

PENERAPAN OPENSSSH DAN BASH SCRIPT UNTUK SIMULTANEOUS REMOTE ACCESS CLIENT PADA LABORATORIUM STMIK AMIKOM YOGYAKARTA

Andika Agus Slameto¹, Lukman²

¹Dosen Teknik Informatika, Program Strata 1, STMIK AMIKOM Yogyakarta

² Dosen Teknik Informatika, Program Strata 1, STMIK AMIKOM Yogyakarta

Jl. Ring Road Utara, Condong Catur, Sleman, Yogyakarta

Telp: (0274) 884201-207, Fax: (0274) 884208 Kodepos: 55283

E-mail: rmkt.andika@amikom.ac.id, masman@amikom.ac.id

Intisari

Openssh merupakan aplikasi open source yang dibuat oleh para pengembang sistem operasi OpenBSD dan pada akhirnya dijadikan aplikasi remote shell pada platform Unix termasuk di dalamnya adalah GNU/Linux. Openssh memungkinkan seseorang untuk menjalankan sebuah komputer berplatform Unix secara remote. Bash (Bourne Again Shell) merupakan antarmuka awal antara pemakai dan komputer pada sistem operasi berbasis Unix. Bash memungkinkan pemakai untuk membuat program. Program Bash ditulis pada file sebagai file teks dan biasa disebut skrip Bash atau Bash Scripting. Skrip Bash dapat berupa perintah internal Bash, perintah GNU/Linux (utilitas-utilitas GNU/Linux atau Unix), program aplikasi atau skrip yang lain. Pada penelitian ini dengan menggabungkan antara Openssh dengan skrip Bash yang dapat dimanfaatkan untuk mengendalikan jaringan yang memiliki platform GNU/Linux dan dapat dilakukan secara simultan. Selain itu penelitian ini juga sebagai alternatif untuk mentransfer atau mendistribusikan file atau direktori pada komputer klien. Penelitian ini dapat diterapkan pada jaringan dengan skala lebih besar yaitu internet misalkan untuk perawatan atau mengendalikan pc router, gateway, proxy, atau server dengan berbagai kegunaanya.

Kata kunci: OpenSSH, Bash Script, Linux, Remote Access, server

Abstract

Openssh is an open source application that is created by the developers of the OpenBSD operating system and ultimately be used as a remote application shell on Unix platforms including the GNU / Linux. Openssh allows one to run a Unix platform computer remotely. Bash (Bourne Again Shell) is the initial interface between the user and the computer on a Unix-based operating system. Bash allows the user to create a program. Bash program is written to the file as a plain text file called a Bash script or Bash Scripting. Bash script can be either internal Bash commands, command GNU / Linux (utilities GNU / Linux or Unix), an application program or other scripts. In this study, by combining the OpenSSH with a Bash script that can be used to control the network that has the platform GNU / Linux, and can be carried out simultaneously. In addition, this study also as an alternative to transfer or distribute a file or directory on the client computer. This research can be applied on a larger scale network with internet suppose to treatment or control pc router, gateway, proxy, or servers with different role.

Keyword: Keywords: OpenSSH, Bash Script, Linux, Remote Access, the server

PENDAHULUAN

Laboratorium Jaringan Komputer Sekolah Tinggi Manajemen Informatika dan Komputer (STMIK) Amikom Yogyakarta memiliki kapasitas tujuh puluh lima komputer. Laboratorium ini berbeda dengan laboratorium komputer lainnya karena masing-masing komputer klien terinstal dua sistem operasi, yaitu sistem operasi Windows XP dan sistem operasi GNU/Linux distributor Ubuntu 12.04.

Pada saat pemeliharaan sistem operasi GNU/Linux pasca masa praktikum atau di akhir semester mengalami kendala waktu yang berlarut-larut sehingga tidak efisien waktu. Pemeliharaan sistem operasi GNU/Linux meliputi hal berikut.

Membersihkan atau menghapus direktori-direktori dan *file-file* yang tidak terpakai pasca praktikum pada tiap komputer klien.

Mengembalikan konfigurasi atau seting komputer pasca praktikum jika hal tersebut diperlukan.

