

18.301/H/2003



MILIK PERPUSTAKAAN
INSTITUT TEKNOLOGI
SEPULUH - NOPEMBER

PERANCANGAN DAN PEMBUATAN PERANGKAT LUNAK
PENGELOLAAN SAMBA BERBASIS WEB BERBAHASA
INDONESIA

TUGAS AKHIR



RSIF
005.1
Mar
P-1
2001

PERPUSTAKAAN ITS	
Tgl. Terima	11-7-2002
Terima Dari	H
No. Agenda Prp.	217961

Disusun Oleh :

I MADE MASTRA

NRP. 2696 100 046

JURUSAN TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA
2001

**PERANCANGAN DAN PEMBUATAN PERANGKAT LUNAK
PENGELOLAAN SAMBA BERBASIS WEB BERBAHASA
INDONESIA**

TUGAS AKHIR

**Diajukan Guna Memenuhi Sebagian Persyaratan
Untuk Memperoleh Gelar Sarjana Komputer
Pada
Jurusan Teknik Informatika
Fakultas Teknologi Industri
Institut Teknologi Sepuluh Nopember
Surabaya**

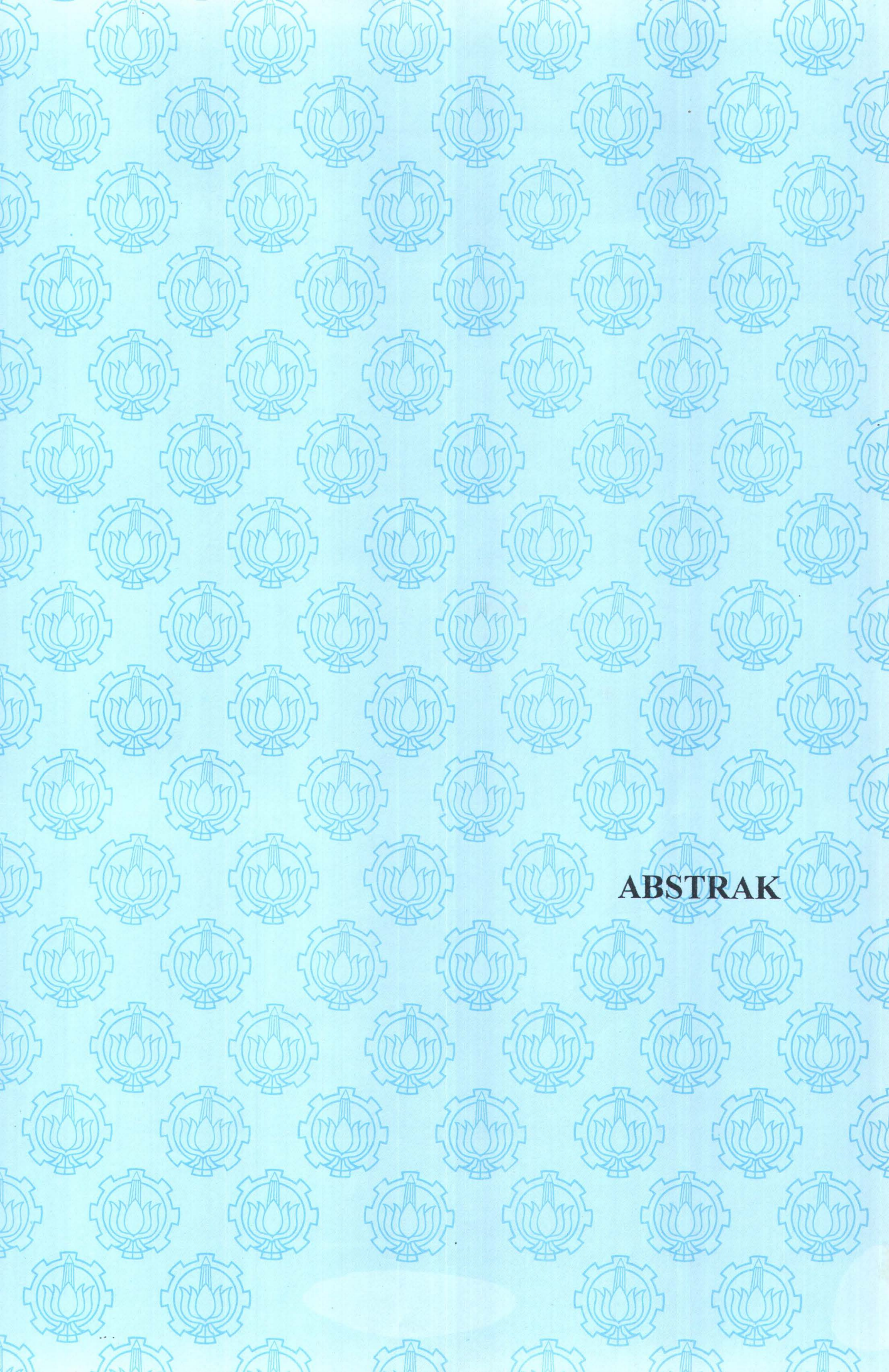
Mengetahui / Menyetujui,

Dosen Pembimbing



Ir. Muchammad Husni, M.Kom.
NIP. 131 141 100

SURABAYA
Agustus, 2001



ABSTRAK

ABSTRAK

Konfigurasi Samba dengan menggunakan cara konvensional (melalui shell) membutuhkan pengetahuan akan begitu banyaknya parameter-parameter yang disediakan oleh file-file yang berhubungan dengan konfigurasi samba, salah satunya file smb.conf.

Aplikasi untuk mengelola Samba berbasis web sebenarnya sudah ada dari paket distribusi samba sendiri yaitu SWAT (Samba Web Administration Tool) dan aplikasi lain seperti Webmin. Berdasarkan pengamatan penulis, aplikasi-aplikasi ini mempunyai beberapa kekurangan, diantaranya pembacaan parameter Samba yang kurang akurat, petunjuk konfigurasi Samba yang kurang jelas, parameter yang disediakan tidak lengkap dan bahasa yang digunakan masih berbahasa Inggris jika penulis kaitkan dengan kondisi di Indonesia yang seharusnya mengutamakan bahasa Indonesia.

Pada Tugas Akhir ini dikembangkan suatu aplikasi pengelolaan Samba berbasis web yang berbahasa Indonesia. Aplikasi ini menyediakan petunjuk konfigurasi Samba yang jelas, pembacaan parameter yang akurat, parameter-parameter yang disediakan sangat lengkap dan tentunya berbahasa Indonesia.



KATA PENGANTAR

KATA PENGANTAR

Puji dan syukur penulis panjatkan ke hadirat Tuhan Yang Maha Kuasa, karena atas perkenan-Nya lah maka penulis dapat menyelesaikan segala rangkaian pengerjaan Tugas Akhir. Laporan Tugas Akhir ini dengan judul :

Perancangan dan Pembuatan Perangkat Lunak

Pengelolaan Samba Berbasis Web Berbahasa Indonesia

disusun sebagai penutup rangkaian pelaksanaan Tugas Akhir yang merupakan kewajiban bagi setiap mahasiswa dalam menyelesaikan program strata satu (S1) pada Jurusan Teknik Informatika, Fakultas Teknologi Industri, Institut Teknologi Sepuluh Nopember, Surabaya.

Penulis dalam kesempatan ini mengucapkan terima kasih dan penghargaan yang sebesar-besarnya bagi semua pihak yang telah membantu dalam pelaksanaan Tugas Akhir, baik secara langsung maupun tidak langsung. Secara khusus penulis mengucapkan terima kasih kepada :

1. Kedua orang tua penulis yang tercinta, ayahanda I Made Mudra dan ibunda Ni Nyoman Dasni. Terima kasih yang tak terhingga atas kasih sayang, arahan, dukungan, didikan dan segala bantuan yang telah kalian berikan. Hanya Tuhan yang mampu membalasnya.
2. Saudara penulis, Ni Wayan Murni. Terima kasih atas persaudaraan serta segala bantuan dan semangat yang diberikan hingga penulis selesai kuliah.
3. Bapak Dr. Ir. Arif Djunaidy, MSc, selaku ketua Jurusan Teknik Informatika.



DAFTAR ISI

DAFTAR ISI

ABSTRAK.....	i
KATA PENGANTAR.....	ii
DAFTAR ISI.....	iv
DAFTAR GAMBAR.....	vii
BAB I PENDAHULUAN	
1.1 Latar Belakang.....	1
1.2 Permasalahan.....	2
1.3 Ruang Lingkup Permasalahan.....	4
1.4 Tujuan.....	4
1.5 Metodologi dan Sistematika Laporan.....	4
BAB II PENGENALAN SAMBA	
2.1 Pengertian Samba.....	7
2.2 Implementasi Microsoft.....	14
2.3 Distribusi Samba.....	24
2.4 Dasar-dasar Konfigurasi File smb.conf.....	26
2.5 Mengetes File Konfigurasi	27
2.6 Menjalankan Daemon Samba.....	29
2.7 Mengetes Daemon Samba.....	33
BAB III PENGELOLAAN SAMBA BERBASIS WEB	
3.1 Metode Pengelolaan Samba Berbasis Web.....	35
3.2 Contoh Aplikasi Sejenis.....	36
3.2.1 SWAT.....	36
3.2.2 WEBMIN.....	40

BAB IV	PERANCANGAN DAN PEMBUATAN PERANGKAT LUNAK	
4.1	Perancangan Perangkat Lunak.....	42
4.1.1	Arsitektur.....	42
4.1.2	Antarmuka Pengguna (User Interface).....	44
4.1.3	Proses-proses.....	48
4.2	Pembuatan Perangkat Lunak.....	64
4.2.1	Otentikasi Nama User dan Password.....	64
4.2.2	Security.....	64
4.2.3	Mengedit File smb.conf.....	66
4.2.4	Menjalankan Daemon Samba.....	73
4.2.5	Menginformasikan Status Koneksi Samba.....	76
4.2.6	Melihat Konfigurasi Samba.....	79
4.2.7	Menangani Password Samba.....	80
4.2.8	Logout.....	84
BAB V	UJI COBA DAN EVALUASI	
5.1	Uji Coba.....	86
5.1.1	Uji Coba Otentikasi Nama User dan Password.....	86
5.1.2	Mengedit Bagian Global.....	88
5.1.3	Mengedit Bagian Share.....	90
5.1.4	Mengedit Bagian Printer.....	91
5.1.5	Menjalankan Daemon Samba.....	93
5.1.6	Menginformasikan Status Koneksi Samba.....	95
5.1.7	Melihat Konfigurasi Samba.....	97
5.1.8	Letak File Konfigurasi dan Utilitas Samba.....	98
5.1.9	Menangani Password Samba.....	99
5.1.10	Penanganan Masalah Samba.....	101
5.1.11	Logout/Keluar dari Aplikasi.....	102
5.2	Analisa.....	103
BAB VI	PENUTUP	

6.1	Kesimpulan.....	104
6.2	Saran.....	104

DAFTAR PUSTAKA



DAFTAR GAMBAR

DAFTAR GAMBAR

Gambar 2.1	Skema Jaringan.....	8
Gambar 2.2	Network Neighbourhood.....	9
Gambar 2.3	Share di Server dilihat dari Phoenix.....	10
Gambar 2.4	Pemetaan network drive ke Windows letter-drive.....	11
Gambar 2.5	Network Printer Tersedia di hydra (dilihat chimaera).....	12
Gambar 2.6	Output Utilitas smbstatus.....	13
Gambar 2.7	Windows Domain Sederhana.....	14
Gambar 2.8	Menggunakan domain controller untuk otentikasi.....	16
Gambar 2.9	Windows Domain dengan Local Master dan Local Backup Browser.....	19
Gambar 2.10	Windows Workgroup Span Multi Subnet.....	22
Gambar 2.11	Konfigurasi smb.conf Sederhana.....	26
Gambar 2.12	Output Testparm.....	28
Gambar 2.13	Kode Tambahan pada BSD style-Unix.....	30
Gambar 2.14	Kode Tambahan pada System V Unix.....	31
Gambar 2.15	Output utilitas smb.client.....	33
Gambar 3.1	Login SWAT.....	37
Gambar 3.2	SWAT Bagian GLOBAL.....	37
Gambar 3.3	SWAT Bagian Share.....	38
Gambar 3.4	Parameter-parameter Share.....	39
Gambar 3.5	Login pada Webmin.....	40
Gambar 3.6	Samba Share Manager Webmin.....	41
Gambar 3.7	Editor File Share Webmin.....	41
Gambar 4.1	Input nama user dan password.....	64
Gambar 4.2	Otentikasi user terhadap file/etc/shadow.....	64
Gambar 4.3	Memberikan Cookie.....	65
Gambar 4.4	Mengecek Cookie pada file .php.....	65

Gambar 4.5	Mengecek Cookie pada file .cgi.....	65
Gambar 4.6	Membaca dan merapikan file smb.conf.....	66
Gambar 4.7	Fungsi untuk membaca bagian global/share/printer.....	66
Gambar 4.8	Membaca parameter.....	67
Gambar 4.9	Menulis Bagian Gobal.....	68
Gambar 4.10	Memilih Nama Bagian Share.....	69
Gambar 4.11	Menghapus bagian share/printer.....	70
Gambar 4.12	Membuat bagian share baru.....	71
Gambar 4.13	Memilih Nama Bagian Printer.....	72
Gambar 4.14	Membuat Bagian Printer Baru.....	72
Gambar 4.15	Menjalankan utilits testparm	73
Gambar 4.16	Menjalankan utilitas testprns.....	74
Gambar 4.17	Menjalankan utilits restart smb.....	75
Gambar 4.18	Menjalankan utilitas smbclient -L localhost.....	76
Gambar 4.19	Menjalankan utilitas smbstatus.....	77
Gambar 4.20	Menampilkan informasi di browser.....	78
Gambar 4.21	Melihat Konfigurasi Samba.....	79
Gambar 4.22	Mengecek cookie.....	80
Gambar 4.23	Input nama user dan password.....	80
Gambar 4.24	Merubah password user samba.....	81
Gambar 4.25	Menghapus user samba.....	81
Gambar 4.26	Menambah user Samba.....	82
Gambar 4.27	Mendisable user Samba.....	82
Gambar 4.28	Mengenable user Samba.....	83
Gambar 4.29	Input nama user, password dan remote machine.....	83
Gambar 4.30	Merubah password user Samba di Windows NT.....	84
Gambar 4.31	Logout/Keluar dari Swatindo.....	85
Gambar 5.1	Tampilan otentikasi nama user dan password.....	87
Gambar 5.2	Tampilan login yang gagal.....	87
Gambar 5.3	Editor global sederhana.....	88
Gambar 5.3	Editor global lengkap.....	89

Gambar 5.4	Menu awal editor share	90
Gambar 5.5	Editor share sederhana.....	91
Gambar 5.6	Menu awal editor printer.....	92
Gambar 5.7	Editor printer sederhana.....	93
Gambar 5.8	Menjalankan daemon dan utilitas.....	94
Gambar 5.9	Contoh output menjalankan utilitas smbclient -L localhost.....	94
Gambar 5.10	Status koneksi samba.....	95
Gambar 5.11	Status koneksi Samba (bawah).....	96
Gambar 5.12	Konfirmasi pemutusan koneksi.....	96
Gambar 5.13	Konfigurasi Samba.....	97
Gambar 5.14	Letak file konfigurasi dan Utilitas Samba.....	98
Gambar 5.15	Konfirmasi path file konfigurasi.....	99
Gambar 5.16	Mengelola password server.....	100
Gambar 5.17	Mengelola password client-server (NT).....	100
Gambar 5.18	Contoh artikel untuk membantu memecahkan masalah samba..	101
Gambar 5.19	Konfirmasi tidak berhak menggunakan SWATINDO.....	102
Gambar 5.20	Konfirmasi logout.....	102



BABI
PENDAHULUAN

BAB I

PENDAHULUAN

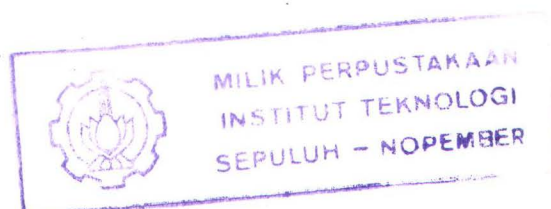
1.1 Latar Belakang

Jika dalam satu jaringan terdapat berbagai macam sistem operasi dengan berbagai macam platform, maka dibutuhkan sebuah program yang menjembatani sehingga semua mesin dapat saling berhubungan satu sama lain. Program yang menjembatani Unix (dan Linux pada umumnya) dengan mesin Windows yang cukup terkenal adalah Samba.

Samba merupakan aplikasi Linux yang memanfaatkan protokol *SMB* (*Server Message Block*). Sebagian sistem operasi memanfaatkan SMB dalam komunikasi *client-server*-nya, termasuk Windows dan OS/2. Samba memungkinkan mesin Linux berkomunikasi dengan mesin Windows/Dos atau OS/2.

Samba dapat digunakan untuk :

- Menghubungkan setiap mesin Unix (termasuk Linux) dengan mesin Dos/Windows
- Menempatkan mesin Unix (Linux) sebagai *Primary Domain Controller* sebagaimana yang dilakukan NT



Oleh sebab itu, Samba dapat dibuat sebagai *server* dengan *clientnya* dari Microsoft Windows, dengan layanan :

- Berbagi (*sharing*) file dan printer
- Membantu pengguna *browsing* di network neighborhood
- Memberikan otentikasi kepada tiap *client* yang login ke dalam domain
- Memberikan atau membantu dengan WINS *name server resolution*

Samba memiliki banyak sekali parameter yang harus dikonfigurasi agar dapat dijalankan secara optimal. File konfigurasi tersebut adalah *smb.conf*. Biasanya terletak di : */etc/*.

Cara konvensional untuk mengedit Samba adalah dengan menggunakan *text editor*, atau cara lainnya menggunakan aplikasi berbasis web seperti *SWAT* dan *Webmin*.

1.2 Permasalahan

Berdasarkan pengamatan yang dilakukan penulis, aplikasi pengelolaan Samba berbasis web yang ada sekarang masih memiliki kekurangan yang dikaitkan dengan kebutuhan administrator jaringan (Samba) di Indonesia, diantaranya :

- Petunjuk yang kurang jelas saat mengedit file *smb.conf*. Contohnya: Parameter-parameter mana saja yang perlu ditambahkan, berikut petunjuk nilai yang harus diisi. Kurangnya petunjuk mengedit file *smb.conf* ini penulis dapatkan pada aplikasi *Webmin*.

- Kurangnya petunjuk saat administrator Samba mengalami masalah dengan instalasi dan konfigurasinya Sambanya. Jadi masalah Samba itu bukan saja terletak pada konfigurasinya. Contoh : mendiagnosa *server* samba. Kurangnya petunjuk saat administrator menemui masalah dengan instalasi samba penulis dapatkan pada aplikasi *Webmin*
- Masih berbahasa Inggris. Ini merupakan kendala yang cukup berarti bagi administrator pemula dalam bahasa, belum lagi harus menguasai secara teknik masalah samba itu. Sampai saat ini belum ada aplikasi pengelolaan Samba yang berbahasa Indonesia.
- Tidak lengkapnya pilihan parameter yang disediakan. Beberapa aplikasi pengelolaan Samba hanya menyediakan pilihan parameter yang sedikit dari seharusnya. Dengan tersedianya parameter yang lengkap (kurang lebih 250 parameter), seorang administrator dapat membuat konfigurasi Samba dari yang paling sederhana sampai yang paling kompleks. Keadaan ini penulis dapatkan pada aplikasi *Webmin*.
- Pembacaan parameter Samba yang kurang akurat. Kadang-kadang parameter yang tidak ada di file *smb.conf* dinyatakan ada oleh salah satu aplikasi pengelolaan Samba, sehingga akan membuat bingung administrator tersebut. Keadaan ini penulis dapatkan pada aplikasi *SWAT*.

1.3 Ruang Lingkup Permasalahan

Pada penulisan tugas akhir ini penulis memfokuskan permasalahan dengan ruang lingkup sebagai berikut :

- Samba yang digunakan adalah versi 2.0.7
- Memberikan petunjuk yang jelas tiap item parameter.
- Memberikan petunjuk dalam menangani masalah konfigurasi dan instalasi Samba.
- Menginformasikan secara detail status Koneksi Samba.
- Menangani password Samba.
- Tidak memfokuskan pada keamanan aplikasi web.
- Hanya digunakan untuk lingkungan intranet saja.
- Bahasa yang digunakan adalah bahasa Indonesia, tentu saja dengan tetap mempertahankan istilah-istilah komputer.

1.4 Tujuan

Aplikasi ini bertujuan untuk membantu administrator jaringan baik pemula maupun yang sudah berpengalaman dalam hal mengelola Samba, baik itu mengedit file smb.conf maupun menangani masalah instalasi dan konfigurasi Samba.

1.5 Metodologi dan Sistematika Laporan

Metodologi yang digunakan dalam penyusunan Tugas Akhir ini adalah :

1. Studi Literatur
2. Perancangan dan Pembuatan Aplikasi
3. Uji Coba dan Evaluasi
4. Penulisan Laporan Tugas Akhir

Sistematika Laporan Tugas Akhir dibagi kedalam beberapa bab sebagai berikut :

Bab I Pendahuluan

Bab ini berisi latar belakang, permasalahan, ruang lingkup permasalahan, tujuan serta metodologi dan sistematika laporan.

Bab II Pengenalan Samba

Penjelasan mengenai fungsi samba, distribusinya.

Bab III Pengelolaan Samba Berbasis Web

Penjelasan mengenai metode pengelolaan Samba berbasis Web yang ada.

Disertai dengan contoh aplikasi yang sudah ada.

Bab IV Perancangan dan Pembuatan Aplikasi

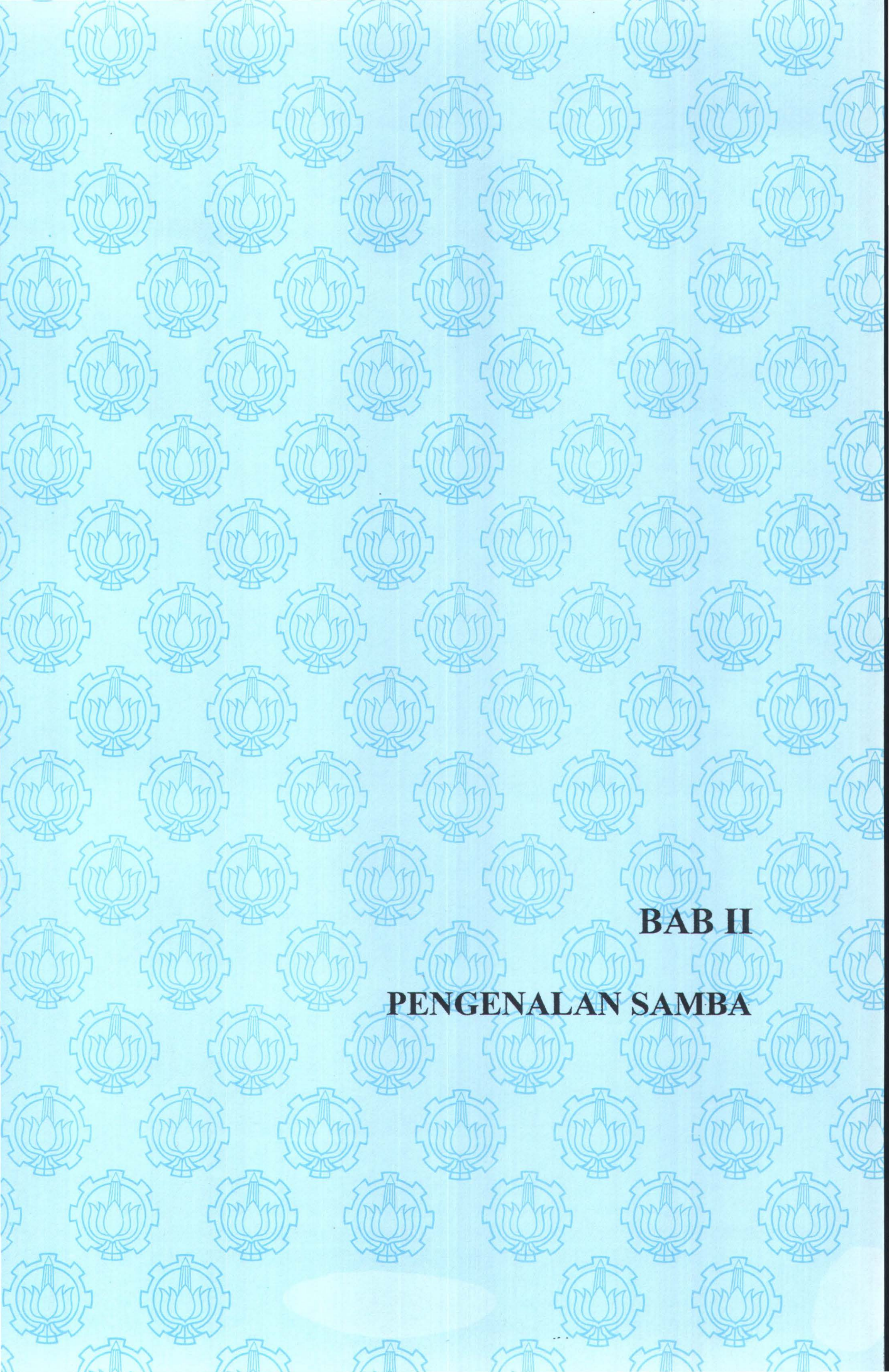
Merancang sistem, antar muka pengguna, proses-proses yang harus dilakukan. Dilanjutkan dengan mengimplementasikan hasil perancangan itu.

Bab V Uji Coba dan Evaluasi

Menguji coba aplikasi ini dan mengevaluasi hasilnya.

Bab VI Penutup

Berisi kesimpulan yang dapat diambil dari pelaksanaan Tugas Akhir beserta saran untuk pengembangan selanjutnya.



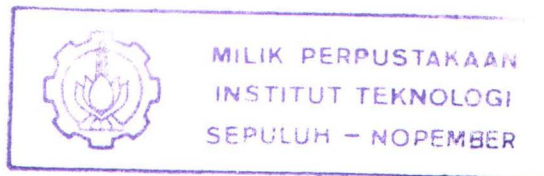
BAB II

PENGENALAN SAMBA

BAB II

PENGENALAN SAMBA

Pada bagian ini penulis akan membahas Samba secara umum. Pembahasan meliputi: pengertian Samba, implementasi Microsoft sampai dengan mengkonfigurasi file `smb.conf`.



