhnya dijalankan di sisi *server* sedangkan di sisi klien yang disebut juga terminal hanya menerima perubahan *frame* dalam bentuk GUI dan pengiriman *keystroke* ke sisi server menggunakan protokol X11. Untuk lebih jelasnya mengenai arsitektur LTSP tersebut dapat dilihat pada gambar 1.



Gambar 1. Arsitektur LTSP

2.2 Keuntungan Implementasi LTSP

Pada dasarnya LTSP menggunakan *boot image control* yang memudahkan untuk menciptakan hubungan *thin client*. Sehingga dengan menggunakan *boot image* tersebut dapat memenuhi kebutuhan *user* dalam hal konektifitas terhadap aplikasi yang dijalankan di *server*, yang pada akhirnya akan:

- Mengurangi pembiayaan *admin*. LTSP menggunakan arsitektur *thin client*, di mana manajemennya dilakukan pada sisi *server*. Hal ini memudahkan *admin* untuk melakukan pengaturan administrasi. Dengan demikian TCO dari *admin* menjadi semakin tinggi
- Mempermudah pengamanan sistem. LTSP menggunakan arsitektur *thin client*, sehingga untuk pengamanan dengan sisi paling ekstrim, dapat didesain sehingga aplikasi dibuat hanya berada di sisi *server* dan tidak ada pada sisi klien. Sehingga dapat dilakukan pengamanan yang terpusat di sisi *server*.
- Pembiayaan perangkat keras yang hemat. LTSP menggunakan arsitektur *thin client* di mana pembiayaan perangkat keras secara garis besar menjadi lebih hemat, akibat dari sisi klien hanya menggunakan kapasitas memori dan prosesor yang sedikit dan juga dapat bersifat *diskless* dalam arti tidak menggunakan sumber daya *harddisk* secara maksimal sehingga meningkatkan nilai TCO dari penggunaan perangkat keras. Dengan menggunakan arsitektur *thin client*, memori dapat dibuat berbagi-pakai. Bila beberapa *user* di sisi klien menjalankan suatu aplikasi yang sama, maka hanya perlu me-*load* aplikasi tersebut pada RAM yang terdapat pada *server* pusat.
- Konsumsi energi yang rendah. LTSP menggunakan arsitektur *thin client* yang mana perangkat keras yang digunakan pada *thin client* hanya mengkonsumsi energi yang rendah. Hal ini tidak hanya mengurangi pembiayaan dari penggunaan energi namun juga dalam hal perawatan perangkat seperti pada kasus tidak perlu meningkatkan kebutuhan dari sistem *air-conditioning* yang sudah ada.
- Efesiensi penggunaan *bandwidth*. LTSP yang merupakan teknologi *terminal server* hanya akan mentransfer pergerakan *mouse*, *keystroke* dan *screen update*. Sehingga dapat dilakukan efesiensi dari penggunaan *bandwidth*.
- Penekanan pembiayaan lisensi *software*. Biaya lisensi *software* dapat ditekan karena semuanya dilakukan secara terpusat, sehingga tidak perlu harus mengimplementasikan *software* atau aplikasi yang digunakan di masing-masing klien.

2.3 Kerugian Implementasi LTSP

Adapun kerugian dari penggunaan LTSP terkait dengan arsitekturnya yang mirip dengan arsitektur *thin client* adalah sbb:

- Spesifikasi *server* yang tinggi. Karena semua aplikasi ditangani secara terpusat di *server*, maka spesifikasi perangkat *server* yang digunakan harus cukup tinggi. Namun kemampuan *server* akan selalu dapat ditingkatkan karena LTSP juga dapat menggunakan fitur pemrosesan yang dilakukan oleh beberapa prosesor secara bersama seperti dengan menggunakan *cluster*, sedangkan dari sisi manajemen jaringan dapat dilakukan segmentasi jaringan dengan menggunakan beberapa *LTSP server*, melakukan *load balancing* dan dari sisi penyimpanan data dapat dikombinasikan dengan LDAP.
- Kurang fleksibel. Karena segala manajemen dilakukan terpusat seolah-olah tidak memberikan fleksibilitas yang cukup pada *user*. Konfigurasi *terminal server* agak sulit jika digunakan bagi *user* yang sangat aktif atau sering disebut *power user*. Misalnya untuk *editing video* atau hal-hal yang membutuhkan *user resource* yang cukup banyak.

3. SOLUSI IMPLEMENTASI TERHADAP KASUS

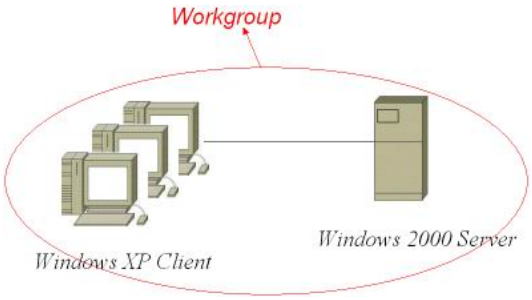
3.1 Gambaran Kasus dan Hipotesa

Seperti yang telah dipaparkan sebelumnya, seringkali peyelenggaraan infrastruktur laboratorium komputer di perguruan tinggi terbentur pada masalah pembiayaan dan optimalisasi penggunaan perangkatnya yang harus selalu menyesuaikan perkembangan teknologi yang diadopsi seperti sistem operasi atau perangkat lunak yang digunakan. Hal ini disebabkan penggunaan arsitektur yang umumnya mengharuskan kompatibilitas semua perangkat yang digunakan. Seperti arsitrektur *client-server* yang paling umum digunakan di laboratorium komputer, mensyaratkan kompatibilitas perangkat keras komputer *client*-nya agar bisa mengadopsi teknologi sistem operasi dan perangkat lunak yang digunakan. Padahal dari sisi kompatibilitas perangkat keras komputer klien tersebut sudah harus