Merestart atau star ulang (jika diperlukan) dan mematikan komputer pasca perawatan mengalami kelambanan.

Apabila pemeliharaan dilakukan secara *remote* (dikendalikan dari komputer lain) memerlukan banyak perintah yang harus dijalankan sehingga memerlukan ketelitian dan waktu yang lebih banyak.

Saat masa praktikum dengan sistem operasi GNU/Linux juga mengalami beberapa kendala. Kendala tersebut di antaranya.

Asisten dosen mengalami kelambanan saat harus mendistribusikan suatu *file* misal *file* yang berisi soal atau tes kemampuan ke komputer klien yang digunakan mahasiswa.

Saat mengumpulkan hasil tes kemampuan juga mengalami hal yang sama dengan di atas.

Pada akhir sesi pengajar mengalami kendala waktu untuk mematikan atau *merestart* (star ulang) seluruh komputer klien.

Openssh merupakan aplikasi *open source* yang dibuat oleh para pengembang sistem operasi OpenBSD dan pada akhirnya dijadikan aplikasi *remote shell* pada *platform* Unix termasuk di dalamnya adalah GNU/Linux. Openssh memungkinkan seseorang untuk menjalankan sebuah komputer dengan *platform* Unix secara *remote*.

Bash (Bourne Again Shell) merupakan antarmuka awal antara pemakai dan komputer pada sistem operasi berbasis Unix. Bash memungkinkan pemakai untuk membuat program. Program Bash ditulis pada *file* sebagai *file* teks dan biasa disebut skrip Bash atau Bash *Scripting*. Skrip Bash dapat berupa perintah internal Bash, perintah GNU/Linux (utilitas-utilitas GNU/Linux atau Unix), program aplikasi atau skrip yang lain.

TINJAUAN TEORI

1. Tinjauan Pustaka

Ginting (2009), Meneliti tentang Tundaan pada jaringan adalah suatu hal yang tidak dapat dihindari ketika mentransmisi data digital antara perangkat-perangkat kontrol. Sistem kendali jaringan yang dibahas disini terdiri dari sensor clock driven, pengendali even driven, aktuator dan plant. Dalam sistem kendali jaringan terdapat tundaan dan periode sampling yang mempengaruhi kestabilan dari sitem tersebut. Daerah kestabilan untuk sistem kendali jaringan dapat digambarkan melalui hubungan antara Maximum Allowable Transfer interval dan periode sampling. [1]

Muhammad Iswan (2010), Melakukan penelitian dengan menggunakan VPN remote

access dengan menggunakan OpenSwan untuk membangun sebuah sistem jaringan yang digunakan untuk menghubungkan komunikasi jaringan lokal dengan jaringan publik secara aman. Sistem ini berjalan diatas protokol Ipsec sehingga komunikasi yang berjalan terjamin keamanannya. [2]

Kunang (2009), Membuat sebuah laboratorium virtual dengan memanfaatkan Telnet, Secure Shell (SSH), Virtual Network Computing (VNC), dan Diskless . Dalam penelitian tersebut, telah di uji kestabilan dan kemampuan sistem yang dibuat dapat memberikan nilai tambah di dalam laboratorium. [3]

2. Landasan Teori

a. GNU Linux

Proyek GNU diluncurkan pada tahun 1984 untuk mengembangkan sebuah sistem operasi lengkap yang serupa dengan Unix yang berbasis perangkat lunak bebas yaitu sistem GNU. Kernel GNU tidak pernah selesai, sehingga GNU menggunakan kernel Linux. Kombinasi GNU dan Linux merupakan sistem operasi GNU/Linux, yang kini digunakan secara luas. Terkadang, kombinasi ini secara keliru disebut Linux. GNU merupakan singkatan rekursif dari “GNU’s Not Unix” (GNU bukan Unix), serta dilafalkan *ge-nuu*.