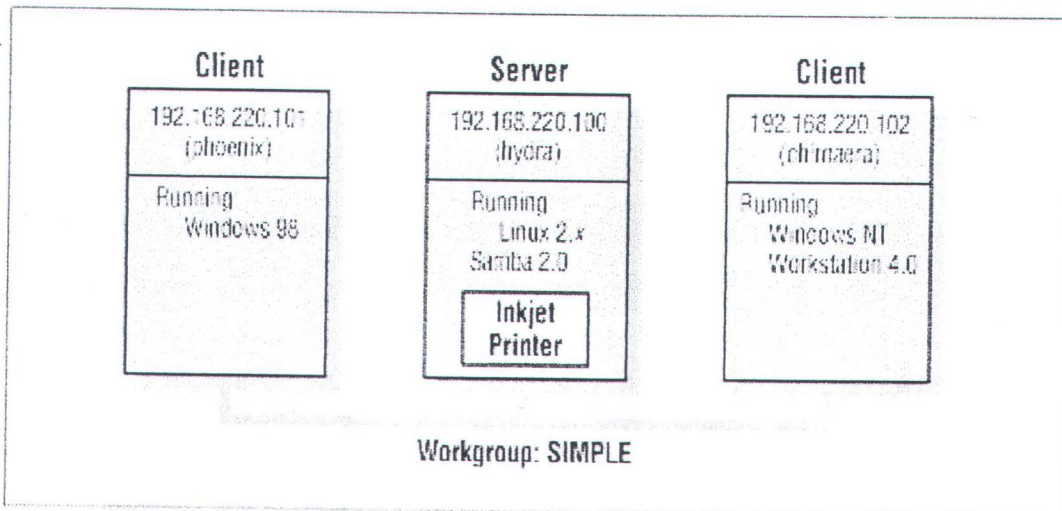
2.1 Pengertian Samba

Samba adalah sekumpulan dari aplikasi Linux yang menggunakan protokol SMB (*Server Message Block*). Banyak sistem operasi, termasuk Windows dan OS/2, menggunakan SMB untuk membentuk *client-server networking*.

Dengan mendukung protokol ini, Samba memungkinkan *server* Linux untuk lebih berperan, berkomunikasi dengan protokol jaringan yang sama seperti produk Microsoft Windows. Kemudian, Samba memungkinkan mesin Linux untuk menyamar sebagai *server* di jaringan Microsoft dan akan memberikan servis-servis berikut:

- Berbagi pakai (*sharing*) satu atau lebih *filesystem*
- *Sharing* printer baik itu di *server* maupun *client*.
- Membantu *client* saat *browsing* dengan *Network Neighborhood*.
- Mengotentikasi logging *client* ke domain Windows.
- Menyediakan atau membantu dengan WINS name *server* resolution

Sebagai contoh penggunaan Samba ini, penulis memberikan contoh konfigurasi jaringan sederhana sebagai berikut: Samba yang dijalankan dari mesin Linux, diberi nama *hydra* dan pasangan dari *client* Windows, diberi nama *phoenix* dan *chimaera*, semua terkoneksi melalui *local area network* (LAN). *Hydra* juga mempunyai printer inkjet lokal, *lp*, dan share disk dinamakan *network*. Skema jaringan sederhana ini dapat ditunjukkan pada gambar 2.1.

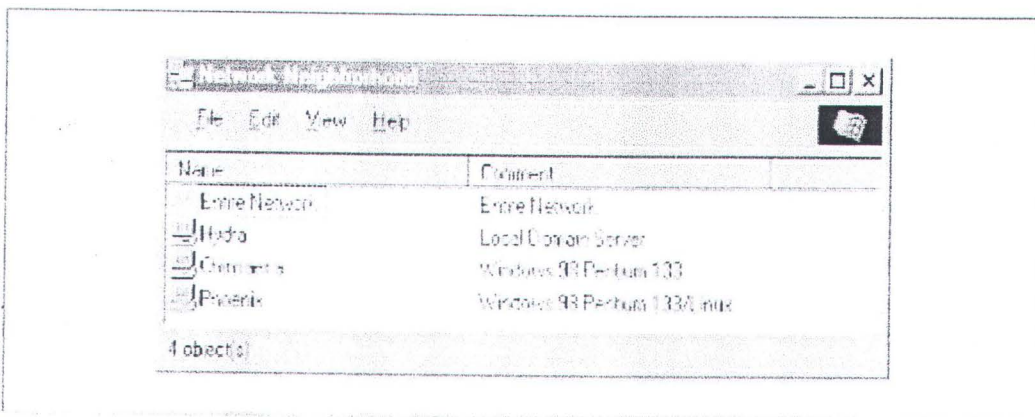


Gambar 2.1 Skema Jaringan

Pada jaringan ini, setiap komputer terdaftar pada workgroup yang sama. Workgroup adalah nama group yang mengidentifikasi koleksi tertentu dari komputer dan sumber daya di jaringan SMB. Bisa terdapat beberapa workgroup di jaringan pada sewaktu-waktu, tapi untuk contoh jaringan sederhana ini, hanya mempunyai satu: workgroup SIMPLE.

2.1.1 Berbagi Layanan Disk (*Sharing Disk Service*)

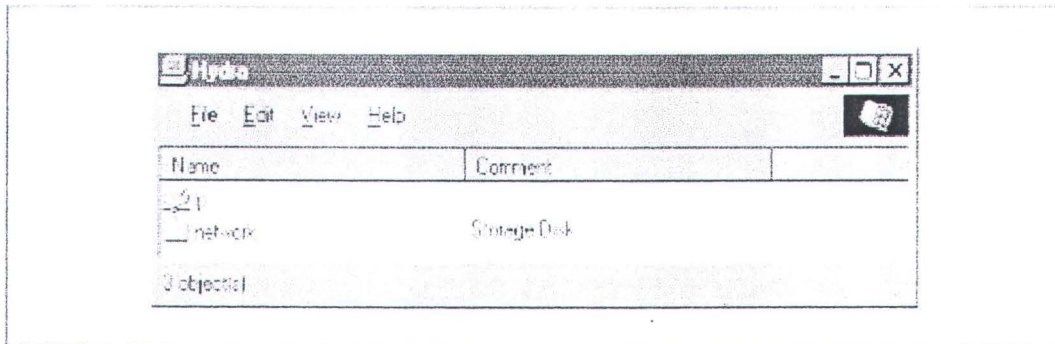
Jika semua dikonfigurasi dengan benar, seharusnya terlihat server Samba, *hydra*, melalui Network Neighborhood dari desktop *phoenix*. Gambar 2.2 menunjukkan Network Neighborhood dari komputer *phoenix*, termasuk *hydra* dan setiap komputer yang berada di workgroup *SIMPLE*. Perhatikan icon Entire Network di daftar paling atas. Sebagaimana penulis sebutkan, bisa terdapat lebih dari satu workgroup di network SMB. Jika user mengklik icon Entire Network, dia akan melihat daftar dari semua workgroup yang saat itu ada di jaringan.



Gambar 2.2 Network Neighborhood

Server *hydra* dapat diamati lebih jauh dengan klik ganda pada iconnya. Hal ini akan mengontak *hydra* dan meminta daftar sumber daya file dan printer yang disediakan oleh mesin itu. Untuk contoh ini, terdapat printer dengan nama *lp*, disk share dengan nama *network* di server, sebagaimana terlihat di gambar 2.3. Perhatikan bahwa screen Windows menunjukkan *hostname* dalam format *mixed case* (Hydra). Case ini tidak relevan di *hostname*, jadi mungkin akan terlihat

hydra, Hydra, dan HYDRA diberbagai display dan command output, tapi mereka semua menunjuk ke *single system*. Berkat Samba, Windows 98 melihat server Linux sebagai server SMB yang valid dan dapat mengakses folder *Network* yang seakan-akan hanya dari folder sistem lain.



Gambar 2.3 Share di Server dilihat dari Phoenix

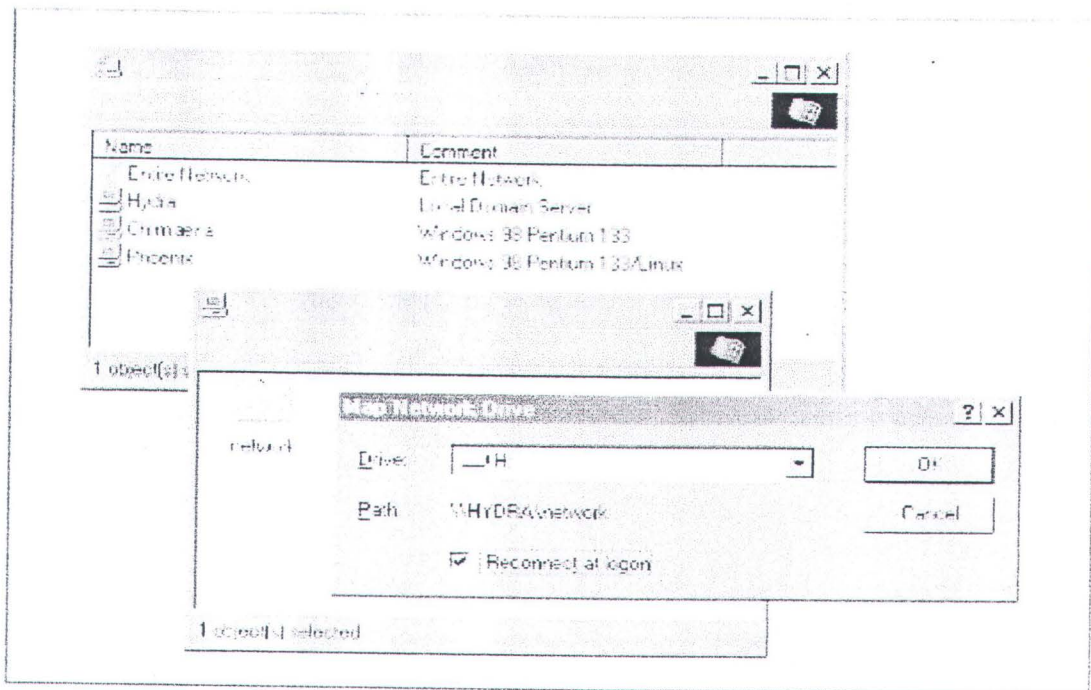
Satu fitur terkenal dari Windows 95/98/NT adalah dapat memetakan (*map*) *letter-drive* jaringan yang diketahui menggunakan opsi Map Network Drive di Windows Explorer. Selesai memetakan, aplikasi anda dapat menyimpan data, menginstall dan menjalankan program dari folder itu, dan bahkan password-melindunginya terhadap pengunjung yang tidak diundang. Gambar 2.4 memperlihatkan contoh pemetaan *letter-drive* untuk direktori *network*.

Perhatikan pada Path: entri di kotak dialog dari gambar 2.4. Cara yang sama untuk menghadirkan direktori pada mesin jaringan adalah dengan menggunakan dua *backslashe*, diikuti dengan nama dari mesin jaringan, backslash lagi diikuti direktori jaringan dari mesin, seperti berikut ini :

\\mesin\direktori jaringan

Ini dikenal dengan UNC (Universal Naming Convention) di dunia Windows. Sebagai contoh, kotak dialog pada gambar 2.4 mempresentasikan direktori jaringan di server hydra sebagai berikut:

\\HYDRA\network

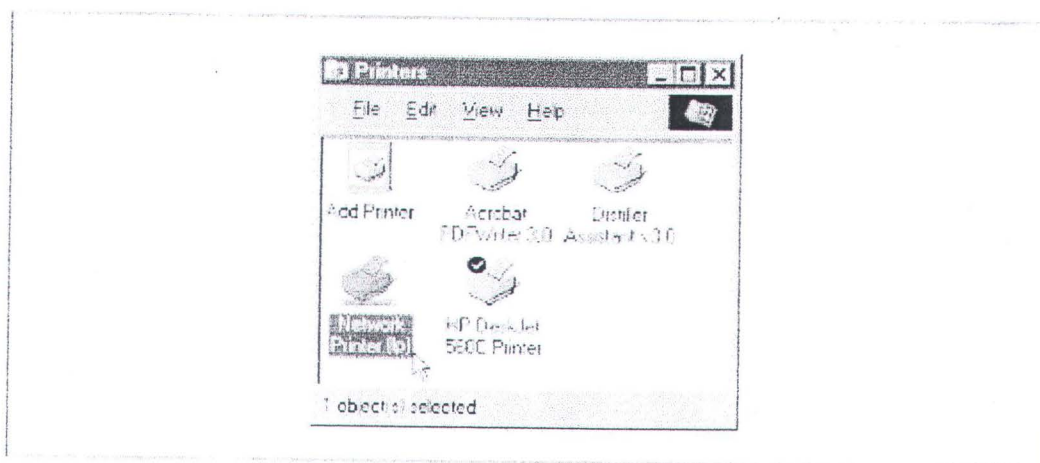


Gambar 2.4 Pemetaan *network drive* ke *Windows letter-drive*

2.1.2 Berbagi Printer (*Sharing Printer*)

Printer *lp* terlihat sebagai share yang tersedia di *hydra* pada gambar 2.3. Hal ini mengindikasikan bahwa server Linux mempunyai printer yang dapat di *share* oleh berbagai *client* SMB di workgroup. Data yang dikirimkan ke printer dari berbagai *client* akan di spool di server Linux dan diprint sesuai urutan yang diterimanya.

Konfigurasi Samba dengan printer yang di *enable*, dari sisi Windows kadang-kadang lebih mudah daripada konfigurasi disk share. Dengan klik ganda di printer dan mengidentifikasi manufaktur dan model, driver printer ini dapat diinstall di client Windows. Windows dapat mengatur berbagai informasi yang dikirim ke network printer dan mengaksesnya sebagaimana lokal printer Gambar 2.5 menunjukkan hasil network printer di window Printer di Windows 98. Sekali lagi, perhatikan pipeline attachment dibawah printer, yang mengidentifikasi sedang berada di jaringan.



Gambar 2.5 Network Printer Tersedia di *hydra* (dilihat chimaera)

2.1.3 Melihat dari Sisi Linux

Telah disebutkan sebelumnya, Samba merupakan sekumpulan dari program-program daemon. Dapat dilihat dengan perintah *netstat* dan *ps* Linux, kemudian dapat dibaca pesan-pesan yang dihasilkan melalui file-file debug biasa atau *syslog* Linux (tergantung bagaimana Samba dikonfigurasi), dikonfigurasi dari

file properti Samba *smb.conf*. Sebagai tambahan, jika ingin mengetahui status Samba dapat menggunakan utilitas *smbstatus*:

```
# smbstatus

Samba version 2.0.4

Service uid gid pid machine
-----
network davecb davecb 7470 phoenix (192.168.220.101) Sun May 16
network davecb davecb 7589 chimaera (192.168.220.102) Sun May 16

Locked files:
Pid DenyMode R/W Opendir Name
-----
7589 DENY_NONE RONLY EXCLUSIVE+BATC /home/samba/quicken/inet/common/system/help.bmp Sun May 16 21:23:40 1999
7470 DENY_WRITE RONLY NONE /home/samba/word/office/findfast.exe Sun May 16 20:51:08 1999
7589 DENY_WRITE RONLY EXCLUSIVE+BATC /home/samba/quicken/lfbmp70n.dll Sun May 16 21:23:39 1999
7589 DENY_WRITE RDWR EXCLUSIVE+BATC /home/samba/quicken/inet/qdata/runtime.dat Sun May 16 21:23:41 1999
7470 DENY_WRITE RONLY EXCLUSIVE+BATC /home/samba/word/office/osa.exe Sun May 16 20:51:09 1999
7589 DENY_WRITE RONLY NONE /home/samba/quicken/qversion.dll Sun May 16 21:20:33 1999
7470 DENY_WRITE RONLY NONE /home/samba/quicken/qversion.dll Sun May 16 20:51:11 1999

Share mode memory usage (bytes):
1043432(99%) free + 4312(0%) used + 832(0%) overhead = 1048576(100%) total
```

Gambar 2.6 Output Utilitas smbstatus

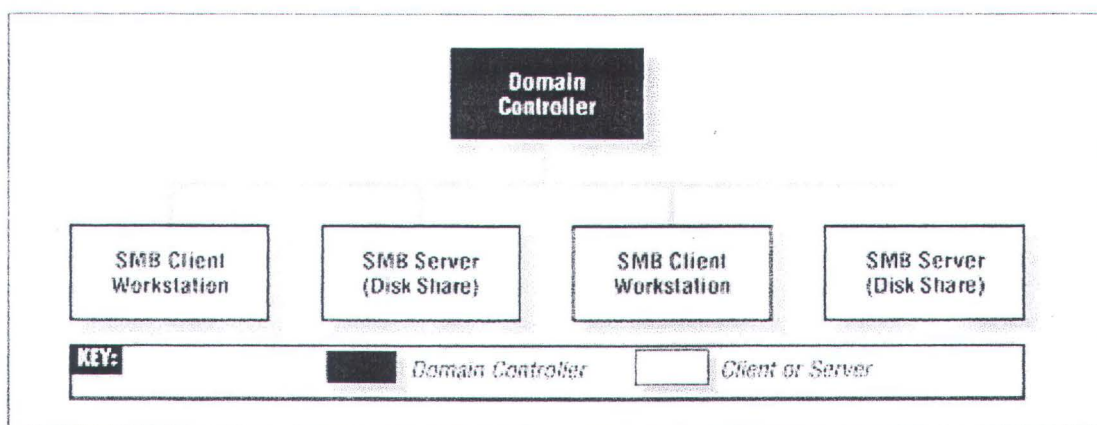
Status Samba menyediakan tiga kumpulan data, setiap data dibagi menjadi bagian terpisah. Bagian pertama memberitahu sistem sudah terkoneksi pada server Samba, mengidentifikasi setiap client dengan nama mesin (*phoenix dan chimaera*) dan alamat IP. Section kedua melaporkan nama dan status dari file-file yang sekarang digunakan di share server, termasuk status read/write dan setiap lock dari file-file. Akhirnya, Samba melaporkan jumlah memory yang telah dialokasikan pada share yang diadminstrasi, termasuk jumlah aktf yang digunakan oleh share ditambah additional overhead.

2.2 Implementasi Microsoft

Berikut beberapa implementasi Microsoft yang mendahului konsep di dunia jaringan *CIFS/SMB*.

2.2.1 Windows Domain

Workgroup adalah koleksi dari komputer SMB yang terletak di subnet dan terdaftar di group SMB yang sama. Windows domain maju selangkah yaitu *workgroup* dari mesin-mesin SMB yang mempunyai satu tambahan: yaitu *server* yang berfungsi sebagai *domain controller*. Harus ada *domain controller* bila ingin mempunyai Windows domain. Jika tidak, itu hanyalah suatu *workgroup*. Perhatikan gambar 2.7 Windows domain disebut "*Windows NT domains*" oleh Microsoft karena mereka mengasumsikan bahwa mesin Windows NT yang memegang kendali dari *domain controller*. Tetapi, Samba juga mempunyai fungsi ini, untuk selanjutnya akan dinamakan dengan "Windows domain" untuk menghindari kebingungan.



Gambar 2.7 Windows Domain Sederhana

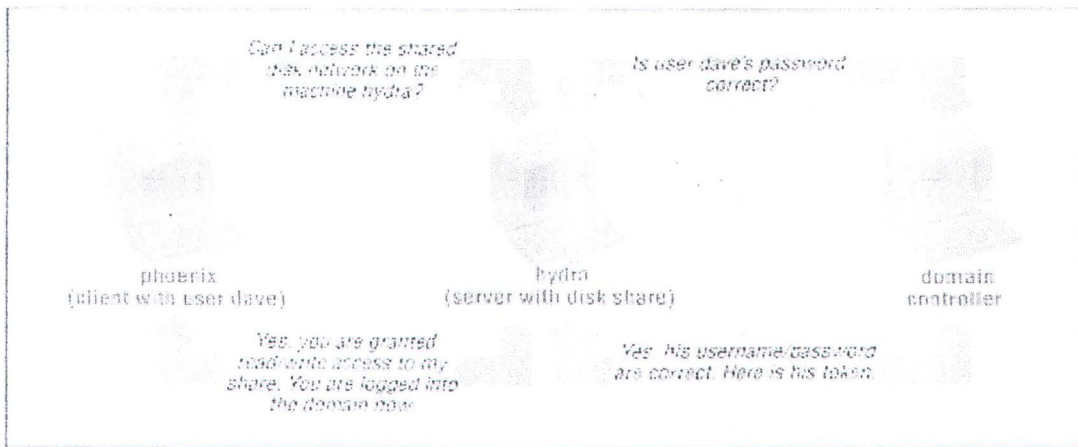
Berikut ini dua protokol terpisah yang digunakan oleh *Domain Controller* (logon *server*): pertama untuk berkomunikasi dengan mesin Windows 95/98, kedua untuk komunikasi dengan mesin Windows NT. Sedangkan Samba 2.0.7 mengimplementasikan protokol *Domain Controller* untuk Windows 95/98 (yang memungkinkan Samba berfungsi sebagai *Domain Controller* untuk mesin Windows 9x), dan tidak sepenuhnya mendukung protokol untuk Windows NT.

2.2.2 *Domain controller*

Domain controller adalah pusat saraf dari domain Windows, seperti *server* NIS adalah pusat saraf dari servis informasi jaringan Unix. *Domain controller* mempunyai berbagai tanggung jawab, salah satunya yaitu otentikasi (*authentication*). Otentikasi adalah proses dari perijinan (*granting*) atau pelarangan (*denying*) *user* mengakses *shared* sumber daya pada mesin jaringan lain, biasanya melalui penggunaan password. Setiap *Domain Controller* menggunakan *Security Account Manager* (SAM) untuk mengelola daftar kombinasi username-password. *Domain Controller* kemudian membentuk pusat penyimpanan password yang dikaitkan pada *username* (satu password per *user*), yang lebih efisien daripada setiap mesin *client* mengatur ratusan password untuk sumber daya jaringan yang tersedia.

Pada Windows domain, ketika *client* non-otentikasi meminta akses *share* pada *server*, *server* akan berkeliling dan menanyakan *domain controller* apakah *user* itu sudah diotentikasi. Jika sudah, *server* akan membuka sesi koneksi dengan hak akses yang sesuai untuk servis dan *user*. Jika tidak diotentikasi, koneksi

dilarang. Sekali *user* sudah diotentikasi oleh *domain controller*, token terotentikasi khusus akan diberikan ke *client* jadi *user* tidak perlu relogin pada sumber daya lainnya di domain. Pada poin ini, *user* diasumsikan "login" pada domainnya sendiri. Perhatikan gambar 2.8.



Gambar 2.8 Menggunakan *domain controller* untuk otentikasi

2.2.3 Primary dan Backup Domain controller

"Redundansi" adalah ide kunci dibelakang Windows domain. *Domain controller* yang saat ini aktif di domain dinamakan *Primary Domain Controller* (PDC). Bisa terdapat satu atau lebih *Backup Domain Controller* (BDC) di domain, yang akan mengambil alih pada saat *Primary Domain Controller* gagal atau menjadi tidak bisa diakses. BDC seringkali menyinkronisasi data SAM mereka dengan *Primary Domain Controller*, maka dari itu, jika permintaan meningkat, siapapun dari mereka dapat memberikan servis DC secara transparan tanpa mempengaruhi *clientnya*. Perhatikan bahwa BDC, hanya mempunyai salinan *read-only* dari SAM, mereka dapat memperbaharui data mereka hanya dengan

menyinkronkan dengan PDC. *Server* di Windows domain dapat menggunakan SAM dari setiap *Primary* atau *Backup domain controller* untuk otentikasi *user* yang mencoba mengakses sumber dayanya dan logon ke domain.

Samba dapat berfungsi sebagai *Primary Domain Controller* untuk mesin Windows 95/98 tanpa banyak masalah. Tapi, Samba 2.0 dapat berfungsi sebagai *Primary Domain Controller* hanya untuk otentikasi, sekarang ini tidak bisa diasumsikan pada tanggungjawab PDC lainnya. Juga karena protokol tertutup yang digunakan oleh Microsoft untuk menyinkronkan data SAM, Samba saat ini tidak bisa melayani sebagai *Backup Domain controller*.

2.2.4 Browsing

Sebelum *browsing*, *user* harus mengetahui nama dari komputer tertentu yang mereka inginkan untuk koneksi di jaringan, secara manual *user* dapat mengetikkan UNC seperti berikut ini di aplikasi atau file manager untuk mengakses sumber daya : \\HYDR \network\

Dengan *browsing*, *user* dapat memeriksa isi dari mesin menggunakan *Window Network Neighborhood* pada *windows client*.

2.2.4.1 Tipe-tipe Browsing

Terdapat dua tipe *browsing* yang akan di jaringan SMB/CIFS:

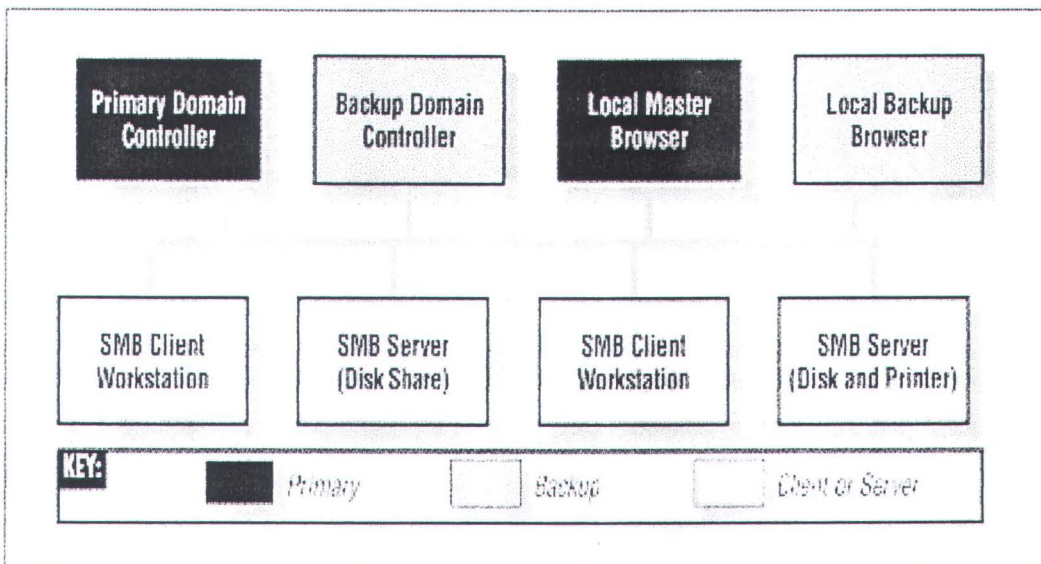
- *Browsing* daftar mesin (dengan sumber daya yang *dishare*)
- *Browsing* sumber daya yang *dishare* pada mesin tertentu.

Mari kita lihat yang pertama. Setiap subnet *Windows workgroup* atau (domain), satu komputer mempunyai tanggungjawab mengelola daftar dari mesin yang saat ini bisa diakses melalui jaringan. Komputer ini dinamakan *local master browser*, dan daftar mesin yang diatur dinamakan *browse list*. Mesin-mesin pada subnet tersebut menggunakan *browse list* untuk memotong sejumlah lalu-lintas jaringan (*network traffic*) yang dihasilkan saat *browsing*. Contohnya setiap komputer yang secara dinamis mengadakan polling untuk menentukan daftar mesin yang tersedia saat ini, komputer dapat dengan mudah meminta (*query*) pada *local master browser* untuk mendapatkan daftar terbaru yang lengkap.

Untuk browse sumber daya sebenarnya di mesin, *user* harus terkoneksi ke mesin tertentu, informasi ini tidak dapat diperoleh dari *browse list*. *Browsing* daftar sumber daya pada mesin dapat dilakukan dengan klik pada icon mesin saat terlihat di *Network Neighborhood* di Windows 95/98 atau NT. Seperti kita lihat pada awal bab, mesin akan merespon dengan daftar *share* sumber daya yang dapat diakses jika *user* sudah diotentikasi.