di-*upgrade* atau malah harus diganti. Hipotesa dari solusi adalah dengan menerapkan arsitektur yang berbeda, dalam hal ini arsitektur *thin-client*, sehingga tidak perlu semua perangkat komputer yang ada di laboratorium harus di-*upgrade* atau diganti. Hanya perangkat di sisi *server* saja yang disesuaikan agar dapat mengadopsi teknologi sistem operasi dan perangkat lunaknya dalam arti memenuhi kompatibilitas yang diinginkan.

3.2 Model Jaringan Lama

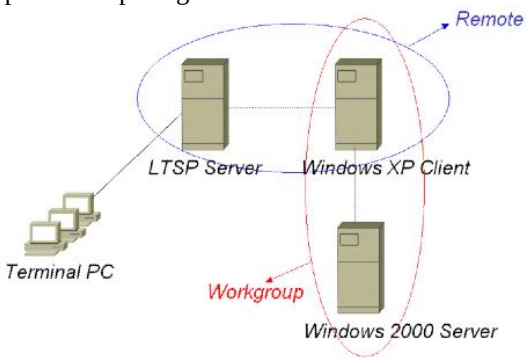
Pada laboratorium komputer yang menjadi studi kasus, diketahui bahwa arsitektur yang digunakan adalah *client-server*. Adapun sistem operasi yang digunakan di sisi *server* adalah Windows 2000 dengan kompatibilitas perangkat *hardware* sekelas PC server. Sedangkan di sisi *client*-nya menggunakan Windows XP. Namun *PC client* hanya beberapa saja yang memenuhi kompatibilitas untuk implementasi sistem operasinya, mengingat masih ada beberapa *PC client* yang memiliki memori di bawah 64 MB. Basisdata yang digunakan adalah *MS Access* yang berada di sisi *server* dan terdapat aplikasi yang harus diinstalkan pada sisi *client*-nya lantaran aplikasi tersebut dibangun dengan *MS Visual Basic*. Untuk lebih jelasnya mengenai model jaringan lama dapat dilihat pada gambar 2.



Gambar 2. Model jaringan lama

3.3 Model Jaringan Baru

Berdasarkan gambaran kasus, hipotesa serta data berupa model jaringan lama, maka untuk model jaringan baru yang akan diimplementasikan dan menjadi solusi adalah dengan menerapkan *LTSP server* sebagai aplikasi *remote server* yang digunakan oleh terminal dalam hal ini *diskless PC client*. Menjadi suatu hal yang menarik adalah terdapat *requirement* di mana lingkungan baru harus bisa membuat administrasi *server* tidak terganggu sama sekali terkait masalah *continuity* dan keamanan, dalam hal ini *PC server* yang menggunakan Windows 2000. Sehingga penulis masih tetap menerapkan satu komputer lagi sebagai klien dengan sistem operasi Windows XP, yang diinstalkan aplikasi yang dibangun dengan *MS Visual Basic* tersebut lantaran tidak dapat diinstalkan pada *LTSP server* mengingat *LTSP server* menggunakan sistem operasi Linux yang tidak kompatibel dengan aplikasi yang dibangun dengan *MS Visual Basic*. Namun bentuk hubungan *client-server* yang digunakan adalah berbentuk *workgroup* dengan jumlah *user account* sebanyak jumlah terminal yang akan digunakan. Sehingga dengan demikian jumlah sistem operasi yang dapat dikloning menggunakan teknologi *remote* dari *LTSP server* adalah sebanyak jumlah terminal yang digunakan. Untuk lebih jelasnya mengenai gambaran model jaringan baru yang digunakan dapat dilihat pada gambar 3.



Gambar 3. Model jaringan baru

4. KESIMPULAN

Berdasarkan pemaparan penulis, terlihat dengan beberapa penyesuaian yang ada, LTSP yang menggunakan arsitektur *thin-client* sangat mendukung untuk dapat dijadikan sebagai salah satu solusi dalam meningkatkan optimalisasi penyelenggaraan infrastruktur laboratorium komputer di perguruan tinggi, dalam hal ini terlihat pada implementasinya di Laboratorium TUK (Tempat Uji Kompetensi). Sehingga diharapkan dapat meningkatkan daya saing bangsa melalui pemberdayaan sistem & teknologi informasi.

PUSTAKA

[1] Aji, Kresno dan Maryanto, Rusmanto. *Panduan Mudah Membuat Diskless System dengan distro K12LTSP*. Dian Rakyat 2004.

[2] Mumpuni, Joko I. dan Wardono, Adisuryo. *Meningkatkan Kemampuan Jaringan Komputer dengan PC Cloning System*. Andi Yogyakarta. 2004.

[3] Mark R., *Hinkle. Desktop Linux: Think Thin*. <http://www.neoware.com/solutions/articles/article_lnxwd_12222003.html>, 22 Desember 2003.

[4] Purbo, W. Onno. *PC Cloning Windows Pakai Linux LTSP*. Andi Yogyakarta. 2006.

[5] Stansel, Paul. *Configuring Citrix Metaframe for Windows 2000*. Syngress. 2000.