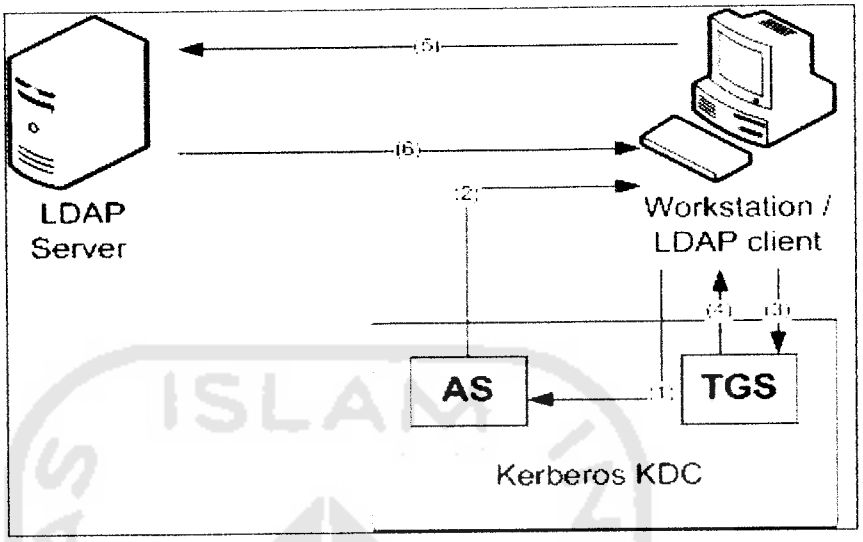
BAB 3 ANALISIS DAN PERANCANGAN SISTEM

3.1 Desain Penelitian

Kerberos dengan LDAP akan diintegrasikan dalam penelitian ini sebagai dasar solusi keamanan otentikasi bagi LDAP. Ada beberapa strategi penggabungan Kerberos dan LDAP. Penelitian ini akan mengimplementasikan salah satu strategi pengintegrasian yaitu melakukan otentikasi *bind* terhadap server LDAP menggunakan tiket Kerberos. Dengan opsi ini *user* akan diotentikasi oleh Kerberos untuk kemudian menggunakan tiket yang diperoleh untuk mengakses direktori LDAP melalui SASL GSSAPI. Secara garis besar, sistem ini dibagi menjadi dua bagian yang berbeda, yaitu otentikasi dan database *user*. Database *user* ini berisi informasi mengenai atribut-atribut yang dimiliki *user* yang selanjutnya akan ditangani oleh server LDAP. LDAP akan menyimpan informasi ini dan membuatnya tersedia bagi seluruh *host* pada *realm*. Fungsi Kerberos hanya untuk mengelola otentikasi yang aman yang tentunya Kerberos tidak tahu menahu mengenai atribut-atribut yang dimiliki oleh *user*.

3.1.1 Desain Arsitektur

Desain arsitektur sistem kerberos pada gambar 3.1 menggambarkan proses-proses yang dilakukan sistem. Model *client server* yang digunakan dalam penelitian ini yaitu server LDAP dengan otentikasi yang ditawarkan Kerberos. *Client* LDAP berupa perangkat lunak tunggal (*standalone*). Interaksi klien dengan direktori LDAP dilakukan dengan operasi-operasi LDAP *client* yang telah dijelaskan di bab sebelumnya.



Gambar 3.1 Arsitektur Sistem Kerberos

Dari gambar 3.1 dapat dijabarkan sebagai berikut :

1. *User* melakukan proses otentikasi ke Kerberos Authentication Server dan mendapatkan *initial ticket* (TGT).

2. *Kerberos Authentication Server* mengembalikan TGT.

3. *User* memulai aplikasi *client* LDAP dengan parameter :

   Mechanism -GSSAPI

   Server -hostname

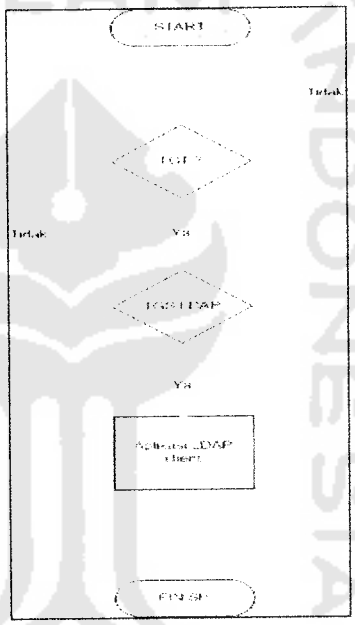
   Aplikasi *client* LDAP secara internal memanggil ldap_sasl_bind() dengan GSSAPI sebagai mekanisme otentikasi yang digunakan. *Client* memverifikasi apakah mekanisme GSSAPI memang didukung. Setelah diverifikasi, klien LDAP mengirim tiket Kerberos yang telah terbungkus *request* LDAP ke server LDAP, Aplikasi *client* (menggunakan ldap_sasl_bind()) meminta *Service Ticket* kepada server LDAP menggunakan TGT yang dimiliki *user.*