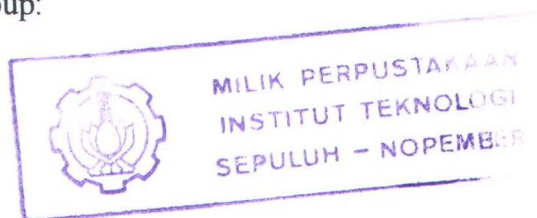
Setiap server di *Windows workgroup* diminta untuk mengumumkan kehadirannya pada *local master browser* setelah server itu meregister nama NetBIOS, dan (secara teori) mengumumkan bahwa server meninggalkan *workgroup* saat server itu *shut down*. Merupakan tanggung jawab dari *local master browser* untuk merekam (*record*) apa yang telah diumumkan oleh server tersebut. Perlu diperhatikan bahwa *local master browser* bukanlah mesin yang sama sebagai NetBIOS name server (NBNS), yang dibahas sebelumnya.

Tidak seperti fungsi sebelumnya, hampir setiap mesin Windows (NT Server, NT Workstation, 98, 95 atau Windows 3.1 for Workgroup) dapat berfungsi sebagai *local master browser*. Seperti *domain controller*, *local master browser* dapat mempunyai satu atau lebih *backup browser* di subnet lokal yang akan mengambil alih fungsi pada saat *local master browser* gagal atau menjadi tidak bisa diakses. Untuk memastikan operasi yang kerap kali berubah, *local backup browser* akan sering mensinkronkan *browse listnya* dengan *local master browser*. Diagram Windows domain diatas diperbaharui untuk mengikutkan kedua *local master* dan *local backup browser*.



Gambar 2.9 Windows Domain dengan *Local Master* dan *Local Backup Browser*

Berikut ini cara kalkulasi jumlah minimum dari *backup browser* yang akan dialokasikan pada workgroup:



- Jika terdapat antara 1 sampai 32 mesin Windows NT workstation di jaringan, atau antara 1 sampai 16 mesin Windows 95/98 di jaringan, *local master browser* mengalokasikan satu *backup browser* sebagai tambahan pada *local master browser*.
- Jika jumlah dari Windows NT workstation antara 33 sampai 64, atau jumlah dari Windows 95/98 workstation antara 17 sampai 32, *local master browser* mengalokasikan dua *backup browser*.
- Setiap group 32 NT workstation atau 16 mesin Windows 85/98 atau lebih, *local master browser* mengalokasikan *backup browser* lainnya.

Saat ini tidak terdapat batas atas dari jumlah *backup browser* yang dapat dialokasikan oleh *master browser*.

2.2.4.2 Pemilihan Browsing (*Browsing Election*)

Saat pemilihan diadakan, setiap mesin akan *broadcast* dengan informasi datagram tentang dirinya. Informasinya termasuk yang berikut ini:

- Versi dari *election protocol* yang digunakan
- Sistem operasi pada mesin
- Telah berapa lama *client* tersebut di jaringan.
- Nama host dari *client*.

2.2.5 Span Multi Subnet pada Windows Workgroup

Spanning multi subnet bukanlah *initial design* dari Windows NT 3.5 atau Windows for Workgroup. Akibatnya, Windows domain yang *span* dua subnet atau lebih, secara kenyataan, menyatukan dua atau lebih workgroup yang share nama identik. Berita baiknya adalah bahwa masih bisa menggunakan *Primary Domain Controller* untuk mengontrol otentikasi melewati setiap subnet. Berita buruknya adalah hal tersebut bukanlah hal yang mudah, semudah browsing.

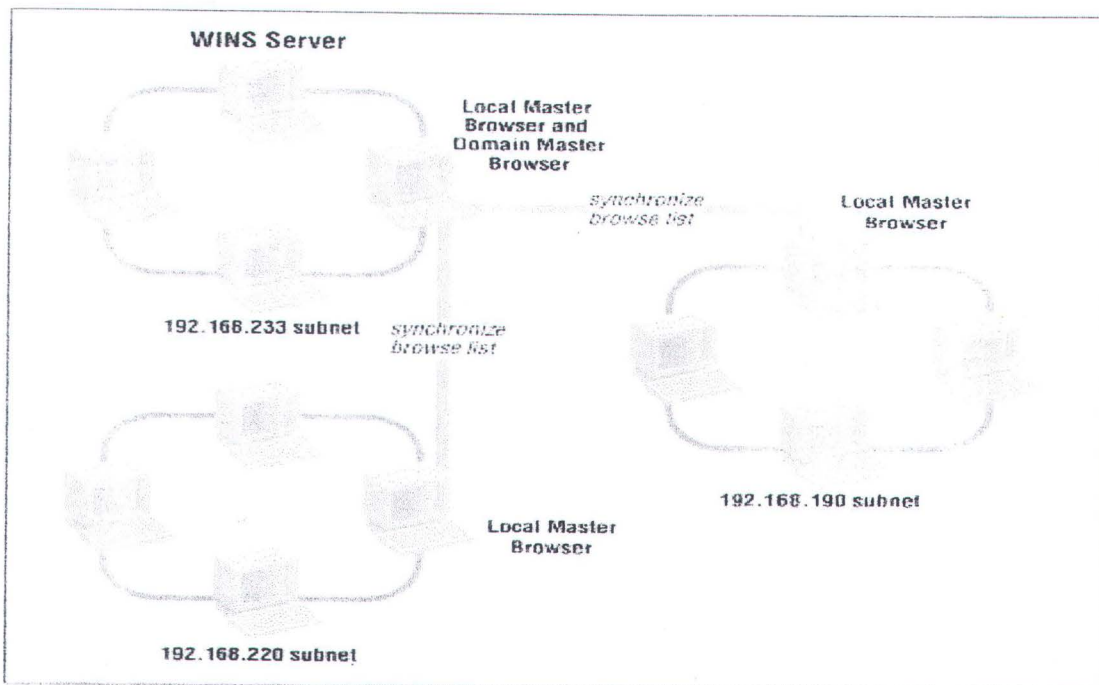
Seperti dijelaskan sebelumnya, setiap subnet harus mempunyai *local master browser*. Saat Windows domain span muli subnet, system administrator harus menugaskan satu mesin sebagai *domain master browser*. Domain master browser akan mengurus browser list untuk seluruh Windows domain. Browse list ini dibuat secara periodik dengan mensinkronkan browse list dari setiap *local master browser* dengan browse list dari *domain master browser*. Setelah sinkornisasi, *local master browser* dan *domain master browser* seharusnya mengandung entri yang sama. Perhatikan gambar 2.10 untuk ilustrasinya.

Kedengarannya bagus ? Tapi, keadaan itu tidak yang berlangsung lama karena beberapa alasan berikut:

- Jika *spanning* itu ada, *primary domain controller* selalu memainkan peran *domain master browser*. Dengan desain Microsoft, keduanya selalu share NetBIOS resource type <1B>, dan (sayangnya) tidak bisa dipisahkan.
- Mesin Windows 95/98 tidak bisa menjadi atau *bahkan menghubungi domain master browser*. Group Samba merasa bahwa ini adalah keputusan marketing dari Microsoft yang memaksa customer untuk mempunyai

paling sedikit satu Windows NT workstation (atau server Samba) pada setiap subnet dari multi-subnet workgroup.

Setiap local master browser subnet melanjutkan untuk mengelola browse list untuk subnetnya, yang menjadi tugasnya. Jadi jika komputer menginginkan untuk melihat daftar server dengan subnetnya sendiri, local master browser dan subnet itu akan diminta. Jika komputer ingin melihat daftar server diluar subnet, komputer masih bisa melihat sejauh yang dilihat oleh local master browser. Hal ini bisa bekerja karena, interval tertentu browse list dari local master browser subnet disinkronkan dengan domain master browser, yang disinkronkan dengan local master browser dari subnet lain di domain. Ini dinamakan *browse list propagation*.



Gambar 2.10 Windows Workgroup Span Multi Subnet

2.2.6 Windows Internet Name Service (WINS)

Windows Internet Name Service (WINS) adalah implementasi Microsoft dari *NetBIOS name server* (NBNS). Contohnya, WINS mewarisi banyak karakteristik NetBIOS. Pertama, WINS adalah flat; kita hanya boleh menamakan mesin *fred* atau seperti *workgroup* CANADA atau USA. Sebagai tambahan, WINS adalah dinamis: saat *client* pertama online, diminta untuk melaporkan nama hostnya, alamatnya, dan *workgroup*nya pada *server* WINS. *Server* WINS akan mengelola informasi selama *client* secara periodik refresh registrasi WINS-nya, yang mengindikasikan bahwa *client* itu masih terkoneksi pada jaringan. Perhatikan bahwa *server* WINS bukanlah domain atau *workgroup* tertentu; mereka dapat muncul dimana saja dan melayani siapa saja.

2.2.7 Apa yang dapat dilakukan Samba ?

Tabel 2.1 menyimpulkan fungsi Samba yang dapat dan tidak dapat dilakukan di Windows NT Domain atau Windows workgroup. Sebagaimana yang kita lihat, karena banyak protokol NT Domain yang khusus dan tidak didokumentasikan oleh Microsoft, Samba tidak dapat menyinkronkan datanya dengan baik dengan Microsoft *server* dan tidak dapat berfungsi sebagai backup pada umumnya. Namun dengan versi 2.0.x, Samba mempunyai dukungan terbatas untuk protokol otentikasi *Primary Domain Controller* dan mengembangkan lagi fungsi setiap saat.

Fungsi	Dapatkah ?
<i>File Server</i>	Ya
<i>Printer Server</i>	Ya
<i>Primary Domain Controller</i>	Ya (Samba 2.1 atau disarankan lebih tinggi)
<i>Backup Domain controller</i>	Tidak
Otentikasi Windows 95/98	Ya
Local Master Browser	Ya
Local Backup Browser	Tidak
Domain Master Browser	Ya
Primary WINS <i>Server</i>	Ya
Secondary WINS <i>Server</i>	Tidak

Tabel 2.1: Fungsi Samba (versi 2.0.7)

2.3 Distribusi Samba

Seperti dijelaskan sebelumnya, Samba sebenarnya terdiri dari beberapa program. Program utama yang datang dengan distribusi Samba dikelompokkan menjadi dua daemon. Tanggung jawab dari masing-masing daemon:

smbd

Daemon *smbd* bertanggungjawab untuk mengurus *share* sumber daya antara mesin *server* Samba dan *client*nya. Daemon ini melayani file, print dan browser untuk *client* SMB melewati satu atau lebih jaringan. *smbd* mengurus

semua informasi antara *server* Samba dan *client* jaringan. Sebagai tambahan, *smbd* bertanggung jawab untuk otentikasi *user* (*user authentication*), *locking* sumber daya dan *data sharing* melalui protokol SMB.

nmbd

Daemon *nmbd* adalah *nameserver* sederhana yang mengikuti fungsionalitas *server WINS* dan *NetBIOS name* sebagaimana dapat ditemukan pada paket LAN MANAGER. Daemon ini mendengar permintaan *nameserver* dan menyediakan informasi yang sesuai saat dipanggil. Juga menyediakan *browse list* untuk *Network Neighborhood* dan berpartisipasi saat *browsing elections*.

Distribusi Samba juga datang dengan sekumpulan kecil *tool command-line* Unix, sebagai berikut :

smbclient

Seperti *client* FTP yang digunakan untuk koneksi ke *share* Samba.

smbtar

Program untuk *backing up* data pada *share*, sejenis dengan *command art* Unix.

nmblookup

Program yang menyediakan *NetBIOS over TCP/IP name lookups*.

smbpasswd

Program yang mengijinkan administrator untuk merubah *encrypted password* yang digunakan oleh Samba.

smbstatus

Program untuk melaporkan koneksi jaringan saat ini pada *share* di *server* Samba.

testparm

Program sederhana untuk memvalidasi file konfigurasi Samba.

testprns

Program yang menguji apakah berbagai printer sudah dikenal oleh daemon *smbd*.

2.4 Dasar-dasar Konfigurasi File *smb.conf*

Kunci untuk konfigurasi Samba adalah file konfigurasi Samba itu sendiri: yaitu *smb.conf*. File ini biasanya ditempatkan di direktori */etc*.

Berikut konfigurasi file *smb.conf* sederhana :

```
[global]
    workgroup = SIMPLE
[test]
    comment = Hanya untuk mengetes saja
    path = /export/samba/test
    read only = no
    guest ok = yes
```

Gambar 2.11 Konfigurasi *smb.conf* Sederhana

File konfigurasi pendek ini memberitahukan *server* Samba agar memberikan direktori */export/samba/test* di *server* sebagai *share* SMB/CIFS yang dinamakan *test*. *Server* juga menjadi bagian dari *workgroup* bernama *SIMPLE*.


```
# mkdir /export/samba/test
# chmod 777 /export/samba/test
```

Setup diatas dilihat dari sisi sekuriti adalah yang paling buruk. Untuk sementara, karena untuk mengetes Samba, sisi sekuritinya dapat diabaikan. Sebagai tambahan, ada beberapa persoalan *encrypted password* yang akan dihadapi dengan *client* Windows.

Jika Windows 98 atau Windows NT Service Pack 3 keatas yang digunakan maka harus ditambahkan entri berikut *encrypt passwords = yes* pada sesi [global] pada file konfigurasi Samba. Sebagai tambahan, program *smbpasswd* (biasanya terletak di */usr/local/samba/bin*) harus digunakan untuk memasukkan kembali kombinasi *username/password* dari *user* itu ke *server* Linux. Sebagai contohnya, jika kita menginginkan untuk mengizinkan *user* linux "made" untuk mengakses *share* dari *client* SMB, kita bisa mengetikkan *smbpasswd -a made*. Pertama kali *user* ditambahkan, program akan mengeluarkan *output error* yang mengatakan kalau database *encrypted password* tidak ada. Jangan khawatir, akan dibuatkan database untuk *user* tersebut. Pastikan kombinasi *username/password* yang ditambahkan pada database terenkripsi sama dengan *username* dan *password* yang ingin kita gunakan di sis *client* Windows.

2.5 Mengetes File Konfigurasi

File konfigurasi perlu dites/diuji untuk memastikan bahwa secara sintak sudah benar. Parser untuk mengetes, *testparm*, akan memeriksa file *smb.conf* untuk

kesalahan sintak dan melaporkan apa yang ditemukan dengan daftar/rincian servis yang di *enable* di mesin kita. Sebagai contoh, dapat dilihat bahwa ditengah kecerobohan sang admin agar *server* segera berfungsi, admin tersebut salah mengetikkan *workgroup* sebagai *workgrp* :

```
Load smb config files from smb.conf
Unknown parameter encountered: -"workgrp"
Ignoring unknown parameter "workgrp"
Processing section "[test]"
Loaded services file OK.
Press enter to see a dump of your service definitions
# Global parameters
[global]
    workgroup = WORKGROUP
    netbios name =
    netbios aliases =
    server string = Samba 2.0.5a
    interfaces =
    bind interfaces only = No
...(isi diabaikan)...
[test]
    comment = For testing only, please
    path = /export/samba/test
    read only = No
    guest ok = Yes
```

Gambar 2.12 Output Testparm

2.6 Menjalankan Daemon Samba

Terdapat dua daemon yaitu *smbd* dan *nmbd* yang harus dijalankan agar Samba berfungsi dengan baik. Terdapat tiga cara untuk menjalankannya:

- Manual (dengan mengetikkan)
- Sebagai daemon yang *stand-alone*
- Dari *inetd*

2.6.1 Menjalankan Daemon Samba Secara Manual

Jika tergesa-gesa, *user* dapat menjalankan daemon Samba secara manual.

Sebagai root, perintah-perintah berikut harus dijalankan:

```
# /usr/local/samba/bin/smbd -D
# /usr/local/samba/bin/nmbd -D
```

2.6.2 Daemon yang Stand-Alone

Untuk menjalankan proses Samba sebagai daemon stand-alone, perlu ditambahkan perintah-perintah ke skrip startup linux. Jenis ini tergantung pada sistem yang digunakan apakah BSD Unix atau System V Unix.

2.6.2.1 BSD Unix

Dengan BSD-style Unix, perlu ditambahkan kode berikut pada file *rc.local*, yang biasanya ditemukan di direktori */etc* or */etc/rc.d*.

```
if [ -x /usr/local/samba/bin/smbd ]; then
    echo "Starting smbd..."
    /usr/local/samba/bin/smbd -D
    echo "Starting nmbd..."
    /usr/local/samba/bin/nmbd -D
fi
```

Gambar 2.13 Kode Tambahan pada BSD style-Unix

Kode ini sangat sederhana, yaitu mengecek untuk melihat jika file *smbd* sudah menjalankan permissionnya, dan kalau sudah, kode ini akan menjalankan setiap daemon Samba saat sistem booting.

2.6.2.2 System V Unix

Dengan System V, mungkin akan sedikit lebih rumit. System V biasanya menggunakan skrip untuk menjalankan dan menghentikan daemon sistem. Oleh karena itu, Samba perlu diinstruksikan saat Samba itu berjalan (running) dan saat berhenti (stop). Isi dari direktori */etc/rc.local* perlu dimodifikasi dan ditambahkan seperti program berikut yang dinamakan *smb*:


```

#!/bin/sh
# Contains the "killproc" function on Red Hat Linux
./etc/rc.d/init.d/functions
PATH="/usr/local/samba/bin:$PATH"
case $1 in
    'start')
        echo "Starting smbd..."
        smbd -D
        echo "Starting nmbd..."
        nmbd -D
        ;;
    'stop')
        echo "Stopping smbd and nmbd..."
        killproc smbd
        killproc nmbd
        rm -f /usr/local/samba/var/locks/smbd.pid
        rm -f /usr/local/samba/var/locks/nmbd.pid
        ;;
    *)
        echo "usage: smb {start|stop}"
        ;;
esac

```

Gambar 2.14 Kode Tambahan pada System V Unix

Dengan skrip ini, servis SMB dapat dijalankan dan dihentikan dengan perintah berikut:

```

# /etc/rc.local/smb start
Starting smbd...
Starting nmbd...
# /etc/rc.local/smb stop
Stopping smbd and nmbd...

```



2.6.3 Menjalankan Samba Daemon dari Inetd

Daemon *inetd* adalah super daemon Internet di system Unix. Daemon ini mendengar pada port TCP yang didefinisikan di */etc/services* dan mengeksekusi program yang sesuai untuk setiap port, yang didefinisikan di */etc/inetd.conf*.

Jika ingin menjalankan Samba Daemon dari *inetd*. Pertama buka */etc/services* di *text editor* kita. Jika belum didefinisikan perlu tambahkan dua baris berikut:

```
netbios-ssn      139/tcp
netbios-ns       137/udp
```

Berikut, edit */etc/inetd.conf*. Cari dua baris berikut dan tambahkan jika mereka tidak ada. Jika sudah mempunyai baris *smbd* dan *nmbd* di dalam file, edit baris tersebut dan arahkan pada *smbd* dan *nmbd* yang telah diinstall. Jenis Unix tertentu mungkin menggunakan sintak yang berbeda di file ini; gunakan entri yang ada dan manual *inetd.conf* sebagai panduan:

```
netbios-ssn stream tcp nowait root /usr/local/samba/bin/smbd smbd
netbios-ns  dgram  udp wait  root /usr/local/samba/bin/nmbd nmbd
```

Akhirnya, kill setiap proses *smbd* dan *nmbd* dan kirimkan proses *inetd* suatu sinyal hangup (HUP). (Daemon *inetd* membaca kembali file konfigurasinya pada sinyal HUP). Untuk melakukan ini, admin dapat menggunakan perintah *ps* untuk menemukan proses ID-nya, kemudian sinyal proses ID dengan perintah berikut:

```
# kill -HUP process_id
```

Setelah itu, Samba seharusnya sudah bisa berjalan.

2.7 Mengetes Daemon Samba

Sungguh sulit untuk dipercaya sudah hampir selesai dengan setup *server* Samba. Sisanya adalah memastikan bahwa semua berjalan dengan baik sebagaimana seharusnya. Caranya adalah dengan menggunakan program *smbclient* untuk memeriksa servis apa yang *server* berikan pada jaringan.

```
# smbclient -U% -L localhost
Added interface ip=192.168.220.100 bcast=192.168.220.255
nmask=255.255.255.0
Domain=[SIMPLE] OS=[Unix] Server=[Samba 2.0.5a]
  Sharename      Type            Comment
  -----
  test           Disk           For testing only, please
  IPC$           IPC           IPC Service (Samba 2.0.5a)

  Server          Comment
  -----
  HYDRA           Samba 2.0.5a

  Workgroup       Master
  -----
  SIMPLE         HYDRA
```

Gambar 2.15 Output utilitas smb.client

Jika muncul masalah, jangan panik ! Cobalah menjalankan daemon secara manual, dan periksa *output* sistem atau file debug di `/usr/local/samba/var/log.smb`

Jika berfungsi dengan baik, berarti sudah sukses setup *server* Samba dengan disk *share*.



BAB III
PENGELOLAAN SAMBA
BERBASIS WEB

BAB III

PENGELOLAAN SAMBA BERBASIS WEB

3.1 Metode Pengelolaan Samba Berbasis Web

Kunci untuk mengkonfigurasi samba adalah file *smb.conf*. Seorang admin dapat membuat konfigurasi dari yang sederhana sampai yang sangat kompleks. Biasanya seorang admin menggunakan *text editor* untuk mengeditnya. Cara lain untuk mengkonfigurasi samba adalah menggunakan aplikasi yang berbasis web.

Keuntungannya adalah dapat mengadminstrasi samba secara remote, dan tentunya dengan web akan tampak lebih indah. Dua aplikasi yang terkenal adalah *SWAT* dan *Webmin*.

Kebutuhan umum yang harus dipenuhi oleh aplikasi web ini adalah :

- Form untuk mengedit bagian global dan share
- Tersedianya parameter yang lengkap.
- Menjalankan daemon *smbd* dan *nmbd*.
- Fasilitas penanganan password Samba.
- Informasi koneksi Samba
- Dokumentasi yang cukup, baik itu manual maupun troubleshooting Samba.

3.2 Contoh Aplikasi Sejenis :

Saat ini terdapat dua aplikasi pengelolaan Samba berbasis web yang cukup terkenal yaitu SWAT dan Webmin. Sub bab ini akan menjelaskan secara singkat kedua aplikasi tersebut.

3.2.1 SWAT

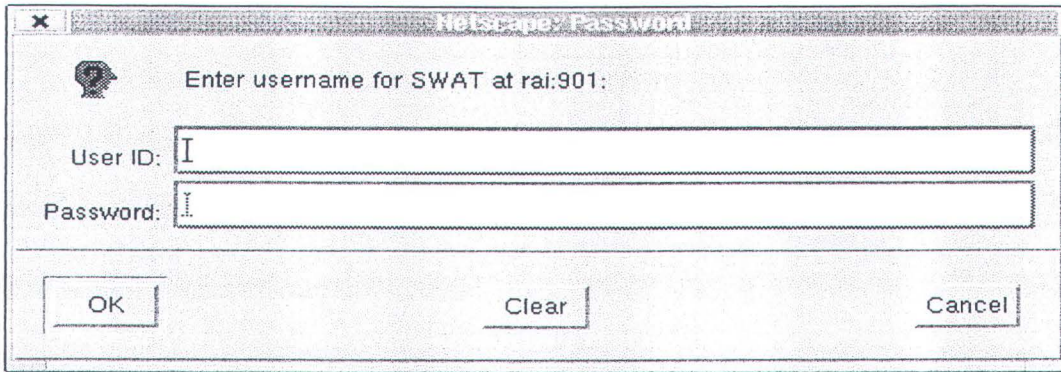
SWAT sudah disertakan sejak Samba 2.0. SWAT berjalan sebagai daemon dibawah inetd dan menyediakan editor berbasis form di web browser untuk membuat dan memodifikasikan file-file konfigurasi SMB.

Langkah mengkonfigurasi yaitu menambahkan *Samba Web Administration Tool (SWAT)* ke file konfigurasi */etc/services* dan */etc/inetd.conf*.

- Untuk menambahkan SWAT, harus ditambahkan baris berikut di akhir file */etc/services* : `swat 901/tcp`
- Langkah kedua menambahkan baris ini ke */etc/inetd.conf*. Namun tidak boleh lupa untuk merubah path ke binary SWAT jika menginstall SWAT di lokasi berbeda dari default */usr/local/samba*.

```
Swat      stream    tcp      nowait.400    root
/usr/local/samba/bin/swat swat
```

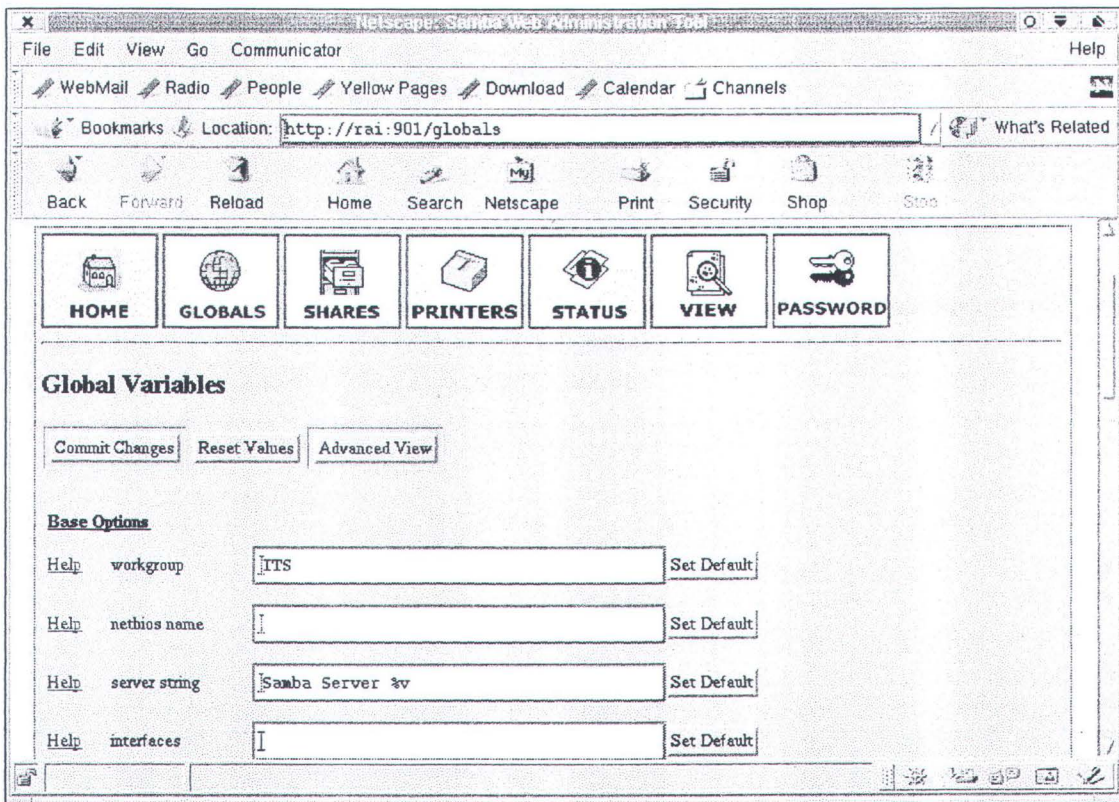
Berikut beberapa tampilan dari aplikasi SWAT ini :



Gambar 3.1 : Login SWAT

Setelah login kita akan melihat ada tujuh item pada panel atas screen.

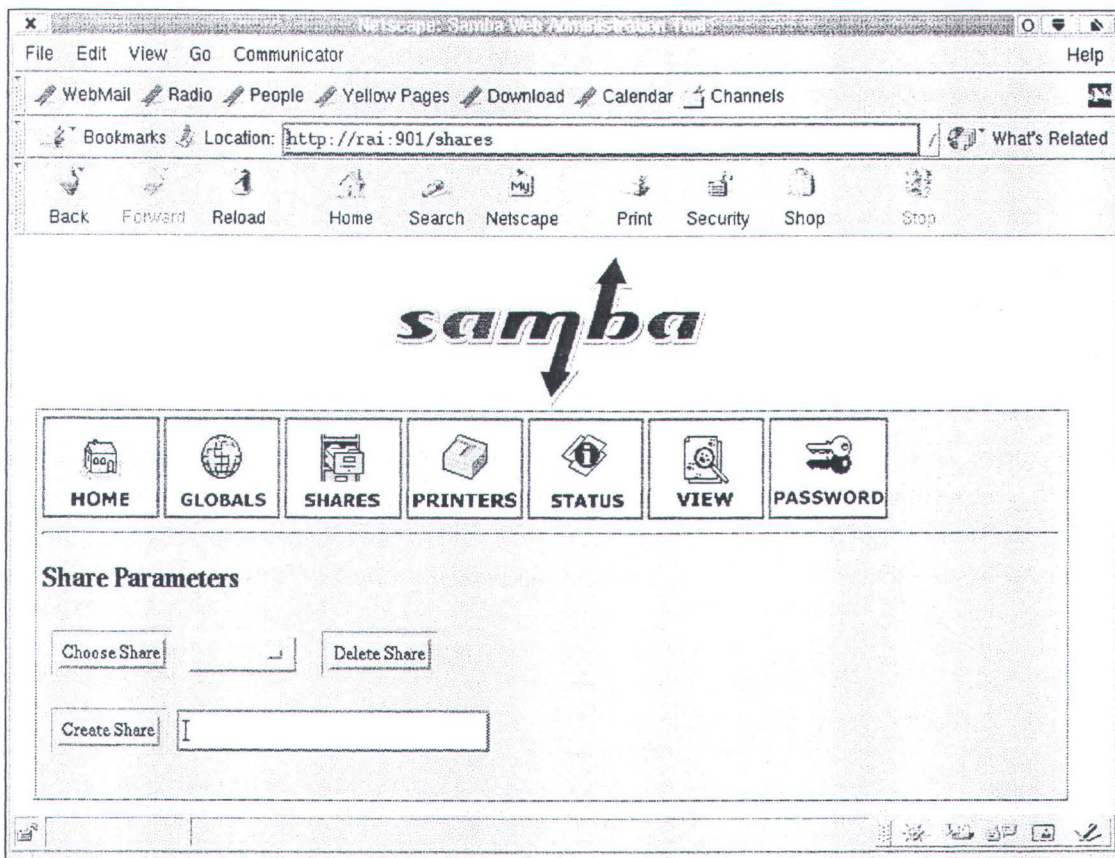
Selanjutnya kita masuki bagian GLOBAL.



Gambar 3.2 : SWAT Bagian GLOBAL

Samba terdiri dari dua bagian secara umum yaitu bagian global dan bagian share, dengan parameter tersendiri. Pada screen GLOBAL diatas terdapat dua pilihan konfigurasi yaitu konfigurasi sederhana (normal) dan lengkap (*advanced*). Konfigurasi sederhana dimaksudkan kalau admin hanya ingin membuat konfigurasi sederhana dan tidak kompleks, tentunya parameter yang digunakan tidak sebanyak parameter pada konfigurasi lengkap.

Berikut tampilan *SHARE* :

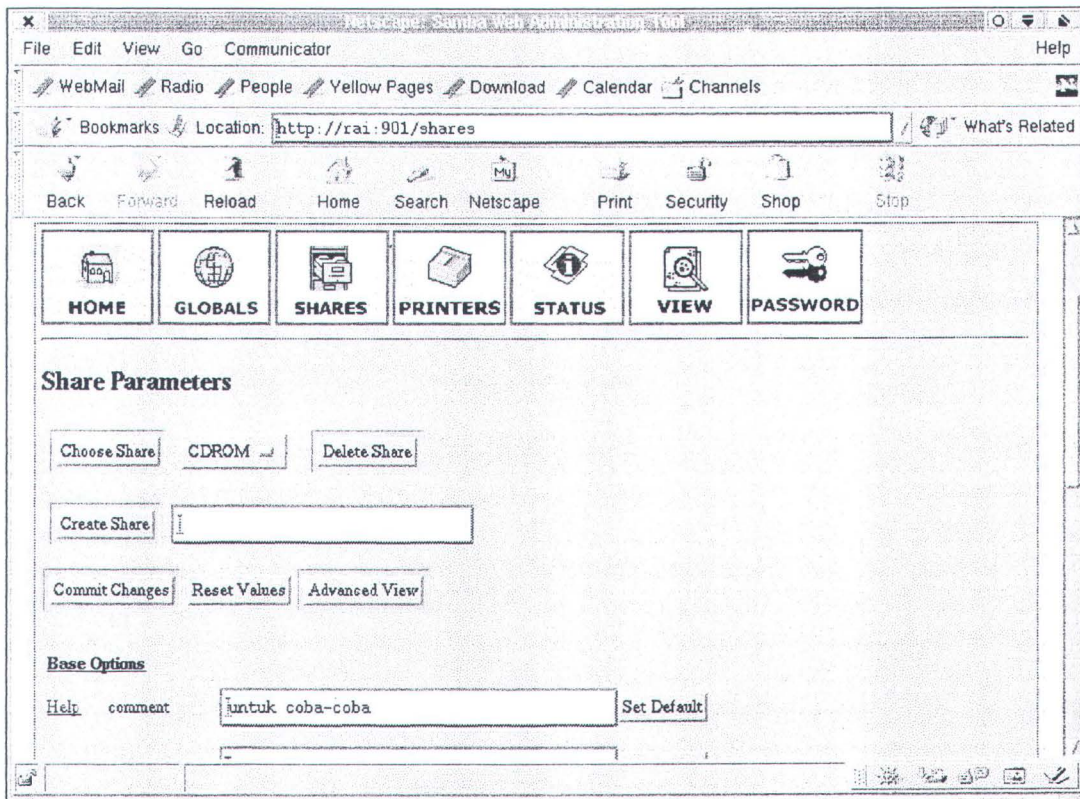


Gambar 3.3 : SWAT Bagian Share

Diatas terlihat ada beberapa tombol dengan fungsi sebagai berikut :

- Choose Share : akan menampilkan share yang terpilih di field sebelah kanannya di halaman berikutnya.
- Delete Share : akan menghapus share yang terpilih.
- Create Share : digunakan untuk membuat share yang baru. Jika nama share yang baru itu sama dengan nama share yang sudah ada, maka share baru tidak dibuat tetapi akan menampilkan konfigurasi share dengan nama tersebut.

Berikut tampilan parameter-parameter share :

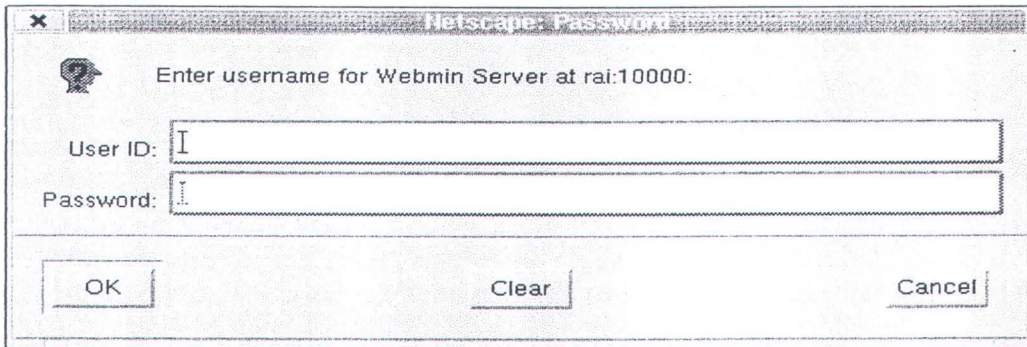


Gambar 3.4 : Parameter-parameter Share.

Demikianlah pengenalan singkat SWAT, penulis tidak membahas semua menu dan fitur yang dimiliki oleh SWAT secara detail.

3.2.2 WEBMIN

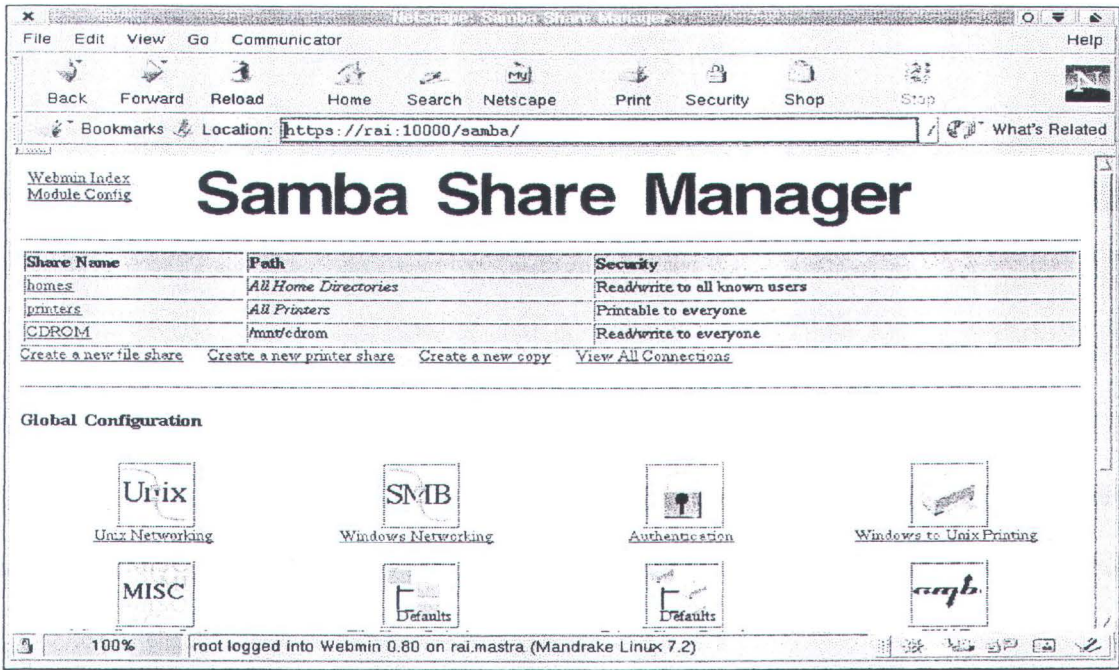
Webmin adalah aplikasi web untuk mengelola *server* secara keseluruhan. Jadi pengelolaan Samba hanyalah salah satu fiturnya saja. Untuk menggunakan Webmin seperti pada aplikasi SWAT kita harus login dulu, dengan tampilan login seperti :



The image shows a Netscape password dialog box. The title bar reads "Netscape: Password". The main text area contains the prompt "Enter username for Webmin Server at rai:10000:". Below this prompt are two text input fields. The first is labeled "User ID:" and the second is labeled "Password:". At the bottom of the dialog, there are three buttons: "OK", "Clear", and "Cancel".

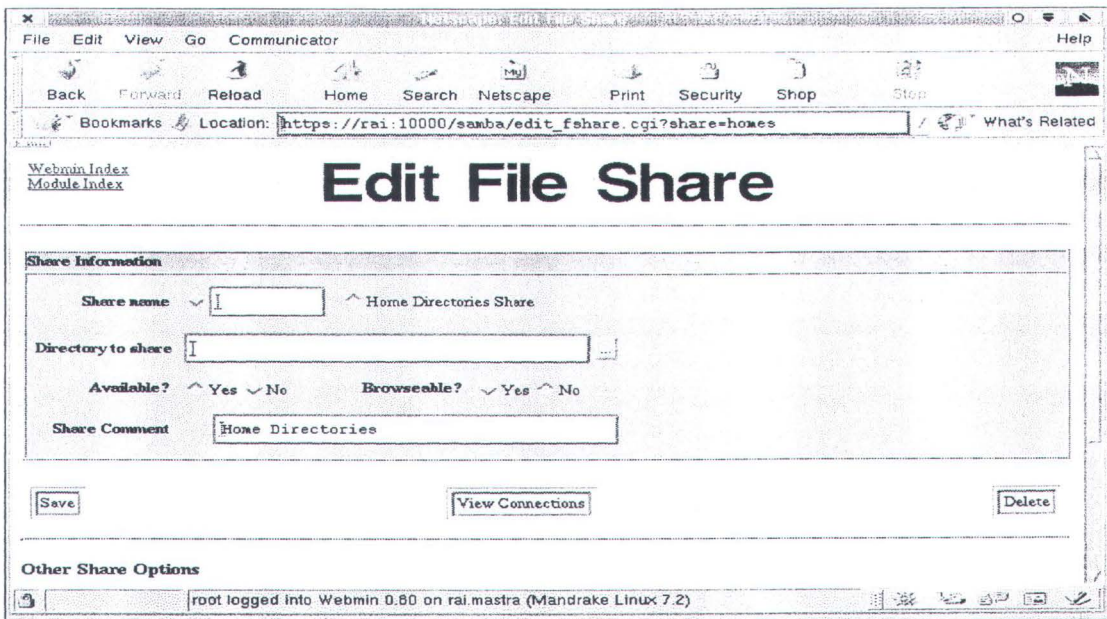
Gambar 3.5 : Login pada Webmin

Setelah itu akan terdapat banyak fitur pengelolaan *server* yang ditawarkan oleh Webmin. Untuk pengelolaan Samba, webmin juga mengatur bagian global dan printer . Dan juga di lengkapi dengan menu merestart daemon Samba. Menu testparm tidak ada karen sudah dijamin paramater yang dihasilkan sudah pasti benar secara sintak. Parameter-parameter yang disediakan tidak selengkap pada aplikasi SWAT.



Gambar 3.6 : Samba Share Manager Webmin

Berikut kita ambil contoh form konfigurasi share dengan share “homes”



Gambar 3.7 : Editor File Share Webmin



BAB IV
PERANCANGAN DAN PEMBUATAN
PERANGKAT LUNAK

BAB IV

PERANCANGAN DAN PEMBUATAN

PERANGKAT LUNAK

Pada bab ini penulis akan menjelaskan perancangan dan pembuatan perangkat lunak pengelolaan Samba berbasis web. Perancangan dan pembuatan aplikasi ini berdasarkan atas cakupan ruang lingkup masalah.

4.1 Perancangan Perangkat Lunak

Tahap perancangan meliputi Arsitektur, Antarmuka Pengguna (*User Interface*) dan Proses.

4.1.1 Arsitektur

Arsitektur pengelolaan Samba berbasis web ini melibatkan Samba 2.0.7, *Web Server Apache* 1.3.17 dan bahasa pemrograman php4 dan Perl.

- **Samba 2.0.7.**

Penulis menggunakan linux distribusi Mandrake 7.2 kernel 2.2.16, yang secara default menyertakan Samba 2.0.7.

- **Apache 1.3.17**

Web server ini mempunyai beberapa kelebihan antara lain :

- Tangguh, fleksibel, *web server* dengan kemampuan HTTP/1.1

- Mengimplementasikan protokol terbaru, termasuk HTTP/1.1
- Dapat dikembangkan dengan modul-modul pihak ketiga
- Dapat dikustomisasi dengan menulis modul menggunakan Apache module API
- Menyediakan source code
- Berjalan di Windows NT/9x, Netware 5.x, OS/2, dan versi dari Unix.

- **PHP4**

PHP4 adalah suatu bahasa script yang ditanamkan pada HTML dan diproses pada sisi *server*. Banyak tata cara penulisannya diambil dari bahasa C, Java dan Perl. Bahasa script ini adalah untuk memungkinkan kita merancang halaman Web membuat suatu halaman dinamis secara cepat.

Dengan pemrograman sisi *server* memungkinkan anda membuat aplikasi CGI (Common Gateway Interface) yang digunakan pengolahan data form sebagai hasil umpan balik pengunjung situs kita.

Suatu kemampuan yang paling menonjol dan nyata dari PHP4 adalah integrasi dengan database, adapun produk database yang didukung oleh adalah : Oracle, Sybase, mSQL, MySQL, Solid, ODBC, Adabas D, FilePro, Velocis, Informix, dBASE, Unix dbm dan PostgresSQL.

- **Perl**

Perl adalah suatu akronim, singkatan dari Practical Extraction and Report Language. Didesain oleh Larry Wall sebagai alat bantu untuk menulis program di lingkungan UNIX. Perl mempunyai kemampuan dan fleksibilitas suatu bahasa pemrograman tingkat tinggi seperti C. Kenyataannya banyak fitur dari suatu bahasa pemrograman yang dipinjam dari C.

Perl tidak membutuhkan compiler dan linker khusus untuk merubah program yang kita tulis menjadi “working code”. Kita cukup menulis program dan memberitahu perl untuk menjalankannya. Perl menyediakan semua fitur dari bahasa script seperti *sed* dan *awk*.

4.1.2 Antarmuka Pengguna (*User Interface*)

Antarmuka yang akan dirancang penulis hampir sama dengan antarmuka SWAT. Berdasarkan pengamatan, penulis menilai tampilan SWAT paling baik untuk mengkonfigurasi Samba. Oleh karena kemiripan tampilan ini aplikasi yang penulis buat dinamakan dengan **SWATINDO**. Terutama mendukung dokumentasi dan petunjuk konfigurasi yang sangat lengkap. Sehingga kalau admin mengalami kesulitan ketika mengisi nilai parameter, admin dapat mengikuti petunjuk yang langsung menjelaskan ke point permasalahan.

Setelah diteliti halaman web (interface) yang dibutuhkan aplikasi ini sebagai berikut :

- **Halaman Otentikasi**

Halaman ini merupakan pintu gerbang menggunakan aplikasi SWATINDO. Pada halaman ini dilakukan otentikasi nama user dan password admin. Dan sedikit diberikan penjelasan mengenai aplikasi SWATINDO ini.

- **Halaman Otentikasi yang Salah**

Halaman ini akan dieksekusi jika admin salah dalam memasukkan *username* dan atau *passwordnya*. Tampilannya hampir sama dengan Halaman otentikasi cuma ditambahkan keterangan login yang dimasukkan masih salah.

- **Halaman Dokumentasi Umum**

Berisikan link-link dokumentasi umum yang berhubungan Samba. Dokumentasi umum yang dimaksud adalah manual-manual :

- Daemon-daemon : file *smbd* dan *nmbd*.
- Utilitas administratif : file *smbstatus*, *SWAT*, *smbpasswd*, *make_smbcodepage*, *make_unicodemap*, *testparm*, *testprns*.
- Utilitas Umum : file *nmblookup*, *smbtar*, *smbclient*, *smbspool*.
- File-file Konfigurasi : *smb.conf*, *lmhosts*, *smbpasswd*.
- Informasi tambahan : file Pengenalan Samba, Join dengan Domain NT, merubah permission Unix menggunakan NT.

- **Halaman Global**

File konfigurasi *smb.conf* dibagi menjadi dua bagian secara umum yaitu bagian global dan *share*. Bagian *share* dibagi menjadi *share* direktori

(selanjutnya dinamakan hanya *bagian share*) dan *bagian printer*. Halaman global ini merupakan form untuk mengkonfigurasi bagian global dari file `smb.conf`.

Halaman global ini dibagi menjadi dua yaitu : global sederhana dan global lengkap. Demikian juga untuk halaman *Share* dan halaman Printer. Yang membedakannya adalah ketersediaan parameter-parameter. Jadi kalau di halaman sederhana, pilihan parameternya lebih sedikit daripada halaman lengkap. Yang secara otomatis mengakibatkan konfigurasinya lebih sederhana daripada yang lengkap.

Baik itu halaman sederhana maupun lengkap harus berisikan : link-link penjelasan masing-masing parameter dan tombol nilai default dari parameter itu.

- **Halaman Share**

Merupakan form untuk konfigurasi bagian *share* dari file `smb.conf`.

Baik halaman *share* sederhana dan *share* lengkap berisikan fasilitas antara lain:

Memilih *share*, menghapus *share* dan membuat *share* baru. Tentu saja fasilitas mengedit parameter-parameter *share*.

- **Halaman Printer**

Merupakan form untuk konfigurasi bagian printer dari file `smb.conf`.

Terdapat fasilitas untuk memilih, menghapus printer, membuat printer baru, dan form untuk mengedit parameter-parameter printer.

- **Halaman Operasi Server**

Merupakan halaman berisikan tombol-tombol untuk menjalankan *server* samba. Diantaranya :

- testparm
- testprns
- restart smbd
- smbclient.

- **Halaman Status Koneksi**

Berisikan informasi status koneksi yang tengah terjadi. Informasi ini dibedakan menjadi tiga kelompok yaitu :

- Koneksi Aktif

Berisikan informasi PID, *Client*, Alamat IP, Tanggal.

- Share aktif

Berisikan informasi Nama *Share*, *User*, Group, PID, *Client*, Tanggal.

- File Terbuka

Berisikan informasi PID, Sharing, R/W, oplock, File, Tanggal.

- **Halaman Konfigurasi Samba**

Mempresentasikan konfigurasi samba saat itu yang dibaca dari file `smb.conf`.

- **Halaman Letak File Konfigurasi Samba**

Form untuk menerima inputan path dari masing-masing file konfigurasi Samba, seperti path file `smb.conf`, path file `smbpasswd` dan lainnya.

- **Halaman Password**

Terdapat dua fasilitas penanganan Samba yaitu :

- Penanganan password *server* (lokal)
- Penanganan password *client-server*.
- **Halaman Tanya Jawab**
Berisikan link-link artikel yang sangat dibutuhkan oleh admin, jika admin mengalami kesulitan dalam mengkonfigurasi samba.

4.1.3 Proses-proses

Aplikasi pengelolaan Samba berbasis web ini secara umum membutuhkan proses-proses sebagai berikut :

- Login / Otentikasi nama user dan password.
- Security (*Cookie*)
- Mengedit file *smb.conf*.
- Menjalankan daemon samba.
- Menginformasikan status koneksi samba.
- Melihat konfigurasi Samba
- Letak file konfigurasi Samba
- Menangani password Samba.
- Petunjuk penanganan masalah Samba
- Logout / Keluar dari aplikasi

4.1.3.1 Login / Otentikasi Nama User dan Password

Login harus dengan nama *user* root dan passwordnya harus sesuai dengan password root sistem. Login dengan *user* lain tidak diperbolehkan. Bila

user salah memasukkan login maka akan ada informasi bahwa login yang dimasukkan salah. Login disini digunakan hanya untuk perijinan penggunaan aplikasi saja. Untuk menjalankan program-program daemon, kita tetap mengeset SUID (*Set User Id On Execution*) pada program tersebut. Hal penting untuk *security* dimulai disini yaitu mengeset *cookie*. Juga dilakukan pencegahan agar password root tadi tidak terlihat lagi jika user menekan tombol *back* pada browser.

4.1.3.2 **Security (Cookie)**

Untuk mendukung keamanan aplikasi ini, penulis mencoba menggunakan *cookie* selain login biasa. Cookie adalah mekanisme untuk menyimpan data pada browser sehingga kita dapat melakukan pelacakan atau pengidentifikasian user yang berkunjung ke web site kita. Pemberian cookie dimulai saat user itu login. Dan setiap user mengakses halaman tertentu, cookie ini selalu dicek. Jika cookie ini ada dan benar maka user berhak mengakses halaman itu dan sebaliknya jika cookie ini salah maka user tidak berhak mengakses halaman itu

4.1.3.3 **Mengedit File smb.conf.**

Setelah login *user* melakukan pengeditan terhadap file *smb.conf*. Pengeditan disini artinya membaca dan menulis perubahan yang dilakukan oleh *user*. Membaca dan menulis disini dibagi menjadi tiga bagian yaitu :

- Bagian global
- Bagian share

- Bagian printer

Tahap pertama sebelum mengedit ketiga bagian diatas yaitu harus membaca dan merapikan file smb.conf. Hasil dari merapikan file ini disimpan di suatu variabel yang akan digunakan juga oleh program-program yang lainnya. Program yang dimaksud adalah yang menangani bagian global, share, printer dan melihat konfigurasi samba.

Berikut algoritma pada pembacaan dan merapikan file smb.conf :

- Mengecek file smb.conf. Jika tidak ada maka akan ada konfirmasi yang memberitahu tidak adanya file smb.conf ini.
- Jika ada maka terdapat beberapa proses diantaranya :
 - Menghilangkan karakter komentar seperti # dan ;
 - Menangani nama-nama share mengikuti kemampuan yang dimiliki oleh SWAT.
 - Merapikan baris-baris parameter Samba.

Variabel dari pembacaan file smb.conf ini disimpan dan digunakan oleh program-program lainnya.

4.1.3.3.1 Bagian Global

Variabel pembacaan file smb.conf itu kemudian diolah untuk mendapatkan bagian-bagian yang lebih kecil seperti bagian global, share dan printer dengan menggunakan teknik-teknik tertentu.

- **Membaca Bagian Global**

Bagian global ditandai dengan [global]. Regular Expression sangat diperlukan disini.

Algoritma pembacaan bagian global :

- Mencari karakter/kata yang sesuai berdasarkan regular expression (RE)

Dengan RE sebagai berikut :

```
^[global\\][\+|\^|\!|\<|\>|?|\{|\\}\|#|\s|\w|\=|\@|\%|\.|\/  
|\-|\,|\*|\\\\|\$|\~|\(|\)|\:\\"|\'|\&]*\/i
```

- **Membaca Parameter Global**

Digunakan saat mengedit bagian global, dan menulis bagian global.

Berdasarkan variabel pada bagian global kita bisa mencari parameter-parameter global yang ada. Algoritma berikut ini juga berlaku untuk bagian share dan printer.

Algoritma membaca parameter, baik itu global, share dan printer.

- Variabel yang berisi bagian global/share/printer di *split* berdasarkan “/n”.
- Sebanyak jumlah *split* tadi dilakukan proses sebagai berikut :
 - *Split* lagi dengan delimiter “=” pada masing-masing baris (hasil *split* diatas diasumsikan baris). Karakter sebelum “=” adalah nama parameter dan setelah “=” adalah nilai parameter itu.

- Dibandingkan nama parameter itu dengan nama parameter yang dicari, jika sama maka akan ada nilai yang dikembalikan berupa satu baris tersebut.

- **Menulis Bagian Global**

Setelah selesai mengedit bagian global, maka dilanjutkan dengan menuliskan nilai parameter pada file `smb.conf`. Menulis bagian global ini agak sedikit berbeda dengan menulis bagian `share/printer`.

Algoritma penulisan bagian global ke file `smb.conf` sebagai berikut :

- Menggunakan variabel hasil pembacaan file `smb.conf` dan variabel pembacaan bagian global.
- Jika file `smb.conf` itu kosong, maka dituliskan `[global]`
- Jika file `smb.conf` itu tidak kosong, hanya mengandung bagian `share/printer` maka ditambahkan pada awal file nama `[global]`
- Variabel hasil pembacaan bagian global ini disalin dengan nama lain. Contoh : variabel pembacaan global lama bernama `global_lama` dan disalin menjadi `global_baru`.
- Nama-nama parameter baik itu `global/share/printer` dimasukkan ke suatu array.
- Sebanyak jumlah array tersebut dilakukan proses sebagai berikut :
 - Jika nilai parameter pada kotak input tidak kosong maka dilakukan

- ✓ Memanggil fungsi membaca parameter untuk mencari apakah parameter sekarang ini ada di variabel `global_baru` ada atau tidak.
- ✓ Jika ada/ketemu baris parameter lama itu diganti dengan baris parameter yang baru. Kalau tidak ketemu maka variabel `global_baru` ditambah dengan baris parameter ini.
- Jika nilai parameter pada kotak input kosong maka dilakukan proses berikut : memanggil fungsi membaca parameter untuk menemukan parameter saat ini di variabel `global_baru`, kalau sudah ketemu maka baris parameter itu dihapus.
- Kemudian variabel `global_lama` di variabel file `smb.conf` itu ganti dengan `global_baru` yang berisikan nilai-nilai terbaru.
- Akhirnya variabel file `smb.conf` ini ditulis ke file `smb.conf`.