4. TGS *(Ticket Granting Server)* memberikan LDAP *service request.*

5. *Aplikasi* klien (ldap_sasl_bind()) mengirim *service ticket* (ST) kepada server LDAP. Server LDAP memverifikasi tiket tersebut menggunakan mekanisme SASL GSSAPI.

6. Berdasarkan hasil dari validasi, ldap_sasl-bind() mengembalikan nilai sukses atau gagal ke aplikasi.

3.1.2 Flowchart Otentikasi Kerberos



Gambar 3.2 Flowchart Otentikasi Kerberos dengan *Client* LDAP

Alur proses yang akan dijalankan sistem dapat diilustrasikan dengan flowchart pada gambar 3.2. Gambar 3.2 menggambarkan rancangan sistem melalui flowchart. Awal menjalankan sistem, *user* harus memiliki TGT. Apabila TGT telah diperoleh, maka proses dilanjutkan. Setelah itu *user* memperoleh TGS, maka proses selanjutnya adalah pengaksesan direktori LDAP melalui aplikasi LDAP *client*.

3.2 Analisis Kebutuhan

3.2.1 Kebutuhan Server

Berikut ini adalah komponen yang dibutuhkan oleh komputer server.

Tabel 3.1 Komponen Komputer Server

Nama	Komponen
Komputer server	a. Fedora 9 b. Kerberos V5 c. *Key distribution center* / KDC d. SASL e. OpenLDAP-2.4.15

3.2.2 Kebutuhan *Client*

Kebutuhan *client* terdiri dari kebutuhan atas perangkat keras serta perangkat lunak. Perangkat lunak yang dibutuhkan adalah sebagai berikut

Tabel 3.2 Kebutuhan Komputer *Client*

Nama	Kebutuhan komponen
Komputer *client*	a. Windows dan Linux b. IP address eth0 = 192.168.1.10 c. IP gateway = 192.168.1.2

3.2.3 Perangkat Lunak Pendukung

Berikut ini adalah perangkat lunak yang digunakan :

1. Fedora 9

   Fedora 9 merupakan salah satu distro linux yang digunakan untuk membangun sistem otentikasi dengan menggunakan *kerberos*. Sistem operasi ini digunakan sebagai server. Pertimbangan menggunakan Fedora 9 dikarenakan adanya dukungan dari *software-software* yang akan digunakan. Implementasi sistem *kerberos* dalam Fedora ini didukung dengan perintah-perintah dasar dari linux sendiri.

2. Kerberos V5

   Kerberos diperlukan pada satu platform dengan struktur *single directory tree* yang berisi file sumber maupun file objek sehingga menjadi lebih sederhana dibandingkan apabila pengimplementasian *kerberos* pada bermacam-macam platform.

3. *Key distribution center*

   KDC menerbitkan tiket *kerberos*. Masing-masing KDC berisi salinan database yang disebarkan ke slave-slave KDC pada waktu-waktu tertentu. Semua perubahan database termasuk perubahan *password* dilakukan pada master KDC. *Slave* KDC menyediakan *ticket granting service,* tetapi tidak menyediakan administrasi database. Oleh sebab itulah *client* masih dapat memperoleh tiket meskipun master KDC tidak tersedia. Namun dalam penelitian ini hanya akan diadakan sebuah master KDC tanpa mengadakan replikasinya.

4. SASL

   Simple Authentication and Security Layer (SASL) merupakan suatu metode penyediaan layanan otentikasi bagi protokol-protokol yang *connection oriented* seperti misalnya LDAP. Standar SASL didefinisikan pada RFC 2222, *Simple*

Authentication and Security Layer. Standar ini memungkinkan *client* dan server menyetujui suatu *layer* keamanan untuk enkripsi. Setelah server dan *client* terhubung, mereka menyetujui suatu mekanisme keamanan untuk percakapan selanjutnya. Salah satu mekanisme ini adalah Kerberos.

5. OpenLDAP-2.4.15

   Dalam penelitian ini digunakan distribusi OpenLDAP-2.4.15 sebagai *software* penyedia protokol LDAP.