4.1.3.3.2 Bagian Share

Proses pada bagian share ini banyak yang sama dengan bagian global, jadi pada beberapa proses hanya memanggil fungsi/file yang sama.

- **Membaca Bagian Share**

Proses membaca bagian share ini hampir sama dengan membaca bagian global, namun diawali dengan proses pemisahan antara nama share dengan nama printer, dengan algoritma sebagai berikut :

- Dipilih terlebih dahulu apakah suatu *share* itu adalah bagian *share* atau bagian printer.
- Yang dijadikan patokan adalah parameter *printable=yes*. Bila suatu bagian ada parameter *printable=yes* maka dapat dipastikan itu adalah bagian printer. Selanjutnya sama dengan membaca bagian global diatas.

- **Membaca Parameter Share**

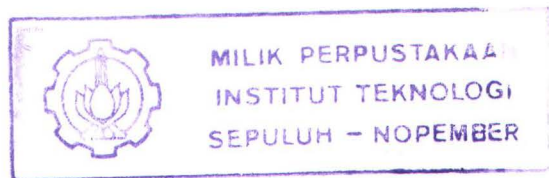
Prinsipnya sama dengan membaca parameter pada bagian global diatas.

Tentu parameternya bagian *share*.

- **Menulis Bagian Share**

Proses menulis bagian share dapat dibagi tiga yaitu :

1. Menuliskan nilai-nilai parameter share (hampir sama dengan menulis bagian global)
2. Membuat share baru
3. Menghapus share



Proses-proses diatas berdasarkan algoritma sebagai berikut :

1. Tidak melakukan pengecekan apakah file *smb.conf* itu kosong atau tidak. Selanjutnya sama caranya dengan membaca bagian global diatas.

2. Untuk membuat share baru cukup dengan menuliskan nama share itu dan diikuti dengan parameter *path=/tmp*.
3. Untuk menghapus suatu share caranya : mencari bagian share yang dimaksud dari variabel file *smb.conf*, kemudian kita ganti bagian share itu dengan karakter “”. Lalu variabel itu dituliskan kembali ke file *smb.conf*.

4.1.3.3 Bagian Printer

Proses pada bagian printer ini banyak yang sama dengan bagian global, jadi pada beberapa proses hanya memanggil fungsi/file yang sama.

- **Membaca Bagian Printer**

Caranya sama dengan membaca bagian share. Kalau suatu bagian ada parameter *printable=yes* berarti bagian itu adalah bagian printer. Selanjutnya cara pembacaan bagian printer ini dibaca sama dengan membaca bagian global.

- **Membaca Parameter Printer**

Prinsipnya sama dengan membaca parameter pada bagian global diatas. Tentu parameternya bagian printer.

- **Menulis Bagian Printer**

Proses menulis bagian printer dapat dibagi tiga yaitu :

1. Menuliskan nilai-nilai parameter printer (hampir sama dengan menulis bagian global)
2. Membuat share printer baru
3. Menghapus share printer

Proses-proses diatas berdasarkan algoritma sebagai berikut :

1. Tidak melakukan pengecekan apakah file `smb.conf` itu kosong atau tidak. Selanjutnya sama caranya dengan membaca bagian global diatas.
2. Untuk membuat printer baru cukup dengan menuliskan nama printer itu dan diikuti dengan parameter `path=/tmp` dan `printable=yes`.
3. Untuk menghapus share printer caranya sama dengan menghapus suatu share biasa.

4.1.3.4 Menjalankan Daemon Samba

Setelah file `smb.conf` selesai diedit, proses selanjutnya adalah menjalankan utilitas dan daemon Samba dengan urutan langkah sebagai berikut :

- **Testparm** : untuk melihat apakah sintak di file `smb.conf` sudah benar atau masih ada kesalahan.

Algoritma yang digunakan :

- Mengecek cookie, jika benar maka akan dilakukan proses berikut :
 - ✓ Menjalankan utilitas `testparm` pada file `smb.conf`
 - ✓ Jika gagal menjalankan `testparm` akan muncul informasi kesalahan yang terjadi

- **smbclient -L localhost** : Menjalankan utilitas *smbclient* -L localhost untuk melihat servis apa saja yang diberikan oleh *server*.

Algoritma yang digunakan :

- Mengecek cookie, jika benar maka akan dilakukan proses berikut :
 - ✓ Menjalankan utilitas *smbclient* -L pada file *smb.conf*
 - ✓ Jika gagal menjalankan *smbclient* -L localhost akan muncul informasi kesalahan yang terjadi
- Jika cookie tidak benar maka akan muncul informasi bahwa user tidak berhak mengakses aplikasi ini.

4.1.3.5 Menginformasikan Status Koneksi Samba

Untuk mengetahui client mana saja yang sedang mengakses server linux, share mana saja yang sedang diakses, kemudian mode sharenya bagaimana dapat digunakan utilitas *smbstatus*. Tapi informasi yang diterima masih belum dikelompokkan. Maksud pengelompokkan ini agar lebih terfokus pada item yang dituju, apakah clientnya, sharenya atau mungkin yang lain. Perlu dipresentasikan status koneksi berdasarkan output utilitas *smbstatus* tadi. Tampilannya nanti mengacu pada aplikasi SWAT.

Algoritma yang digunakan sebagai berikut :

- Mengecek cookie, jika benar maka dilakukan proses-proses berikut :

- ✓ Menjalankan utilitas `smbstatus`, jika ada kesalahan maka akan muncul informasi mengenai kesalahan tersebut. Hasil keluaran `smbstatus` disimpan disuatu variabel `smbstatus`.
- ✓ Variabel ini dimanipulasi sehingga diperoleh informasi yang diperlukan, dan diatur tampilannya dalam tabel tersendiri :
 - Koneksi Aktif
Berisikan informasi PID, *Client*, Alamat IP, Tanggal.
PID ini nanti berupa link, dimana kalau user meng-klik link ini maka koneksi dengan client tersebut akan terputus.
 - Share aktif
Berisikan informasi Nama *Share*, *User*, *Group*, PID, *Client*, Tanggal.
 - File Terbuka
Berisikan informasi PID, *Sharing*, R/W, *oplock*, File, Tanggal.
Juga mengatasi nama file yang berisi spasi.
- Jika cookienya salah maka muncul halaman konfirmasi yang memberitahu user tidak berhak mengakses aplikasi ini.

4.1.3.6 Melihat Konfigurasi Samba

Setelah melakukan sekian banyak perubahan parameter maka user/admin dapat melihat file `smb.conf` secara keseluruhan. File `smb.conf` ini ditampilkan dengan tertata rapi dan memenuhi persyaratan baik itu penulisan nama-nama share maupun nama-nama parameter yang dijamin valid secara sintak.

Algoritma melihat konfigurasi Samba :

- Mengecek cookie, jika benar maka proses berikut akan dilakukan :
 - ✓ Menggunakan variabel file `smb.conf`. Variabel ini adalah hasil manipulasi file `smb.conf` , dimana sudah dilakukan penghapusan karakter komentar, merapikan tampilan parameter, mengecek nama share.
 - ✓ Menampilkan variabel ini di browser.
- Jika cookienya salah maka ada pemberitahuan tidak berhak menggunakan aplikasi ini tanpa login terlebih dahulu.

4.1.3.7 Letak file konfigurasi Samba

Untuk merubah path file konfigurasi atau utilitas-utilitas Samba dapat dilakukan disini. Juga dilakukan pengecekan apakah file tersebut ada atau tidak. Jika file berdasarkan path tersebut tidak ada maka kotak input pada file tersebut akan kosong. Otomatisasi jika samba di *uninstall* maka semua kotak input path ini akan kosong kecuali path file `/etc/smbpasswd`.

Algoritma untuk letak file konfigurasi Samba sebagai berikut :

- Mengecek cookie, jika benar maka akan dilakukan proses-proses berikut :
 - ✓ Menyimpan path masing-masing file konfigurasi dan utilitas Samba ke suatu file teks. File konfigurasi dan utilitas yang

dimaksud adalah : smbconf, smbpasswd, file password smbpasswd, smbclient, smbstatus, smb, testparm dan testprns.

- ✓ Melakukan pengecekan terhadap path yang diberikan untuk masing-masing file tersebut. Jika file itu ada maka path tersebut akan ditampilkan pada kotak input, jika file tersebut tidak ada maka kotak input tersebut akan kosong.
- Jika cookienya salah maka ada pemberitahuan tidak berhak menggunakan aplikasi ini tanpa login terlebih dahulu.

4.1.3.8 Menangani Password Samba

Tidak menggunakan seluruh opsi utilitas smbpasswd yang ada. Jadi hanya menangani password lokal (server) dan merubah password user di NT.

- **Password server**

Algoritma menangani password server :

- Secara umum didahului dengan mengecek cookie, jika benar maka dilanjutkan dengan proses berikut : (sesuai keinginan user)
 - ✓ Merubah password
 - Mengecek apakah nama user dan kedua password yang dimasukkan melalui kotak input sudah benar
 - Jika pengecekannya sudah benar lalu menjalankan utilitas **smbpasswd nama_user password**
 - Pemberitahuan kepada user apakah perubahan password sudah berhasil atau gagal.

✓ Menambah User

- Mengecek apakah nama user dan kedua password yang dimasukkan melalui kotak input sudah benar
- Jika pengecekannya sudah benar lalu menjalankan utilitas **smbpasswd -a nama_user password**
- Jika nama_user sudah ada di file smbpasswd maka yang dilakukan adalah merubah password user Samba.
- Pemberitahuan kepada user apakah penambahan user sudah berhasil atau gagal.

✓ Menghapus User

- Mengecek apakah nama user sudah diisi pada kotak input nama user.
- Jika pengecekannya sudah benar lalu menjalankan utilitas **smbpasswd -x nama_user**
- Pemberitahuan kepada user apakah penghapusan user Samba sudah berhasil atau gagal.

✓ Mendisable User

- Mengecek apakah nama user sudah diisi pada kotak input
- Jika pengecekannya sudah benar lalu menjalankan utilitas **smbpasswd -d nama_user**

- Pemberitahuan kepada user apakah user sudah didisable atau gagal.
- ✓ Mengineable User
 - Mengecek apakah nama user sudah diisi pada kotak input
 - Jika pengecekannya sudah benar lalu menjalankan utilitas **smbpasswd -e nama_user**
 - Pemberitahuan kepada user apakah user sudah dienable atau gagal.
- **Password *client server***

Admin dapat merubah password *user* samba yang terdaftar di NT. NT harus menjadi PDC. Menggunakan opsi : **smbpasswd -r nama_mesin -U user**

4.1.3.9 Penanganan Masalah

Penulis disini memberikan beberapa artikel *troubleshooting Samba* dalam bahasa Indonesia. Untuk memecahkan masalah Samba yang sering ditemui.

4.1.3.10 Logout/Keluar dari Aplikasi

Ketika user logout atau keluar dari aplikasi ini, maka terdapat beberapa proses yaitu menghapus cookie yang dibuat saat user tersebut login.

4.2 Pembuatan Perangkat Lunak

Berdasarkan hasil perancangan tadi dilanjutkan dengan pembuatan perangkat lunak.

4.2.1 Otentikasi Nama User dan Password

- Meminta input nama user dan password (html) :

```
<input type="text"      name="nama"  size="13" value="">  
<input type="password" name="pass"  size="13" value="">
```

Gambar 4.1 Input nama user dan password

- Membandingkan inputan nama user dan password dengan yang terdapat di /etc/shadow :

```
sub checklogin {  
  my ($nama_s, $pass_s);  
  my $file_shadow='/etc/shadow';  
  my $password_benar =1;  
  
  //membuka file /etc/shadow  
  open (fd,$file_shadow) or print ("file shadow tidak bisa dibuka");  
  while (defined(my $baris=<fd>)){  
    ($nama_s, $pass_s)=(split(/:/,$baris))[0,1];  
    last if ($nama_s eq $nama);  
  }  
  close(fd);  
  if (($nama_s ne $nama) or (crypt($pass,$pass_s) ne $pass_s)) {  
    $password_benar =0;  
  }  
  return $password_benar;  
}
```

Gambar 4.2 Membandingkan nama user dan password dengan file /etc/shadow

4.2.2 Security

- Memberikan cookie hanya pada saat login pada .cgi

```
use CGI::Cookie;

my $cook = new CGI::Cookie(-name=>'rootcookie',
                           -value=>'oreo',
                           -expires=>'+30m',
                           -path=>'/' );
```

Gambar 4.3 Memberikan Cookie

- Mengecek cookie pada file .php

```
<? if ($rootcookie!="oreo") {
    header("Location: nocookie.html");
    exit;
}
header ("Pragma: no-cache");
?>
```

Gambar 4.4 Mengecek Cookie pada file .php

- Mengecek cookie pada file .cgi

```
use CGI::Cookie;
my $cgi= new CGI_Lite;
$cook = $cgi->parse_cookies;
$cookie = $cook->{rootcookie} ;

if (($cookie eq "") ||($cookie eq "oreo")){
# cookienya benar
    print "Content-type: text/html\n\n";
    ...coding...
}
#cookienya salah
else {
    print ("Location:../nocookie.html\n");
    print ("Content type:text\html\n\n");
}
}
```

Gambar 4.5 Mengecek Cookie pada file .cgi

4.2.3 Mengedit File smb.conf

- Membaca dan merapikan file smb.conf, kemudian disimpan dalam suatu variabel yang nanti akan digunakan oleh program yang lain.

Variabel ini bernama \$filesmbconf

```
<?
//mengecek file keberadaan file smb.conf,
//$namafilename = path file smb.conf

if (!file_exists($namafilename)) {
    header("Location:../no_path.html");
}
else {
    $temp=file($namafilename);
    for ($i=0; $i<count($temp); $i++) {

        Disini dilakukan proses sebagai berikut :
        ▪ penghapusan komentar # dan ;
        ▪ pengecekan nama-nama share agar sesuai acuan SWAT
        ▪ pengaturan tampilan
    }

    //memanggil fungsi untuk menuliskan ke file smb.conf
    proses_tulis($filesmbconf,$namafilename,"w");
}
?>
```

Gambar 4.6 Membaca dan merapikan file smb.conf

4.2.3.1 Bagian Global

- **Membaca bagian global**

```
<?
function find_section($namakop) {
global $filesmbconf;
if (preg_match_all("/\[ $namakop \] [\+|\^|\!|\<|\>|\?|\{|\}|\\#|\s|
\\w|\=|\@|\%|\.|\\/|\-|\,|\*|\||\|$\|~|\(|\)|\;|\:|\\"|\'|\\&]*\/i",
$filesmbconf,$lhasil))
    $hasil=implode("",$lhasil[0]);
return $hasil;
}
?>
```

Gambar 4.7 Fungsi untuk membaca bagian global/share/printer

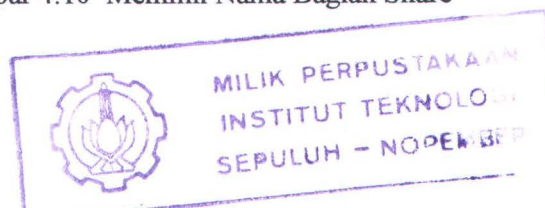
Diawali dengan memilah antara bagian share terhadap printer dan global. Untuk mendapatkan bagian share, tekniknya adalah mencari bagian yang tidak mengandung parameter *printable=yes*.

```

<?
function pilih() {
global $filesmbconf;
$caption="[ \t\\w\\[\\#\\|;\\+\\^\\!\\<\\>\\?\\{\\}\\|=
\\@\\%\\.\\|/\\-\\.\\|*\\|\\|\\$\\|~\\(|\\;\\|\\|\\|'\\|&\\|+";
$pattern_all_kop="/\\[$caption\\]/i";
preg_match_all("$pattern_all_kop",$filesmbconf,$hasil1);
$semua_kop=implode("",$hasil1[0]);
$pattern_detail_printer="/\\[$caption\\][\\(|\\|\\-
\\|\\|\\|\\|\\|*\\|&\\|\\$\\|%\\|^\\|@\\|!\\|~\\|?\\|<\\|>\\|,\\|;
\\|/\\|`\\|'\\|\"\\|\\:|\\{\\|}\\|%\\|\\.\\|#\\|\\w\\|\\t\\|\\s\\|\\+\\|=| ]*printable[
\\t\\|*]=[ \\t\\|*yes\\(\\(|\\|\\-\\|\\|\\|\\|\\|\\|*\\|&\\|\\$\\|%\\|^\\|@\\|!\\|~\\|?\\|
\\|<\\|>\\|\\|,\\|;\\|/\\|`\\|'\\|\"\\|\\:|\\{\\|}\\|%\\|\\.\\|#\\|\\w\\|\\t\\|\\s\\|\\+\\|=| ]*/i
";
preg_match_all("$pattern_detail_printer",$filesmbconf,$hasil2);
$semua_printer=implode("",$hasil2[0]);
$pattern_kop_printer="/\\[$caption\\]/i";
preg_match_all("$pattern_kop_printer",$semua_printer,$hasil3);
$semua_kop_printer = implode("",$hasil3[0]);
$semua_kop_printer = tr_replace("["","]"&["",$semua_kop_printer);
$split_printer=split("&",$semua_kop_printer);
$semua_kop=str_replace("[global]","","",$semua_kop);
$campur=$semua_kop;
$semua_kop=str_replace("["","]"&["",$semua_kop);
$split_all=split("&",$semua_kop);
for ($i=0;$i<count($split_printer);$i++){
    for ($j=0;$j<count($split_all);$j++){
        if ($split_printer[$i]==$split_all[$j]){
            $campur=str_replace($split_printer[$i],"",$campur);
        }
    }
} //akhir for
$share=$campur;
$share=str_replace("["","]"&&cobatebak&&["",$share);
$share=str_replace("&","",$share);
$share=str_replace("&["","]","",$share);
$share=str_replace("["","]","",$share);
$share=split("&cobatebak&",$share);
for ($i=0;$i<count($share); $i++){
    $share[$i]=preg_replace("/^\\[/","",$share[$i]); }
$share=implode("cobatebak",$share);
$share=split("cobatebak",$share);
return $share;
}
?>

```

Gambar 4.10 Memilih Nama Bagian Share



Selanjutnya sama seperti membaca bagian global yaitu dengan memanggil `find_section()` dengan melewati nama bagian share yang diinginkan.

- **Membaca Parameter Bagian Share**

Tekniknya sama dengan membaca parameter bagian global, yaitu memanggil fungsi `find_param()`, dengan melewati nama parameter dan nama bagian share.

- **Menghapus Bagian Share**

Fasilitas ini digunakan jika admin ingin menghapus suatu bagian share. Dibuatkan fungsi `hapus_section()`, dengan melewati variabel berisi nama bagian share (`$kop`). Dengan fungsi RE `preg_match_all()`, akan mencari satu bagian share/printer di variabel yang berisi file `smb.conf`. Kemudian satu bagian share/printer tadi akan dihapus dengan fungsi `str_replace()`.

```
function hapus_section($kop) {
global $filesmbconf;
if (preg_match_all("/\[ $kop\] [\+|\^|\!|\<|\>|\?|\{|\}|\#|\s|\w|\
=|\@|\%|\.|\/|\-|\,|\*|\||\$\|~|\(|\)|\:\|\"|\'|\&]*\/i",
$filesmbconf,$lhasil))
$temp=implode("",$lhasil[0]);
if ($temp!="") {
//share difile kita ganti dengan ""
$temp=str_replace("$temp","",$filesmbconf);
$bukafile=fopen('smb.conf','w+');
fputs($bukafile,$temp);
fclose($bukafile);
}
}
?>
```

Gambar 4.11 Menghapus bagian share/printer

- **Membuat Bagian Share Baru**

Digunakan jika admin ingin membuat suatu bagian share baru, tekniknya yaitu menuliskan [nama_bagian_share] diikuti parameter *path=/tmp* ke file *smb.conf*.

```
<?
  $nama_bagian="[".$bagian_baru."]."\n"."  "." path
=/tmp". "\n";
  $bukafile=fopen('/etc/smb.conf','a');
  fputs($bukafile,$kop);
  fclose($bukafile);
?>
```

Gambar 4.12 Membuat bagian share baru

- **Menuliskan bagian share**

Tekniknya hampir sama dengan menuliskan bagian global.

4.2.3.3 Bagian Bagian Printer

- **Membaca Bagian Printer**

Langkahnya sama dengan membaca bagian share, yaitu harus mencari nama bagian printer terlebih dahulu, yang mengandung parameter *printable=yes*. Dibuatkan fungsi dengan nama fungsi *pilih()*; Perhatikan gambar 4.13. Selanjutnya memanggil *find_section()*, dengan nama bagian printer yang didapat dari fungsi *pilih()* ini.

- **Membuat Bagian Printer Baru**

Tekniknya yaitu menuliskan `[nama_share_printer]` diikuti dengan parameter `path=/tmp` dan `printable=yes`. Perhatikan gambar 4.14

- **Membaca Parameter Bagian Printer**

Langkahnya sama dengan membaca parameter bagian share.

```
<?
function pilih(){
global $filesmbconf;
$caption="[
|\t|\w|\\[|\#|\;|\+|\^|\!|\<|\>|\?|\{|\}|\|=|\@|\%|\.\|\/|\-
|\,|\*|\||\$|\|~|\(|\;|\) |\:|\\"|\'|\&|\+";

$pattern_detail_printer="/\[ $caption\][\(|\)|\|
|\\|\/|\|*|\&|\$|\%|\^|\@|\!|\~|\?|\<|\>|\,|\;|\|\/|\\"|\'|\\"|\:|\{
|\}|\%|\.\|\/|\#|\w|\t|\s|\+|\|=| ]*printable[ |\t]*=[
|\t]*yes[\(|\)|\|
|\\|\/|\|*|\&|\$|\%|\^|\@|\!|\~|\?|\<|\>|\,|\;|\|\/|\\"|\'|\\"|\:|\{
|\}|\%|\.\|\/|\#|\w|\t|\s|\+|\|=|]*\/i";
preg_match_all("$pattern_detail_printer", $filesmbconf, $hasil);
$semua_printing=implode("", $hasil[0]);
preg_match_all("/\[ $caption\]/i", $semua_printing, $hasil1);
$semua_kop_printer= implode("", $hasil1[0]);
$printer=str_replace("[", "&cobatebak&&[", $semua_kop_printer);
$printer=split("&cobatebak&&", $printer);

for ($i=0;$i<count($printer);$i++) {
    $printer1=preg_replace("/^\[/", "", $printer[$i]);
    $result[$i]=str_replace("[", "", $printer1);
}
$printer=implode("&cobatebak&&", $result);
$printer=split("&cobatebak&&", $printer);
return $printer;
}
?>
```

Gambar 4.13 Memilih Nama Bagian Printer

```
<?
$kop="\n"."[".$printerbaru."]".
"\n"."      "."path=/tmp"."      "."printable=yes"." \n";
$bukafile=fopen('smb.conf', 'a');
fputs($bukafile, $kop);
fclose($bukafile);
?>
```

Gambar 4.14 Membuat Bagian Printer Baru

- **Menghapus Bagian Printer**

Langkahnya sama dengan menghapus bagian share.

- **Menulis Bagian Printer**

Langkahnya sama dengan menulis bagian share.

4.2.4 Menjalankan Daemon Samba

Dilanjutkan dengan mengimplementasikan rancangan menjalankan daemon Samba.

- **Testparm**

```
#!/usr/bin/perl
$ENV{PATH}=$ENV{BASHENV}="";$<=$>;$(=);
use CGI_Lite;use CGI qw/:standard/;
use CGI::Cookie;use readconfig;

my $cgi= new CGI_Lite;
$cook = $cgi->parse_cookies;
$cookie = $cook->{rootcookie} ;

if (($cookie eq "")||($cookie eq "oreo")) {
    if ($cookie eq "oreo"){
        print "Content-type: text/html\n\n";
    }
    $testparm = readconfig::baca_config("testparm");
    $sambaconf = readconfig::baca_config("smbconf") ;
    $hasil= system ("$testparm","$sambaconf");

    if ($hasil)
    { print "Gagal menjalankan testparm ! \n";
      print "Kemungkinan Samba belum terinstall atau path
testparm tidak ditemukan.\n";
      print "Periksa kembali path testparm di menu Letak File
Konfigurasi Samba";
    }

}
else
{
    print ("Location: ../nocookie.html\n");
    print ("Content type:text/html\n\n");
}
}
```

Gambar 4.15 Menjalankan utilits testparm

- Testprns

```
#!/usr/bin/perl -U
$ENV{PATH}=$ENV{BASHENV}="";$<=$>;$ (= $) ;

use CGI_Lite;
use CGI qw/:standard/;
use CGI::Cookie;
use readconfig;

my $cgi= new CGI_Lite;
$ccook = $cgi->parse_cookies;
$cookie = $ccook->{rootcookie} ;

if (($cookie eq "") ||($cookie eq "oreo")) {
    if ($cookie eq "oreo"){
        print "Content-type: text/html\n\n";
    }

    $testprns = readconfig::baca_config("testprns");
    $namaprinter = @ARGV[0];
    $hasil= system("$testprns $namaprinter");

    if ($hasil)
        { print "Gagal menjalankan testprns ! \n";
          print "Kemungkinan Samba belum terinstall atau path
testprns tidak ditemukan.\n";
          print "Periksa kembali path testprns di menu Letak
File Konfigurasi Samba";
        }

}
else
{
    print ("Location: ../nocookie.html\n");
    print ("Content type:text/html\n\n");
}
}
```

Gambar 4.16 Menjalankan utilitas testprns

Utilitas testprns ini bertujuan untuk memvalidasi printer oleh smbd. Jika validasi ini gagal maka akan muncul konfirmasi bahwa telah terjadi kegagalan dengan kemungkinan penyebabnya.

- **Restart *smbd***

```
#!/usr/bin/perl -U
$ENV{PATH}=$ENV{BASHENV}="";$<=$>;$(=);

use CGI_Lite;
use CGI qw/:standard/;
use CGI::Cookie;
use readconfig;

my $cgi= new CGI_Lite;
$cook = $cgi->parse_cookies;
$cookie = $cook->{rootcookie} ;

if (($cookie eq "") ||($cookie eq "oreo")){
    if ($cookie eq "oreo") {
        print "Content-type: text/html\n\n";      }
        $smb = readconfig::baca_config("smb");
        $hasil= system("$smb restart");

if ($hasil)
    { print "Gagal merestart daemon smb ! \n";
      print "Kemungkinan Samba belum terinstall atau path
daemon smb tidak ditemukan\n";
      print "Periksa kembali path daemon smb dii menu Letak
File Konfigurasi Samba"; }
    }

else {
    print ("Location: ../nocookie.html\n");
    print ("Content type:text\html\n\n");
}
}
```

Gambar 4.17 Menjalankan utilits restart smb

Untuk mengaktifkan semua perubahan yang dilakukan pada bagian global, share dan printer maka daemon *smbd* dan *nmbd* harus diaktifkan. Jika terjadi kegagalan akan muncul di browser pemberitahuannya.

- **Smbclient -L localhost**

```
#!/usr/bin/perl -U

use CGI_Lite;
use CGI qw/:standard/;
use CGI::Cookie;
use readconfig;

my $cgi= new CGI_Lite;
$cook = $cgi->parse_cookies;
$cookie = $cook->{rootcookie} ;

if (($cookie eq "") ||($cookie eq "oreo")) {
    if ($cookie eq "oreo"){
        print "Content-type: text/html\n\n";
        $ENV{PATH}=$ENV{BASHENV}="";$<=$>;$ (=);
        $smbclient = readconfig::baca_config("smbclient");
        $hasil= system("$smbclient -L localhost");

        if ($hasil)
            { print "Gagal menjalankan smbclient ! \n";
              print "Kemungkinan Samba belum terinstall atau path
smbclient tidak ditemukan.\n";
              print "Periksa kembali path smbclient di menu Letak
File Konfigurasi Samba";
            }
    }

else
{
    print ("Location: ../nocookie.html\n");
    print "Content-type: text/html\n\n";
}
}
```

Gambr 4.18 Menjalankan utilitas smbclient -L localhost

4.2.5 Menginformasikan Status Koneksi Samba

Utilitas *smbstatus* dapat digunakan untuk mendapatkan informasi koneksi Samba. Penulis menjalankan *smbstatus* dan menyimpan hasilnya di suatu file. File ini kemudian diolah untuk mendapatkan informasi selanjutnya.

- Mengolah file status.txt dan menampilkan informasinya di browser

```
<?
if ($rootcookie!="oreo")
{
header("Location: nocookie.html");
exit;
}

$run=`cgi/status.pl`;
if ($run=="error"){
header("Location: no_path.html");
exit;
}
header ("Pragma: no-cache");
?>
```

```
<html>
<body>
<?
```

Setelah file status.txt dimanipulasi maka informasi yang diperoleh ditampilkan disini. Masing-masing item dalam tabel tersendiri.

- Koneksi Aktif
Berisikan informasi PID, *Client*, Alamat IP, Tanggal.
PID ini nanti berupa link, dimana kalau user meng-klik link ini maka koneksi dengan client tersebut akan terputus.
- Share aktif
Berisikan informasi Nama *Share*, *User*, Group, PID, *Client*, Tanggal.
- File Terbuka
Berisikan informasi PID, Sharing, R/W, oplock, File, Tanggal.
Juga mengatasi nama file yang berisi spasi.

```
?>
</body>
</html>
```

Gambar 4.20 Menampilkan informasi di browser

Pada koneksi aktif diatas terdapat link yang jika di klik akan memutuskan koneksi dengan client saat itu juga. Juga sudah menangani nama file yang berisi spasi.

4.2.6 Melihat Konfigurasi Samba

```
<?
//menggunakan variabel file smb.conf
include ("inc/filesmbconf.inc");

//mengecek cookie
if ($rootcookie!="oreo")
{   header("Location: nocookie.html");
    exit;
}
header ("Pragma: no-cache");
?>

<html>
<body>
<?
//menampilkan variabel file smb.conf
Echo("<pre>$filesmbconf</pre>");
?>
</body>
<html>
```

Gambar 4.21 Melihat Konfigurasi Samba

Apabila user mengedit file `smb.conf` melalui editor teks biasa dan tidak mengurus rapi tidaknya susunan parameter juga tidak memperhatikan persyaratan suatu share maka dengan menggunakan menu melihat konfigurasi samba ini semuanya sudah diatasi.

Sehingga user akan melihat file `smb.conf` yang rapi, dan yang terpenting semua parameter dan share sudah valid.

4.2.7 Menangani Password Samba

Ada dua item password Samba yang ditangani yaitu password *server* dan password *client-server*.

- **Password Server**

Admin mampu untuk merubah password user Samba, menambah, menghapus, mendisable dan mengenable user samba.

- Mengecek cookie terlebih dahulu

```
use CGI::Cookie;

$cook = $cgi->parse_cookies;
$cookie = $cook->{rootcookie} ;

if (($cookie eq "") || ($cookie eq "oreo")) {
    print "Content-type: text/html\n\n";

    ...code...
    //penanganan password server mulai disini, baik itu
    menambah, merubah password, menghapus, mendisable dan
    mengenable user.
    ...code...
}
else
{
    print ("Location: ../nocookie.html\n");
    print ("Content type:text\html\n\n");
}
```

Gambar 4.22 Mengecek cookie

- Input nama user dan password

```
<input type="text"      name="nama"  size="35" >
<input type="password" name="pass1" size="35" >
<input type="password" name="pass2" size="35" >
```

Gambar 4.23 Input nama user dan password

- Merubah password user Samba

```
#merubah password user
if ($submit eq "Rubah Password"){
    if ($namal eq ""){
        $isi="<h3>Input nama user harus diisi</h3>";
    }
    elsif ((($pass1 || $pass2) eq "")|| ($pass1 ne $pass2)){
        $isi="<h3>Kedua password harus sama</h3>";
    }
    else {

        if (system("$smbpasswd $namal $pass1 >
/swatindo/smbpasswd.txt")ne 0){
$isi="<h3>Gagal menemukan user <fontcolor=red>$namal</font>
<br>Gagal untuk memodifikasi password $namal</h3>";
        }
        else { $isi="<h3>Password user
<font color=red>$namal</font> sudah dirubah</h3>"; }
    }
}
$pageasal->param('status','$isi');
print $pageasal->output;
}
```

Gambar 4.24 Merubah password user samba

- Menghapus user Samba

```
#menghapus user
elseif
($submit eq "Menghapus User") {
    if ($namal eq ''){
        $isi="<h3>Input nama user harus diisi</h3>";
    }
    else {
        if (system("$smbpasswd -x $namal >
/swatindo/smbpasswd.txt")ne 0){
            $isi="<h3>Gagal menghapus user <font
color=red>$namal</font></h3>";
        }
        else {
            $isi="<h3>User <font color=red>$namal</font>
sudah terhapus";
        }
    }
}
$pageasal->param('status','$isi');
print $pageasal->output;
}
```

Gambar 4.25 Menghapus user samba

- Menambah user Samba

```
if($submit eq "Menambah User Baru") {
if ($namal eq ''){
    $isi="<h3>Input nama user harus diisi</h3>";}
elseif (($pass1 || $pass2) eq "") {
    $isi="<h3>Password tidak boleh kosong</h3>"; }
elseif ($pass1 ne $pass2){
    $isi="<h3>Kedua password harus sesuai</h3>"; }
else {
    if ((&checkuser()) && (system("$smbpasswd -a $namal
$pass1 > /swatindo/smbpasswd.txt") eq 0)) {
        $isi="<h3>Password user <font color=red>$namal</font>
sudah dirubah</h3>";
    }
    elseif ((&checkuser() eq 0) && (system("$smbpasswd -a $namal
$pass1 > /swatindo/smbpasswd.txt") eq 0)) {
        $isi="<h3>User <font color=red>$namal</font>
sudah ditambahkan</h3>"; }
    elseif (system("$smbpasswd -a $namal $pass1 >
/swatindo/smbpasswd.txt") ne 0){
        $isi="<strong>User <font color=red>$namal</font> tidak
terdapat di file password sistem. Tidak dapat menambahkan
account tanpa user sistem lokal yang valid</strong>"; }
    }
    $pageasal->param('status',"$isi");
    print $pageasal->output;
}
```

Gambar 4.26 Menambah user Samba

- Mendisable user samba

```
If ($submit eq "Mendisable User") {
    if ($namal eq ''){
        $isi="<h3>Input nama user harus diisi</h3>"; }
    else {
        if (system("$smbpasswd -d $namal>
/swatindo/smbpasswd.txt")ne 0){
            $isi="<h3>Gagal mendisable user <font
color=red>$namal</font></h3>";
        }
        else {
            $isi="<h3>User <font color=red>$namal</font> sudah di
'disable'</h3>"; }
        }
        $pageasal->param('status',"$isi");
        print $pageasal->output;
    }
```

Gambar 4.27 Mendisable user Samba

– Mengenable user Samba

```
elseif
  ($submit eq "Mengenable User") {
  if ($namal eq ''){
    $isi="<h3>Input nama user harus diisi</h3>";
  }
  else {
    if (system("$smbpasswd -e $namal >
/swatindo/smbpasswd.txt")ne 0){
      $isi="<h3>Gagal mengenable user <font
color=red>$namal</font></h3>";
    }
    else {
      $isi="<h3>User <font color=red>$namal</font> sudah di
'enable'</h3>"; }
    }
    $pageasal->param('status',"$isi");
    print $pageasal->output;
  }
}
```

Gambar 4.28 Mengenable user Samba

• Password Client-Server

Admin mampu untuk merubah password user Samba yang terdaftar di Windows NT Server. NT server harus sebagai PDC. Perhatikan gambar 4.29.

- Mengecek cookie
- Input nama user, password, *remote machine*

```
<input type="text"      size="30" name="user_nt" >
<input type="password" size="30" name="pass_nt" >
<input type="password" size="30" name="pass_nt_1">
<input type="password" size="30" name="pass_nt_2">
<input type="text"     size="30" name="remotemesin">
```

Gambar 4.29 Input nama user, password dan remote machine

- Merubah password user samba di windows NT

Semua input password diatas dimasukkan ke suatu file yang akan digunakan sebagai inputan pada utilitas smbpasswd.

```
//password disimpan di file
$sisifile=$pass_nt."\n".$pass_nt_1."\n".$pass_nt_2."\n";
open(fd,"> ../file_password.txt");
printf (fd $sisifile);
close(fd);

if ($user_nt"){
    $sisi="<strong>Nama user harus diisi</strong>";
}
elseif ((($pass_nt_1 || $pass_nt_2)eq "")||($pass_nt_1 ne
$pass_nt_2)){
    $sisi="<strong>Password tidak boleh kosong dan harus
sama</strong>"
}
elseif (system("/usr/bin/smbpasswd -s -r $mesin -U $user_nt <
file_password.txt") ne 0){
    $sisi="<strong>Gagal merubah password user <font
color=red>$namal</font></strong>";
}
else {
    $sisi="<strong>Password untuk user <font
color=red>$namal</font> sudah dirubah</strong>";
}
```

Gambar 4.30 Merubah password user Samba di Windows NT

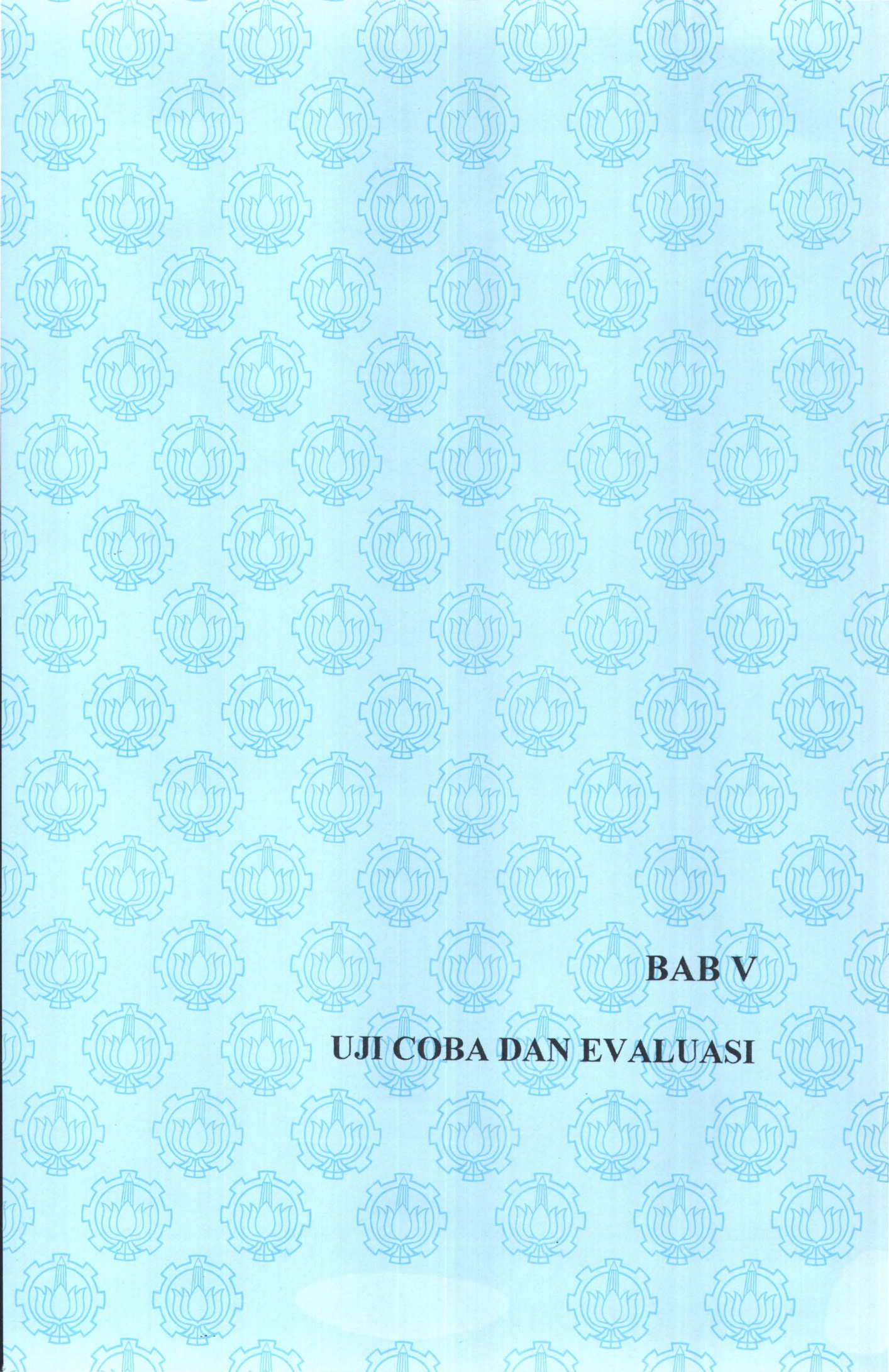
4.2.8 Logout / Keluar dari Aplikasi

- Ketika user keluar dari aplikasi cookie yang dibuat saat login dirubah valuenya dengan waktu random. Sebenarnya cookie ini dihapus, tetapi ternyata ada masalah dengan script cgi kalau cookienya dihapus sehubungan dengan pengecekan kondisi tertentu menggunakan cookie.

```
<?
if (isset($rootcookie))
{
    setcookie("rootcookie","",time(),"/");
}
?>

<html>
<body>
<h2>
Terima Kasih.
Anda sudah keluar dari aplikasi SWATINDO
</h2>
</hml>
</body>
```

Gambar 4.31 Logout/Keluar dari Swatindo



BAB V

UJI COBA DAN EVALUASI

BAB V

UJI COBA DAN ANALISA

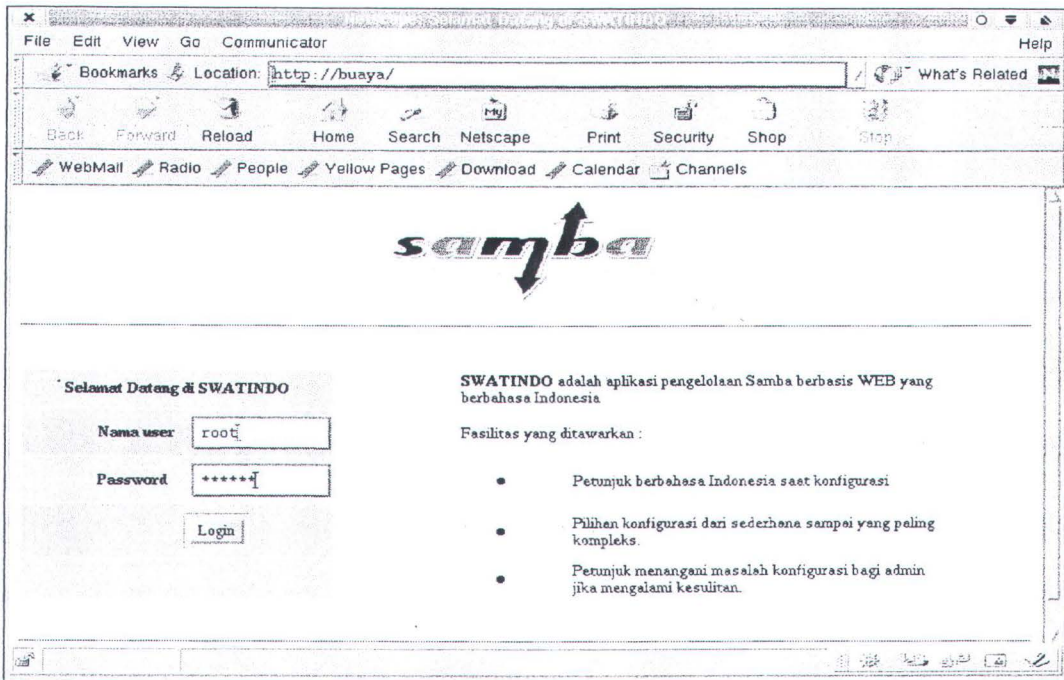
Pada bab ini akan dibahas mengenai uji coba dan evaluasi yang dilakukan pada sistem yang telah dibuat.

5.1 Uji Coba

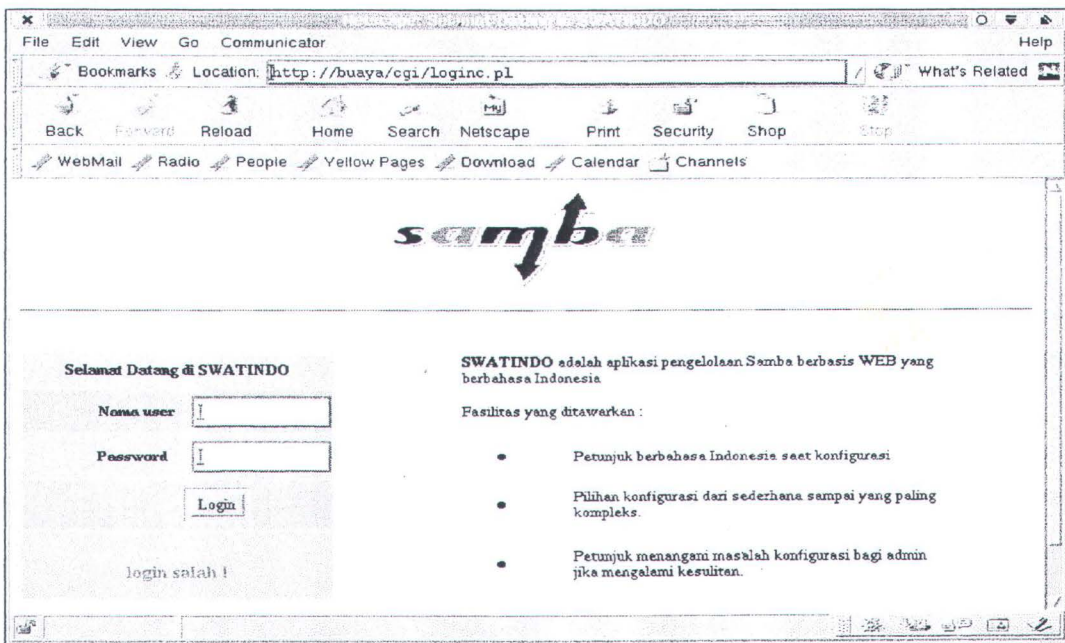
Uji coba dilakukan dengan menggunakan tiga komputer. Komputer A dengan sistem operasi Linux dan komputer B dengan sistem operasi Windows 95 dan komputer C dengan Windows NT Server.

5.1.1 Uji Coba Otentikasi Nama User dan Password

User harus login dengan login root dan passwordnya adalah password root sistem. Bila loginnya salah akan muncul halaman yang sama dengan pesan login salah. Password yang dilewatkan masih berupa plain teks, tidak dienkripsi sehingga akan menjadi *lubang security*. Tetapi nama user dan password ini tidak akan terlihat saat user menekan tombol *back* browser (jika sudah berada di halaman lain). Pemberian nilai pada *cookie* juga dilakukan saat ini. Penggunaan *cookie* ini merupakan upaya di sisi sekuriti selain otentikasi dan pemberian *header pragma no-cache* pada halaman php. Perhatikan gambar 5.1 untuk tampilan otentikasi nama user dan password dan gambar 5.1 memperlihatkan halaman login yang berisi pesan login salah.



Gambar 5.1 Tampilan otentikasi nama user dan password

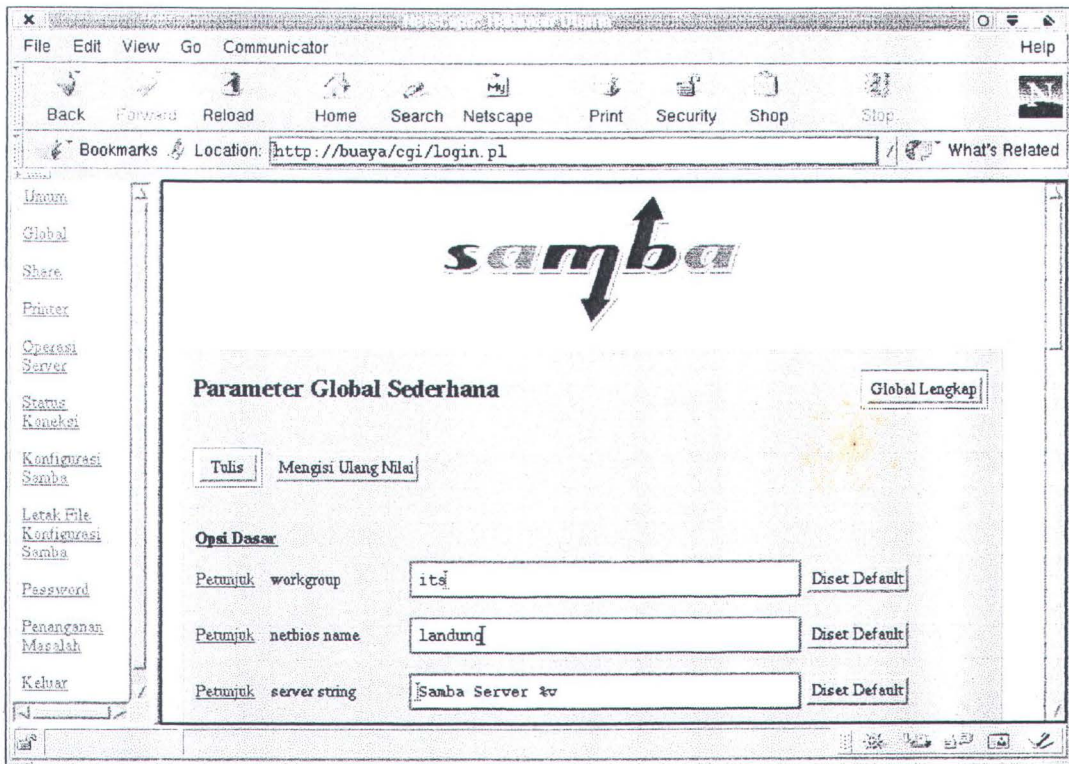


Gambar 5.2 Tampilan login yang gagal

5.1.2 Mengedit Bagian Global

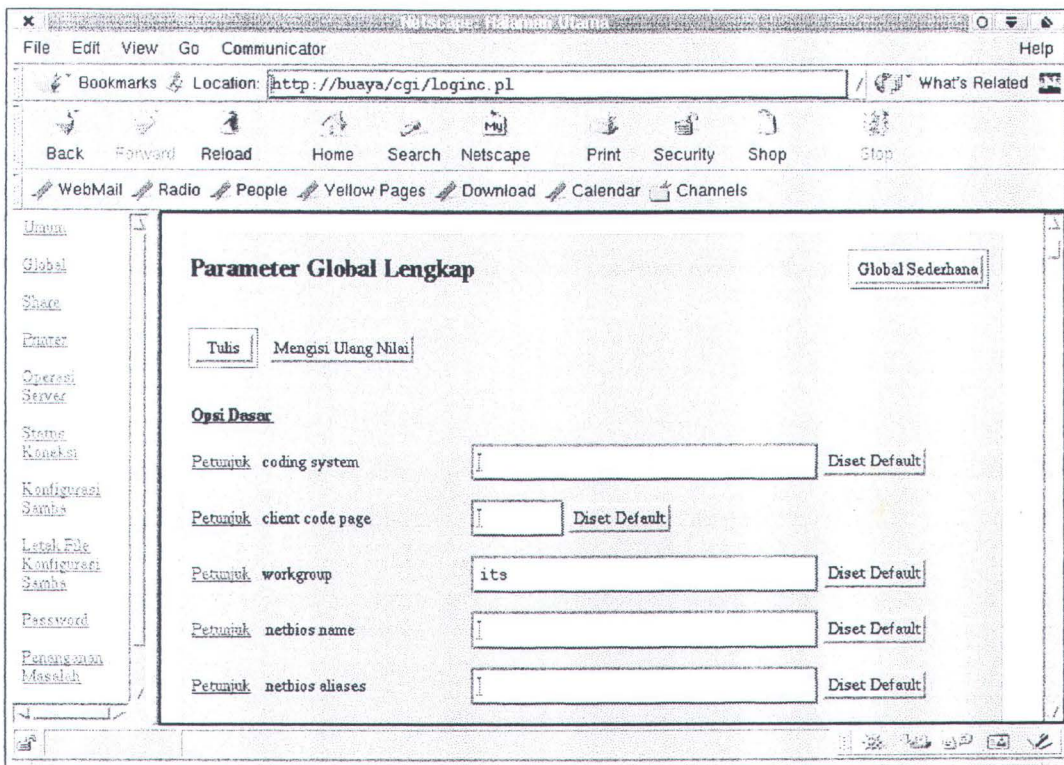
Setelah login user dapat melakukan pengeditan terhadap file smb.conf, melihat status koneksi, melihat konfigurasi smb.conf, mengelola password user samba atau membaca artikel mengenai *trouble shooting* masalah Samba.

Berikut interface untuk mengedit bagian global sederhana. Dinamakan global sederhana karena parameter yang disediakan tidak semua parameter global. Otomatis konfigurasi yang dihasilkan masih sederhana/standar. Tapi sudah cukup untuk membuat koneksi yang aman. Untuk mendapatkan parameter global yang lengkap user dapat berpindah dengan mengklik tombol **Global Lengkap**.



Gambar 5.3 Editor global sederhana

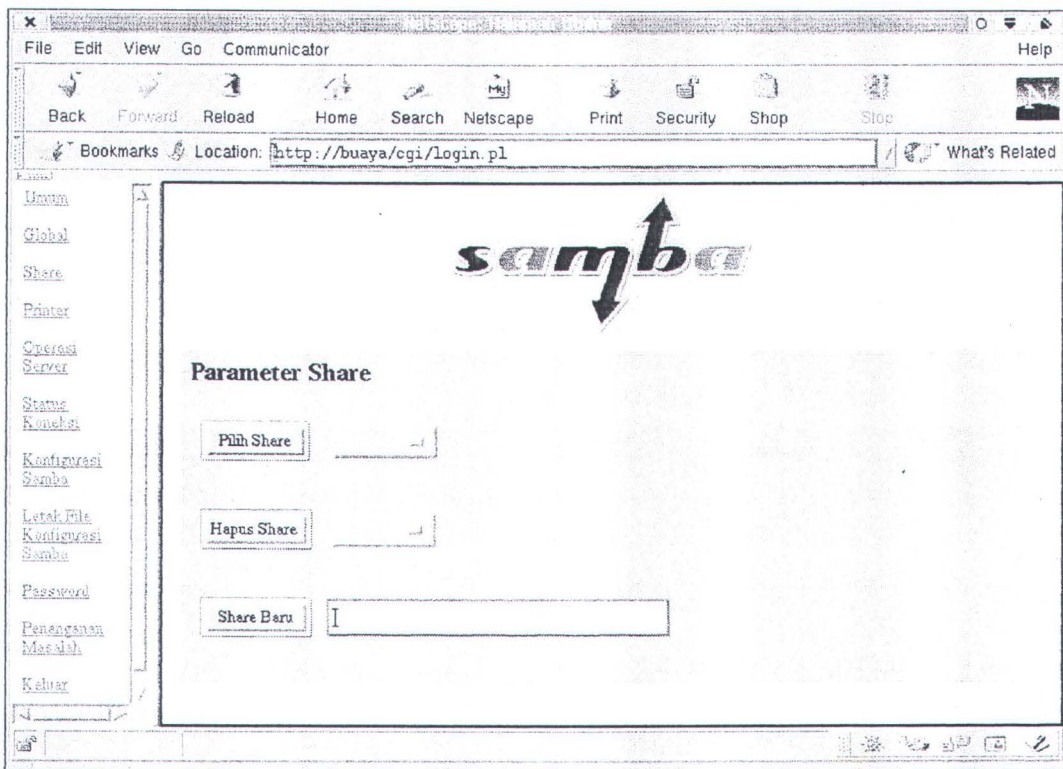
Diatas terlihat link petunjuk dan tombol Diset Default. Jika user mengalami kesulitan memahami parameter global dapat mengklik link tersebut dan akan langsung menemukan jawabannya. Jika user bingung dengan nilai parameter yang harus diisi, untuk amannya dapat menggunakan nilai default dengan menekan tombol Diset Default. Nilai yang terlihat pada kotak input adalah nilai parameter yang benar-benar ada pada file smb.conf, tidak seperti SWAT yang kurang akurat dalam hal pembacaan nilai parameter. Jadi sering terjadi nilai parameter yang tidak ada ditulis juga dikotak input sehingga akan membuat bingung administrator.



Gambar 5.3 Editor global lengkap

5.1.3 Mengedit Bagian Share

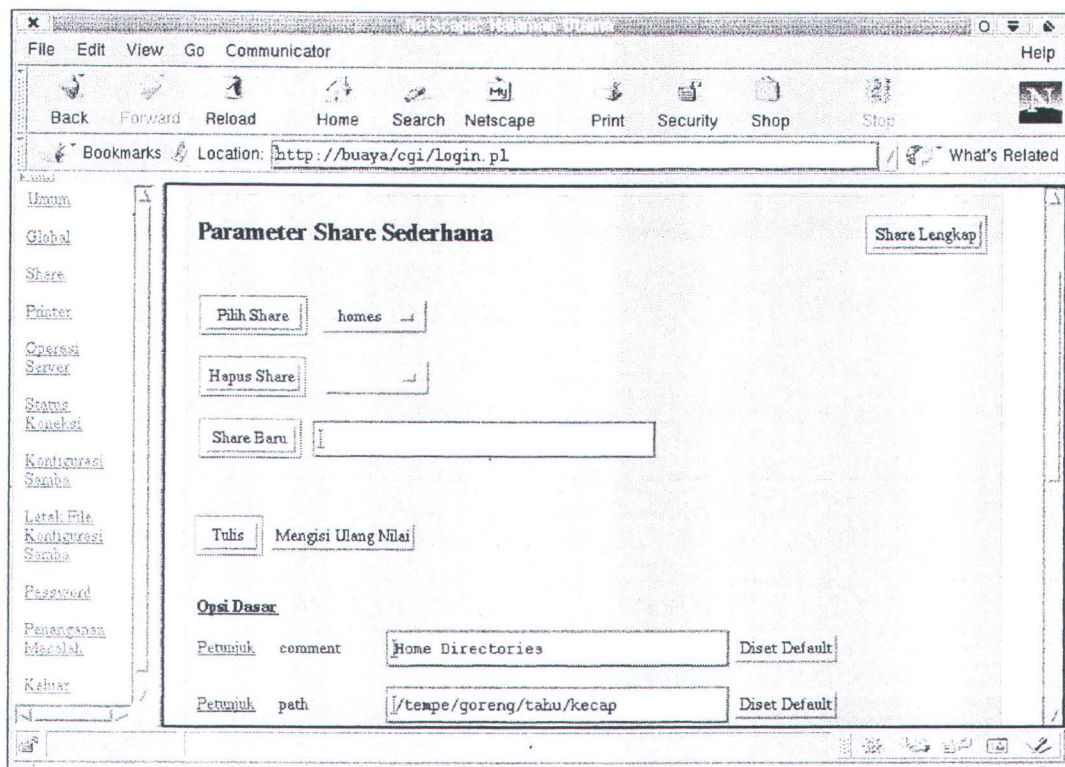
Menu mengedit bagian share ini diawali dengan interface yang terdiri atas tiga sub menu yaitu memilih, menghapus share dan membuat share baru. Sub menu ini juga ada pada editor share sederhana dan share lengkap. Untuk pembuatan share baru sudah dilakukan pengecekan syarat share baru diantaranya share baru tidak boleh terdiri dari hanya spasi saja.



Gambar 5.4 Menu awal editor share

Nama share boleh terdiri dari berbagai macam karakter termasuk spasi, tapi tidak boleh hanya spasi saja. Jika user tetap membuat share yang terdiri dari spasi saja maka user tetap dihadapkan dengan editor awal ini. Jika user sudah membuat suatu share atau memilih suatu share maka akan dilanjutkan dengan

editor share sederhana. Dari share sederhana user dapat menggantinya dengan editor share lengkap.



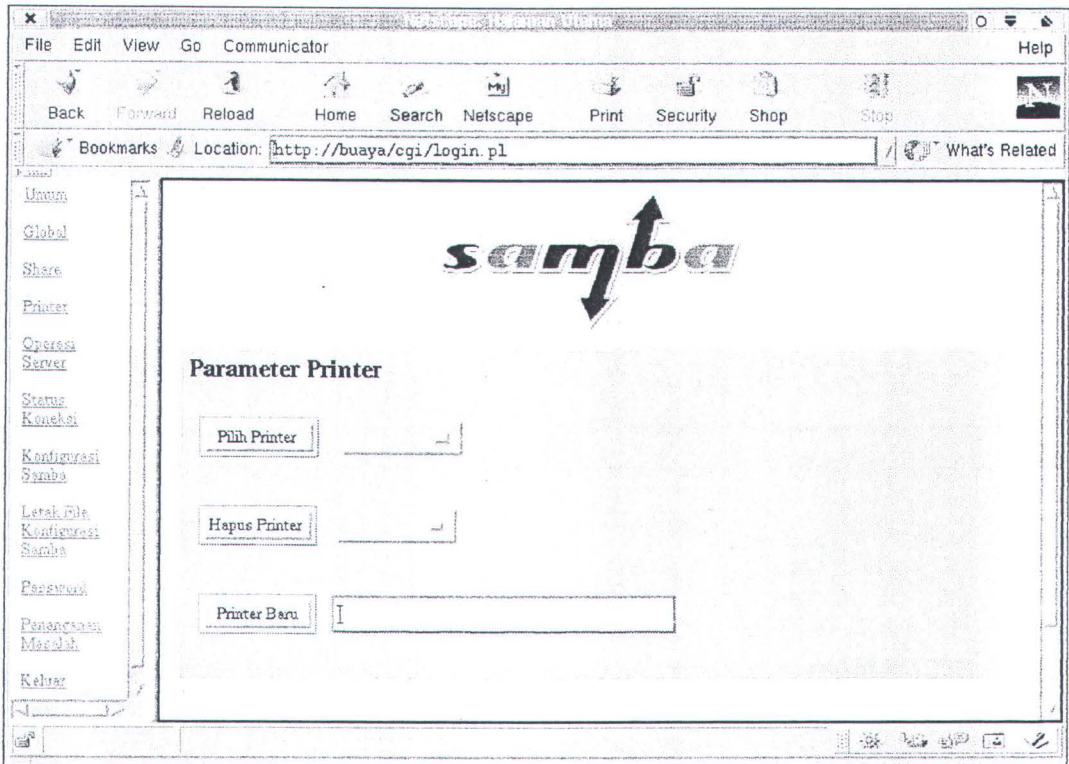
Gambar 5.5 Editor share sederhana

Nama share yang dipilih atau dibuat akan terlihat sehingga user akan yakin bahwa dia sudah mengedit bagaian share yang dimaksud. Disini juga user dapat membuat share baru atau menghapus share . Untuk mendapatkan parameter yang lengkap user dapat berpindah ke **Share Lengkap**.

Setelah selesai mengedit bagian share ini user harus menekan tombol **Tulis** untuk menuliskan perubahan pada file smb.conf. Parameter apa yang telah dimodifikasi dapat dilihat dimenu Konfigurasi Samba.

5.1.4 Mengedit Bagian Printer

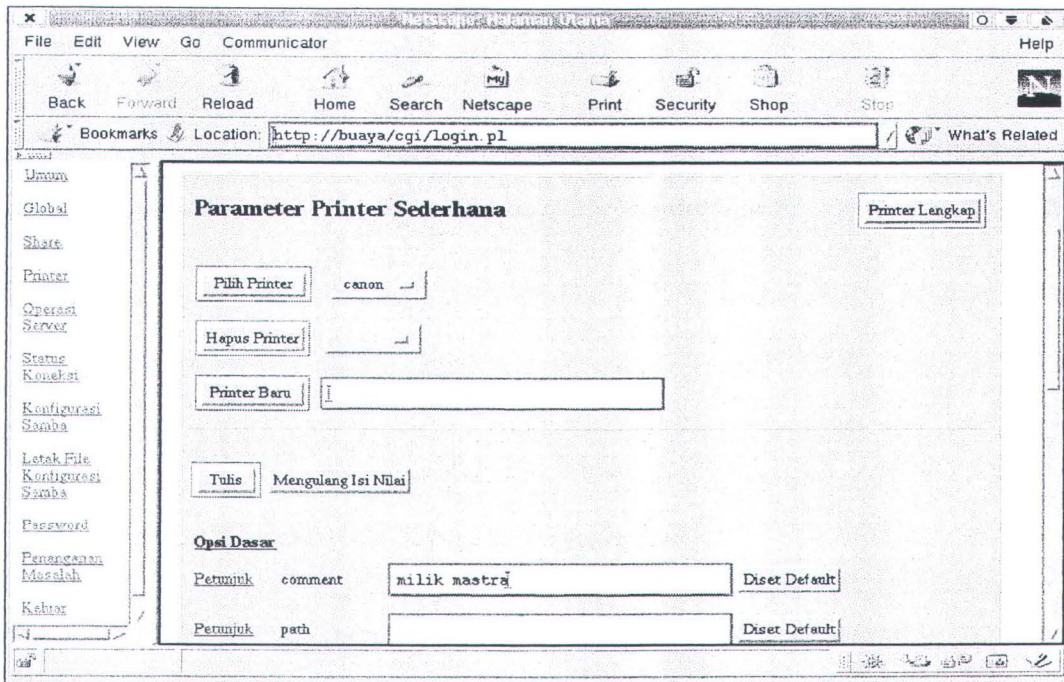
Urutan menu yang ditampilkan sama dengan bagian share, jadi diawali dengan menu awal printer dan dilanjutkan dengan editor printer sederhana.



Gambar 5.6 Menu awal editor printer

Penulis berusaha membuat aplikasi ini dengan SWAT sebagai acuanya. Sehingga kalau dilihat dari segi desain tampilannya sangat menyerupai SWAT. Tapi aplikasi SWATINDO ini untuk share printer masih mempunyai kekurangan yaitu belum menggunakan konfigurasi printer yang terdapat di file /etc/printcap.

Pembuatan share printer baru ini juga mengikuti standar SWAT yaitu [nama printer] yang diikuti oleh parameter *path=/tmp* dan *printable=yes*.

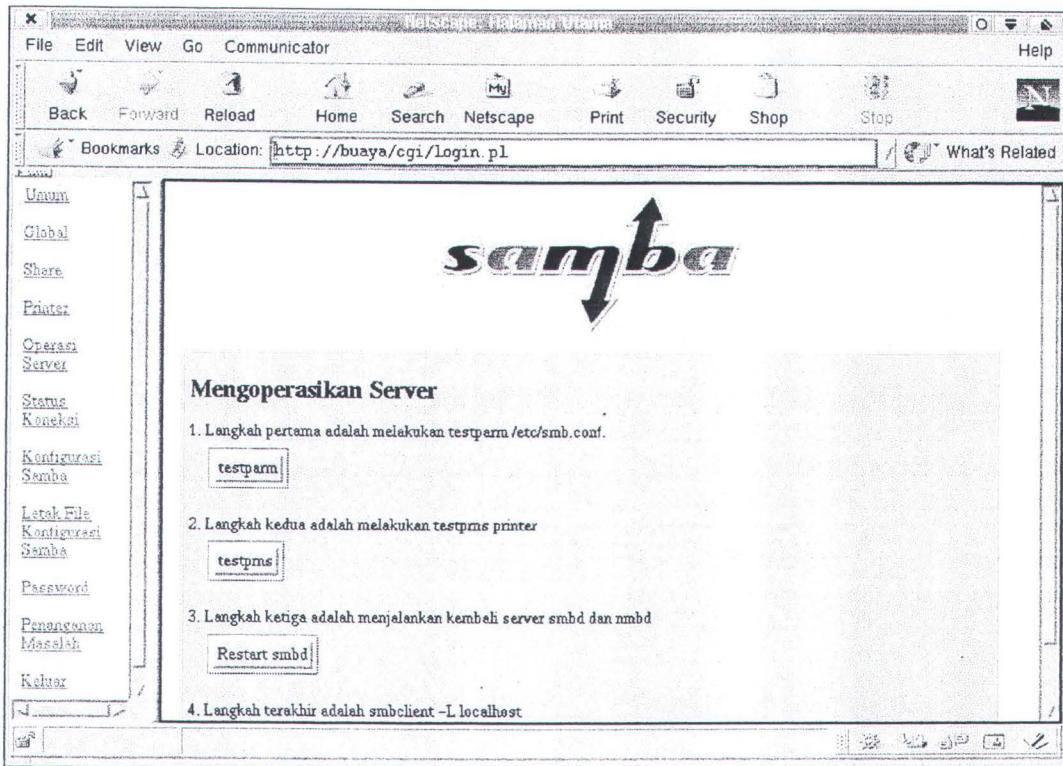


Gambar 5.7 Editor printer sederhana

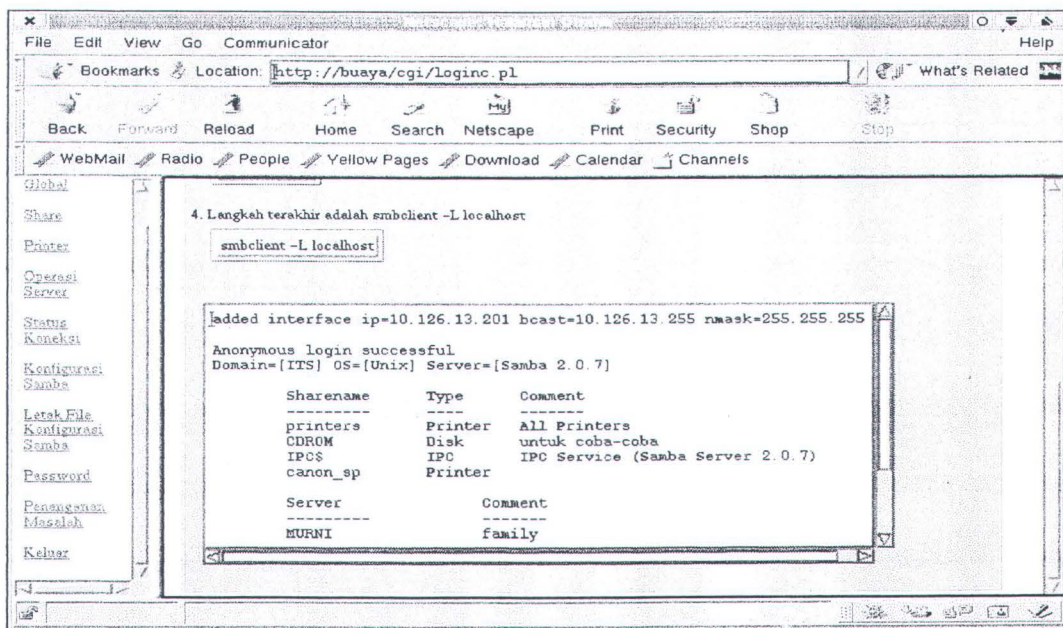
5.1.5 Menjalankan Daemon Samba

Untuk mengaktifkan konfigurasi Samba yang telah dibuat oleh user, maka daemon `smbd` dan `nmbd` harus diaktifkan. Kalau kita mengedit file `smb.conf` dengan editor teks maka setelah selesai harus menjalankan `testparm` dulu untuk memastikan secara sintak semua paramete dan share sudah benar. Lalu dilanjutkan dengan `testprns` kalau membuat konfigurasi printer, baru merestart daemon `smbd` dan `nmbd`. Untuk melihat share apa saja yang diberikan oleh server Samba user dapat menggunakan utilitas `smbclient -L localhost`.

Pada aplikasi SWATINDO penulis tetap menyediakan sub menu `testparm`, jikalau user kurang yakin dengan kebenaran sintak parameter yang dibuat oleh aplikasi SWATINDO ini.



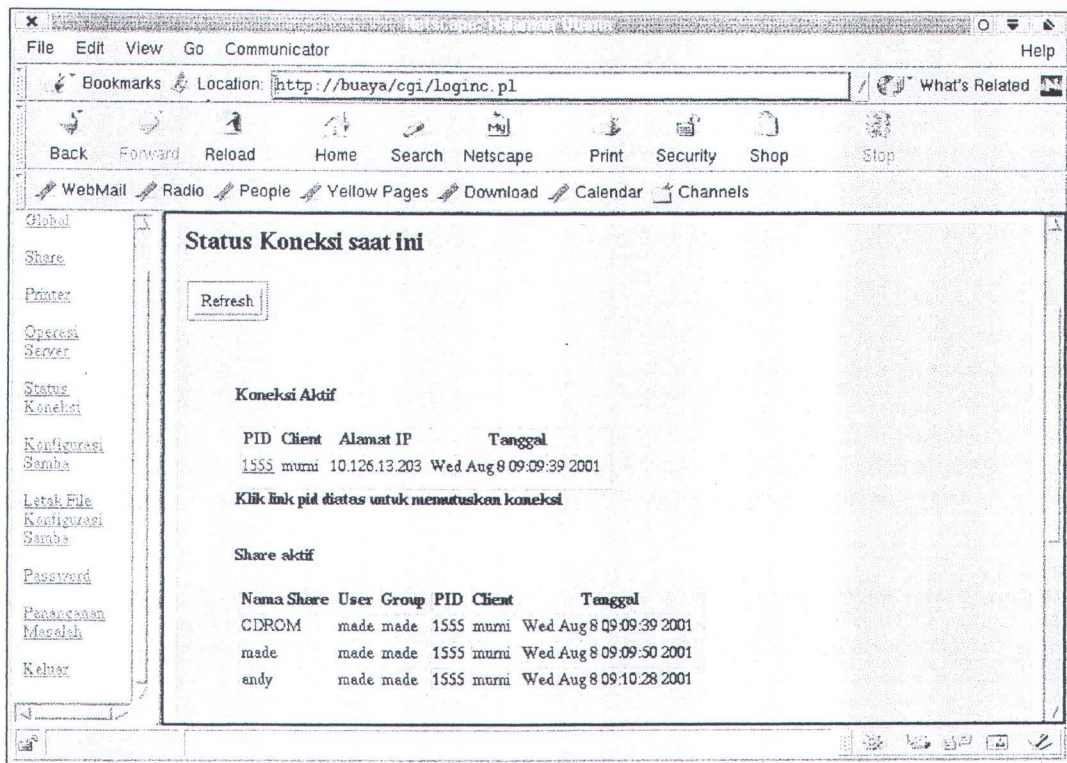
Gambar 5.8 Menjalankan daemon dan utilitas



Gambar 5.9 Contoh output menjalankan utilitas smbclient -L localhost

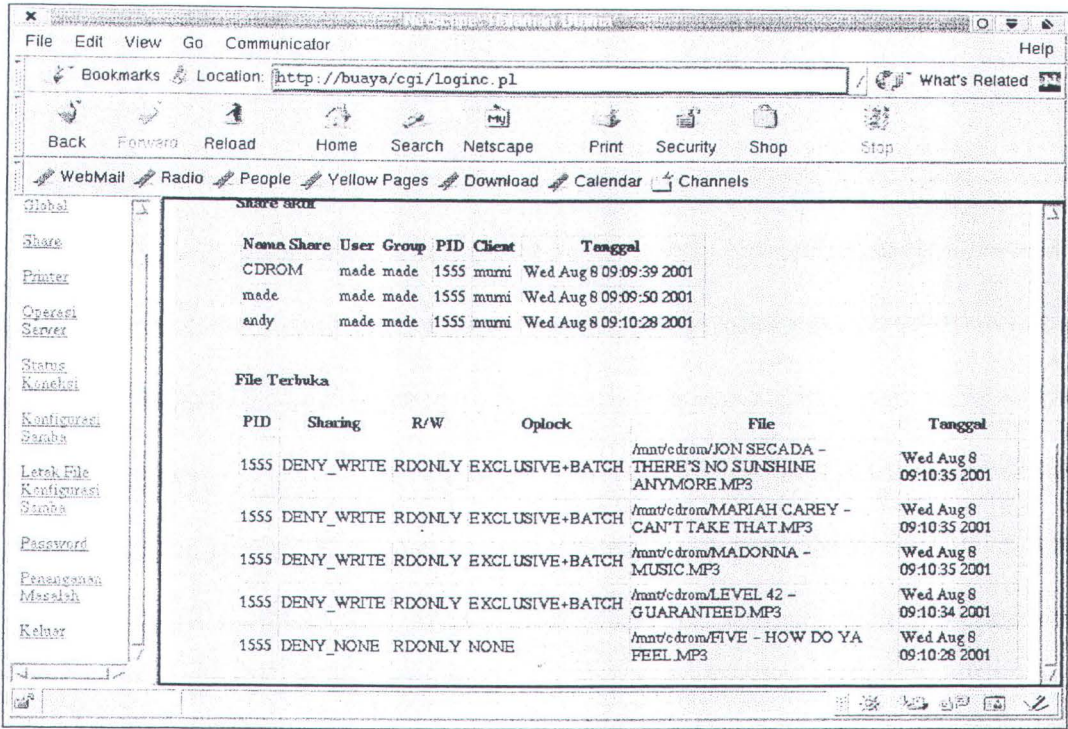
5.1.6 Menginformasikan Status Koneksi Samba

Menu ini digunakan untuk memonitor koneksi Samba. Dapat dilihat client yang sedang mengakses, share mana saja yang diakses bahkan user dapat memutuskan koneksi dengan client yang sedang mengakses suatu share.



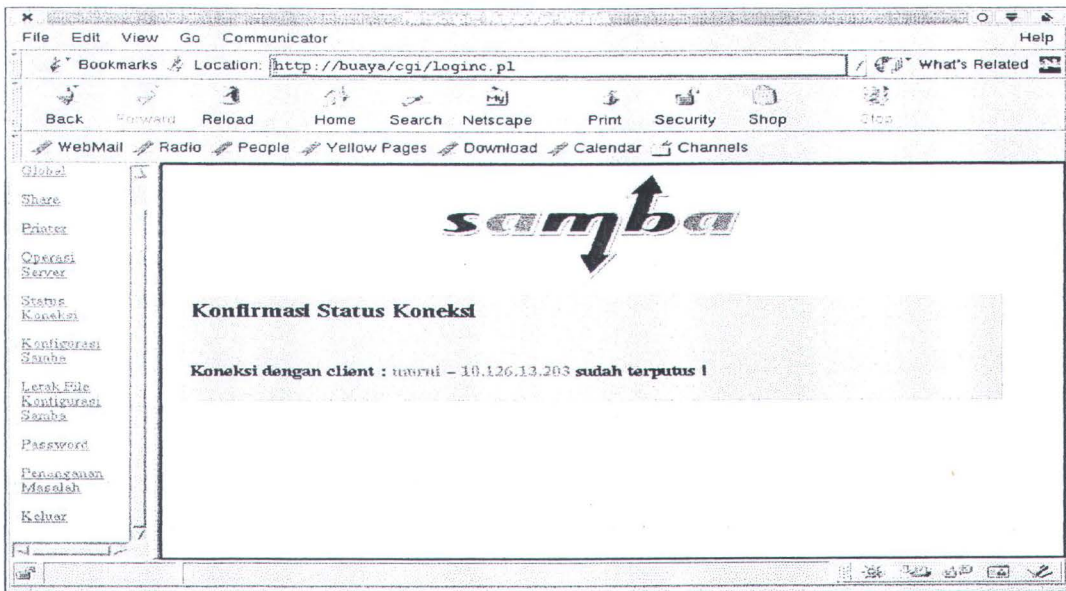
Gambar 5.10 Status koneksi samba

Pada gambar 5.10 diatas terlihat client **murni** sedang mengakses share-share yang ada. Setiap client mempunyai satu PID. Terlihat juga share yang sedang diakses dari sekian share yang ada. Informasi yang tidak tampak pada gambar yaitu informasi file mana yang sedang terbuka, perhatikan gambar 5.11.



Gambar 5.11 Status koneksi Samba (bawah)

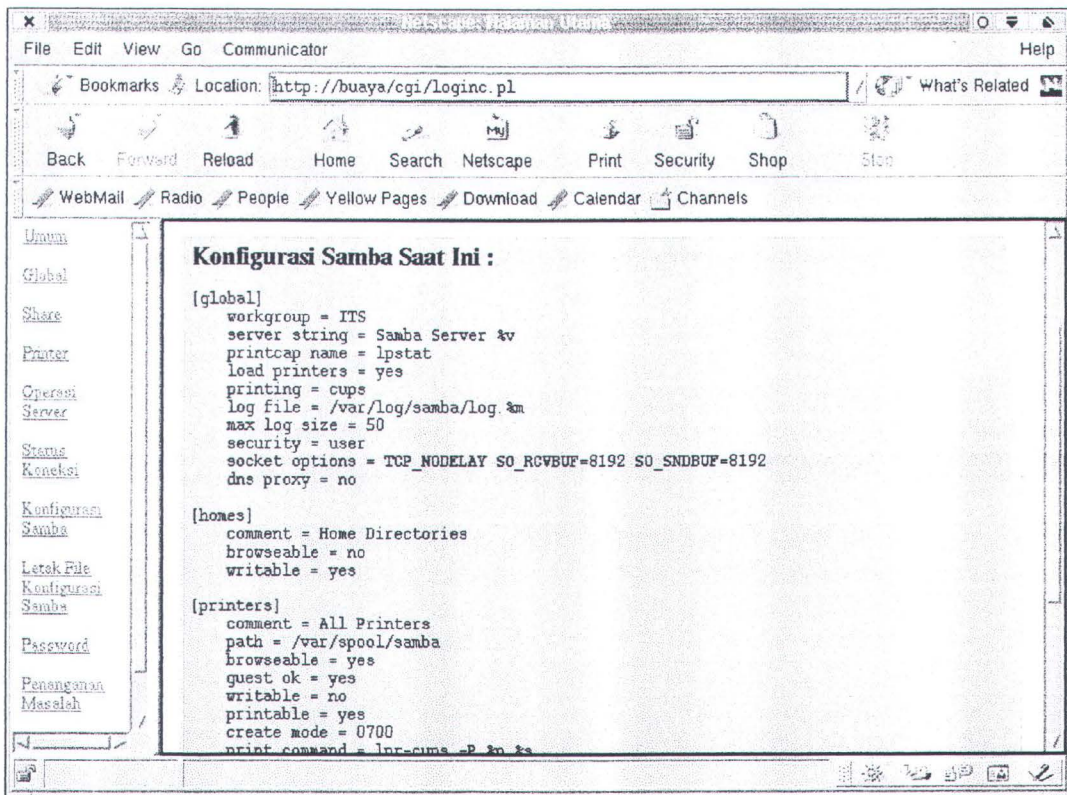
Jika user memutuskan koneksi maka akan ada konfirmasi bahwa koneksi dengan PID tersebut telah terputus. Perhatikan gambar 5.12.



Gambar 5.12 Konfirmasi pemutusan koneksi

5.1.7 Konfigurasi Samba

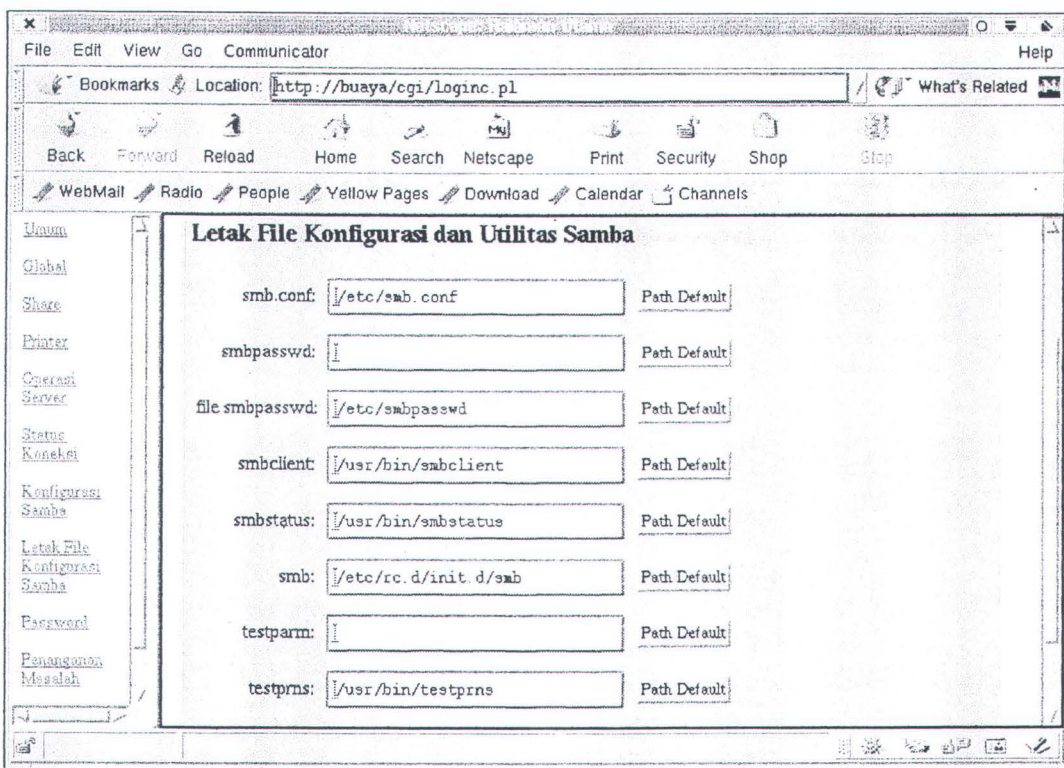
Menu ini kesannya sederhana tapi sangat membantu user, karena user dapat melihat konfigurasi Samba yang telah tersusun rapi, semua parameter dan nama share telah valid secara sintak. Pada aplikasi SWAT *view* konfigurasi Samba kurang sesuai dengan yang terdapat di file smb.conf. Pada file smb.conf yang default dari instalasi Samba pasti mempunyai banyak karakter komentar. Dengan menu konfigurasi Samba ini akan langsung merapikan file tersebut seperti terlihat dibawah ini. Karakter komentar dihilangkan, karena untuk komentar yang merupakan petunjuk ini sudah bisa didapatkan dari link-link yang lebih lengkap dan terfokus penjelasannya.



Gambar 5.13 Konfigurasi Samba

5.1.8 Letak File Konfigurasi dan Utilitas Samba

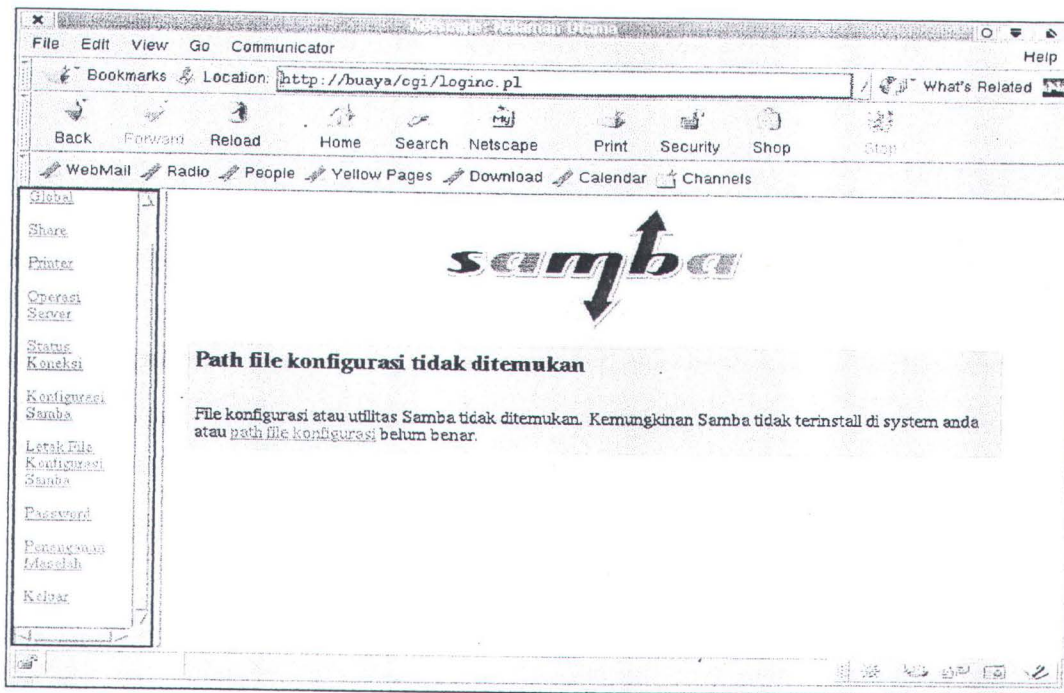
Menu ini digunakan untuk mengedit path file konfigurasi dan utilitas Samba. Jika path yang diberikan ternyata filenya tidak ada maka path tersebut tidak akan ditampilkan kotak input (sama seperti aplikasi Webmin). Otomatis jika Samba di *uninstall* maka path file ini akan hilang semua kecuali path file `/etc/smbpasswd`. Perhatikan gambar 5.14.



Gambar 5.14 Letak file konfigurasi dan Utilitas Samba

Jika Samba di *install* kembali maka path-path file konfigurasi akan muncul kembali di kotak input tersebut. Tombol Path default berguna jika user tidak mengetahui dimana letak file konfigurasi yang dimaksud.

Program-program yang menggunakan file konfigurasi akan selalu mengecek keberadaan file konfigurasi ini. Jika file yang digunakan tidak ada maka akan ada konfirmasi sebagai berikut :

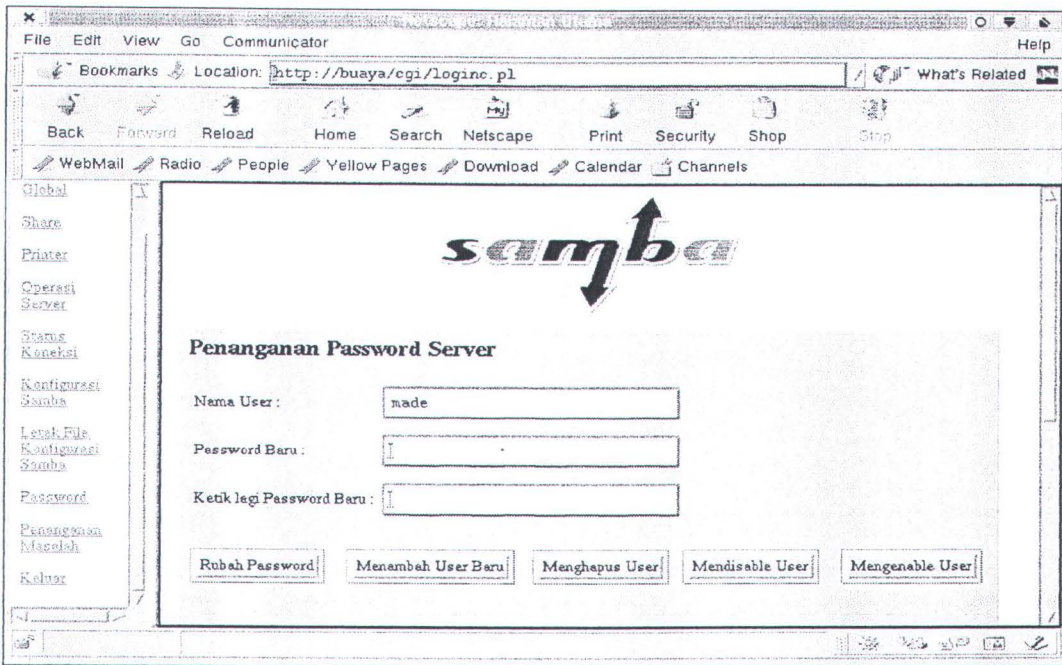


Gambar 5.15 Konfirmasi path file konfigurasi

5.1.9 Menangani Password Samba

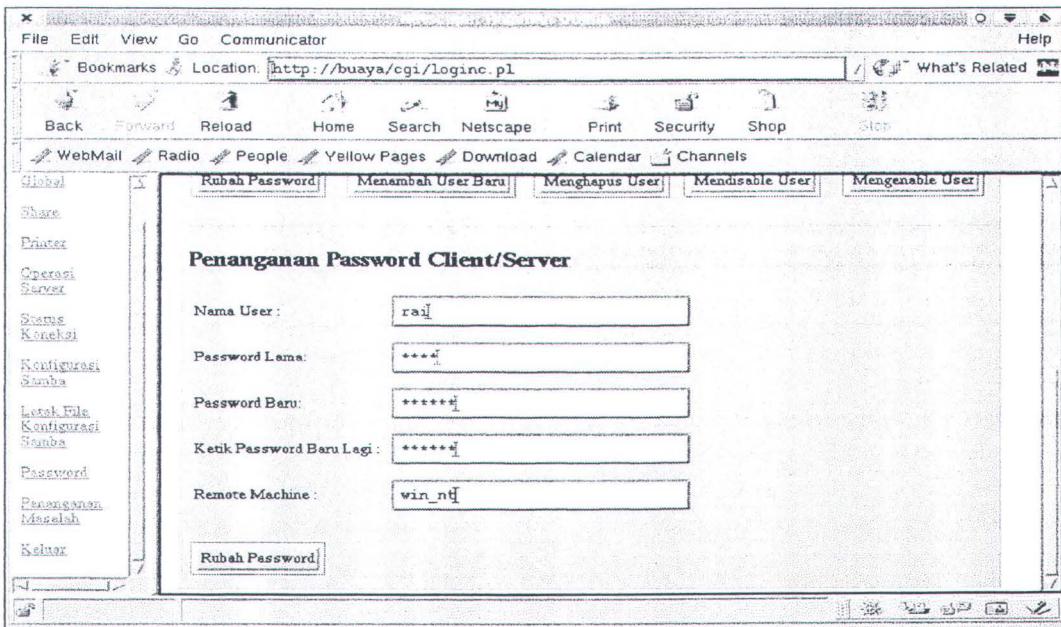
Menu menangani password Samba ini juga terdapat di aplikasi SWAT dan tidak semua opsi dari utilitas smbpasswd digunakan. Opsi yang digunakan hanya yang penting-penting saja. Password yang ditangani/dikelola adalah password lokal (server) dan perubahan password user samba di Windows NT.

5.1.9.1 Mengelola Password Server



Gambar 5.16 Mengelola password server

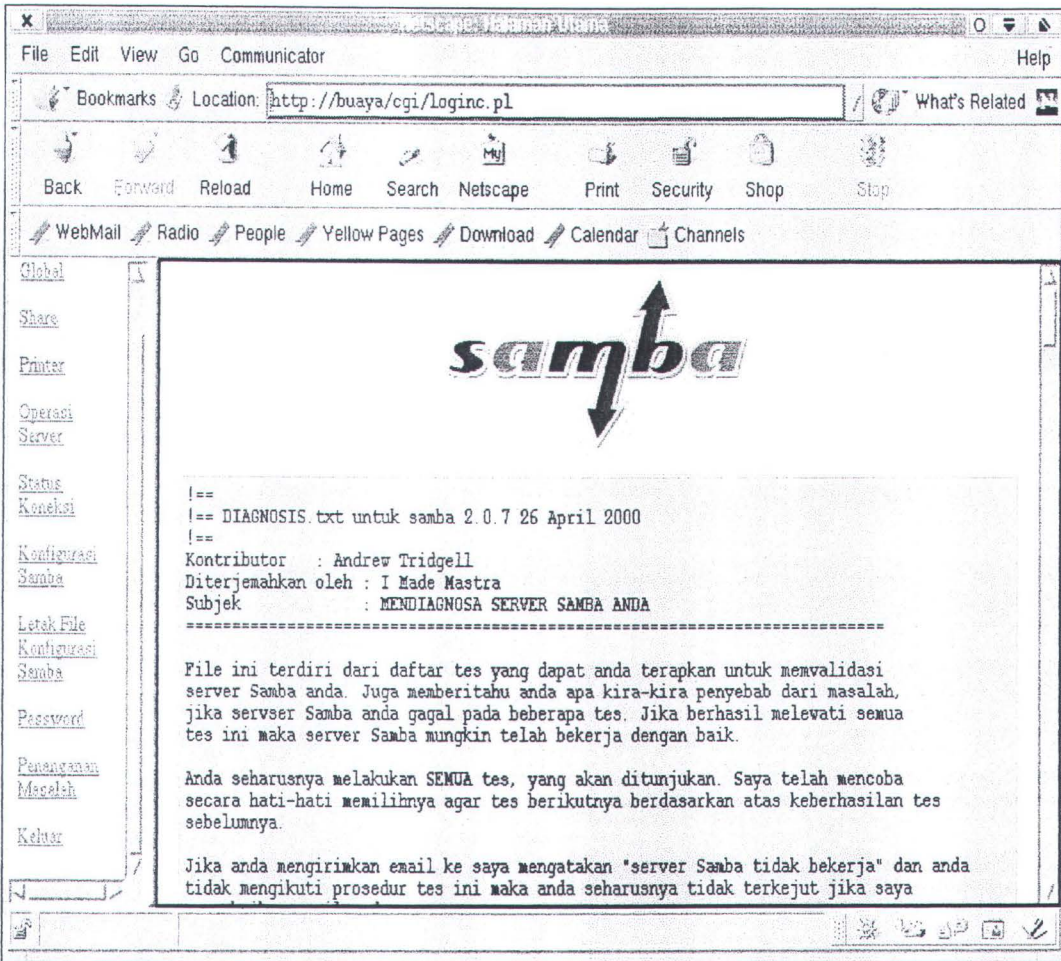
5.1.9.2 Mengelola Password Client-Server



Gambar 5.17 Mengelola password client-server (NT)

5.1.10 Penanganan Masalah Samba

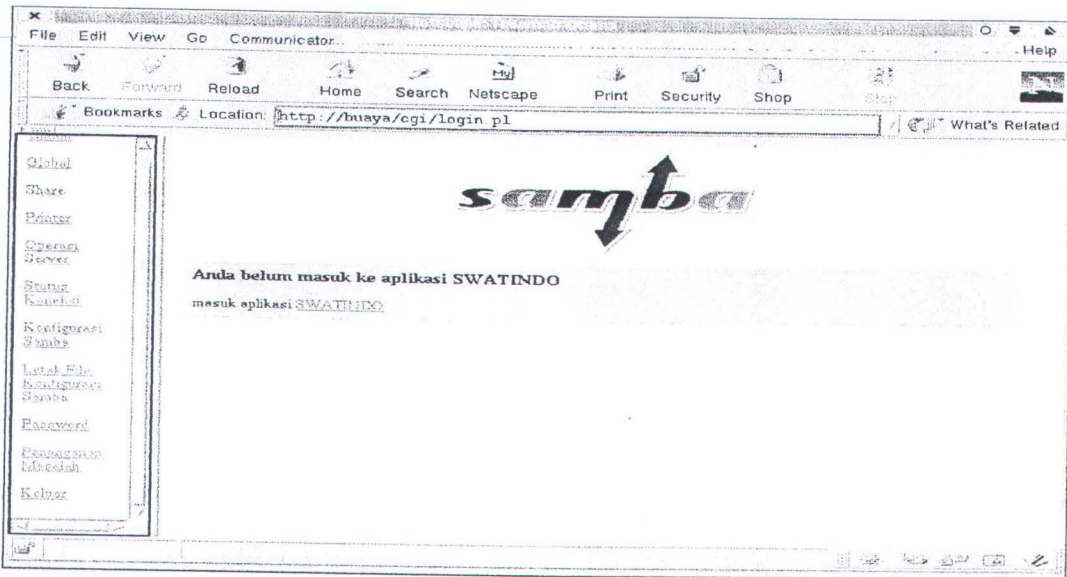
Jika user mengalami kesulitan dalam memecahkan masalah baik itu instalasi dan konfigurasi samba dapat menggunakan artikel berbahasa ini untuk membantu memecahkan masalah.



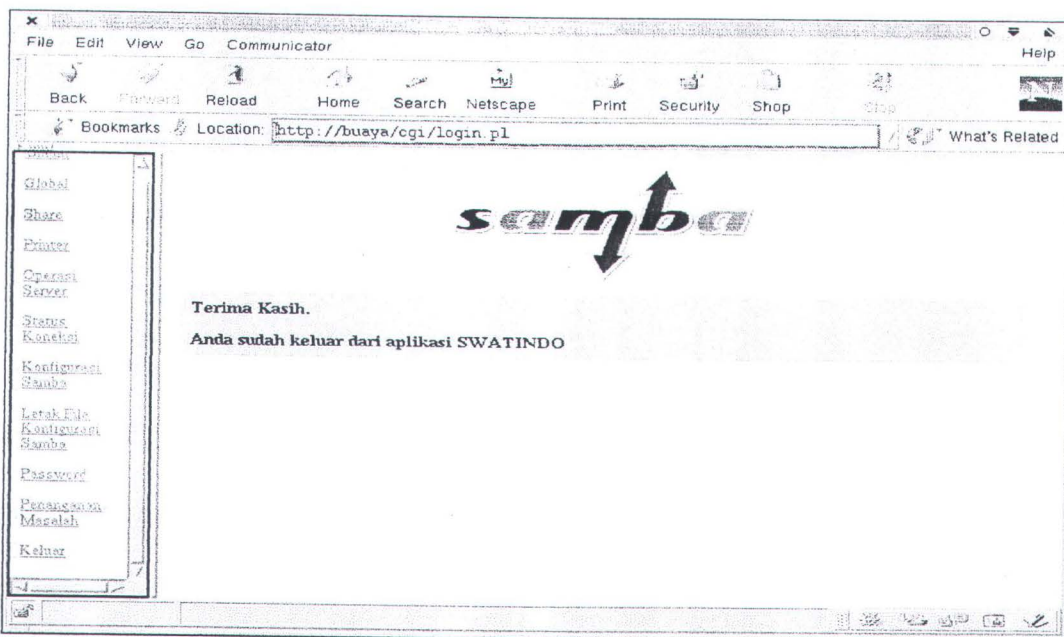
Gambar 5.18 Contoh artikel untuk membantu memecahkan masalah samba

5.1.11 Keluar dari Aplikasi

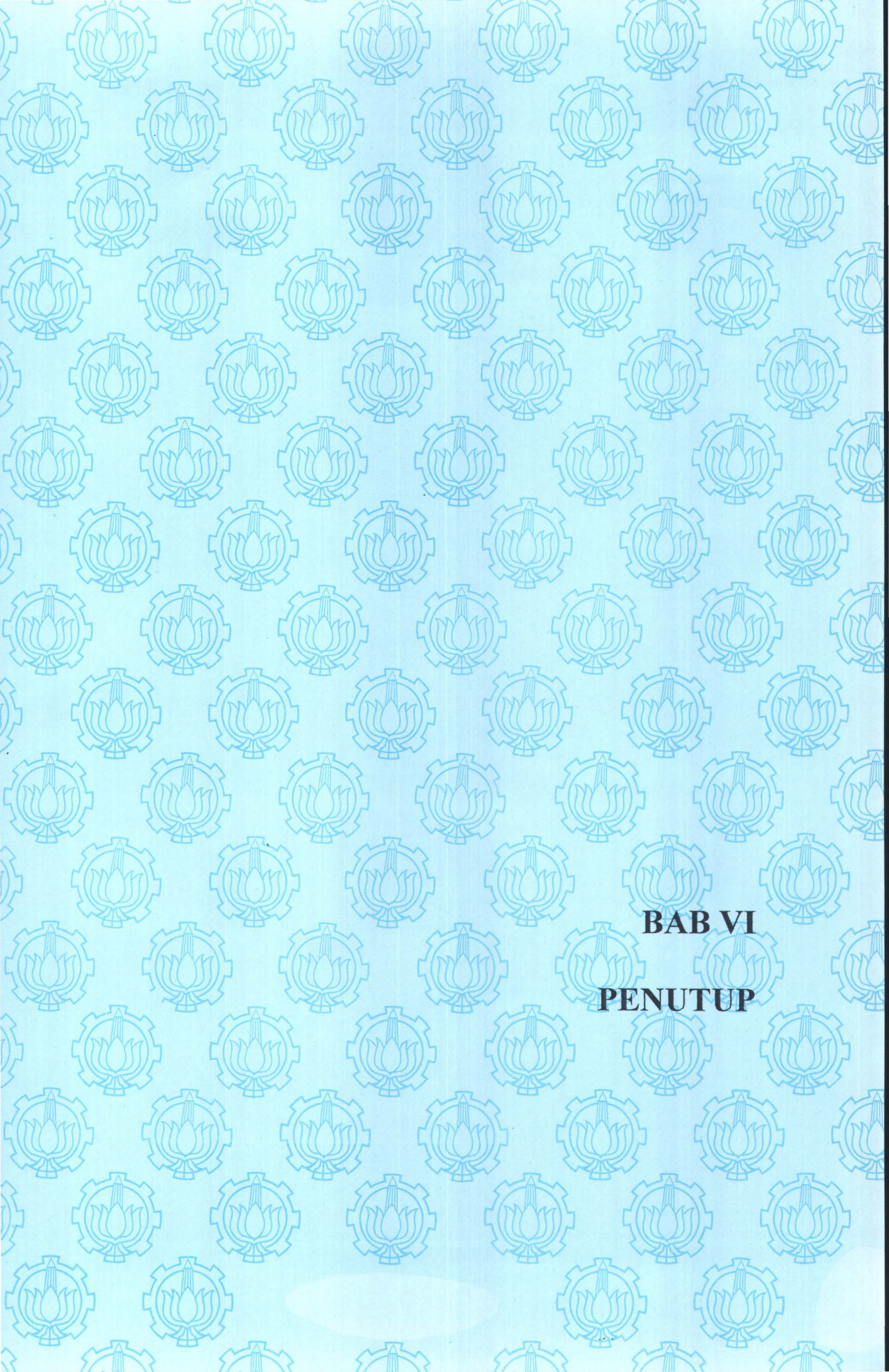
Untuk menggunakan aplikasi ini user harus login. Kalaupun user mengetahui path url dari aplikasi ini namun tidak login, tetap tidak akan bisa menggunakan aplikasi ini dan akan keluar konfirmasi sebagai berikut :



Gambar 5.19 Konfirmasi tidak berhak men SWATINDO



Gambar 5.20 Konfirmasi logout



BAB VI
PENUTUP

BAB VI

PENUTUP

6.1 Kesimpulan

Mengelola Samba akan lebih mudah bila dilakukan dengan aplikasi yang tepat. Aplikasi pengelolaan Samba berbasis web berbahasa Indonesia ini merupakan salah satu tool yang tepat. Karena sudah didukung dengan manual-manual berbahasa Indonesia yang lengkap. Tersedianya parameter Samba yang lengkap, sehingga seorang administrator dapat membuat konfigurasi Samba dari yang sangat sederhana sampai yang paling kompleks. Bila administrator menemui kesulitan saat menginstall dan mengkonfigurasi Samba, sudah ada petunjuk untuk menangani permasalahan tersebut.

6.2 Saran

Untuk pengembangan selanjutnya diharapkan memperhatikan hal-hal berikut :

- Kemanan yang lebih baik

Penggunaan SSL (Secure Socket Layer) dapat dipertimbangkan untuk lebih menjamin pada transfer data, meskipun harus merelakan waktu akses menjadi lebih lambat.

5.2 Analisa

Aplikasi pengelolaan samba berbasis web ini akan sangat membantu administrator pemula dan berpengalaman untuk mengkonfigurasi sambanya. Karena didukung dengan petunjuk berbahasa Indonesia, dan parameter-parameternya yang lengkap sudah dikelompokkan dengan fungsinya.

Selesai mengedit file `smb.conf`, admin dapat menjalankan utilitas-daemon samba, seperti : mengecek apakah keseluruhan sintak sudah benar, mengecek nama printer, merestart `smbd` dan mengetahui servis apa saja yang diberikan oleh samba kita.

Bila admin mengalami masalah saat konfigurasi dapat langsung membuka artikel yang mengulas mengenai masalah konfigurasi samba.

- Penyediaan utilitas yang lebih lengkap. Seperti tersedianya opsi yang lebih lengkap pada `testparm`, `testprns`, `smbclient`, `smbpasswd`.



DAFTAR PUSTAKA

DAFTAR PUSTAKA

- [1] Daniel Lopez Ridruejo. Apache Overview How-to. ridruejo@apache.org.2000
- [2] Jesus Castagnetto, Haris Rawat, Sascha Schumann. Professional PHP Programming. Wrox Press. 1999
- [3] Mícheál Ó Foghlú. Perl Quick Reference. Que Corporation. 1966
- [4] Paul J. Wagner. Andy Lindeman. Samba Tips. pwagner@uwstout. 1996
- [5] Robert Eckstein, David Collier-Brown. Using Samba. O'Reilly & Associates Peter Kelly.1999
- [6] Timoth Parker. Linux System Administrator's Survival Guide. *SAMS.net*.1996