Sejarah sistem operasi Linux berkaitan erat dengan proyek GNU, proyek pengembangan perangkat lunak bebas (*free software*) yang didasari pandangan bahwa perangkat lunak itu harusnya tidak menyertakan batasan apapun (bebas) saat diterima oleh pengguna. Pandangan ini tertuang dalam satu lisensi dengan nama General Public Licence (GPL) Framework.

i. Open Secure Shell (OpenSSH)

Salah satu keunggulan dari sistem operasi berbasis GNU/Linux adalah tidak harus mendatangi mesin ketika bekerja dengan jaringan, melainkan cukup dengan akses *remote* ke mesin tersebut. Akses *remote* yang dilakukan bergantung pada aplikasi seperti telnet (*Telecommunication Network*), rlogin (*Remote Login*), ssh (*Secure Shell*) dan aplikasi berbasis teks. Penggunaan aplikasi berbasis teks ini tidak memerlukan GUI (*Graphical User Interface*) seperti pada aplikasi *remote desktop*. Aplikasi *remote dekstop* membutuhkan *resource* cukup besar dan tidak dapat digunakan pada mesin yang tidak menyalakan X, seperti *server* atau *pc router*. [4]

Akses *remote* dilakukan dengan menggunakan aplikasi seperti telnet, rsh, rlogin dan ssh dengan cara melakukan *otentikasi* ke mesin *remote* (yaitu mesin atau komputer yang dikendalikan dari mesin lain melalui sarana jaringan), setelah proses *otentikasi* maka sebuah komputer dapat menjalankan perintah-perintah pada komputer lain yang dijadikan sebagai mesin *remote* seolah-olah berada di depannya. Telnet merupakan salah satu aplikasi akses *remote* yang dibuat pada tahun 1969 dan menjadi standar internet pertama.

Pemakaian aplikasi *remote* seperti telnet saat melewati datanya dalam kondisi tidak terenkripsi, sehingga memungkinkan terjadinya *connection hijacking* juga dikenal sebagai *Man in the Middle Attack* (suatu serangan jahat yang mencegah komunikasi yang sah antara dua *host* untuk mengendalikan jalur komunikasi dan untuk menghapus atau menambah informasi yang dikirim oleh pengirim yang sebenarnya tanpa diketahui olehnya), *eavesdropping* (penyadapan informasi atau data pada suatu jaringan) dan

serangan lainnya. SSH ditujukan sebagai pengganti rlogin dan rsh untuk menyediakan komunikasi yang aman dan terenkripsi antara dua *host* menggunakan jaringan.

Ssh adalah program untuk memasuki ke sebuah mesin remote dan untuk mengeksekusi perintah-perintah pada mesin remote. Ssh dimaksudkan untuk menggantikan rlogin dan rsh, dan dilengkapi komunikasi yang dienkripsi secara aman antara dua *host* (komputer) yang tidak saling mempercayai melalui sebuah jaringan yang rentan. Ssh terhubung dan masuk ke dalam suatu *hostname* (dengan nama pengguna pilihannya). Pengguna harus memberikan identitasnya pada mesin remote menggunakan salah satu dari beberapa metode sesuai versi protokol yang digunakan.

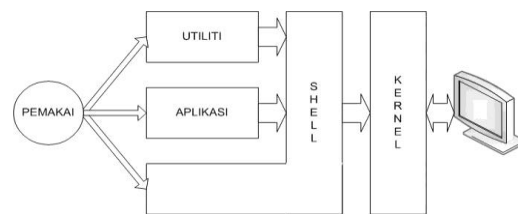
Openssh merupakan salah satu aplikasi SSH yang disediakan secara gratis, perangkat ini dikembangkan oleh OpenBSD Project. *Software* ini dikembangkan dalam lingkungan yang memungkinkan ekspor kriptografi. Perangkat Openssh menggantikan rlogin dan telnet dengan program ssh, rcp dengan scp, dan ftp dengan sftp. Termasuk didalam paket Openssh adalah sshd sebagai paket yang dijalankan pada sisi server (komputer yang akan *diremote*), dan lain seperti ssh-add, ssh-agent, ssh-keysign, ssh-keyscan, ssh-keygen dan sftp-server. [5]

b. Shell dan Bash (Bourne Again Shell)

Shell pada sistem operasi GNU/Linux atau Unix memiliki kedudukan sebagai perantara awal antara pemakai dan komputer. *Shell* memiliki peranan penting dalam sistem operasi GNU/Linux sebagai penerjemah perintah (*command interpreter*). *Shell* (disebut juga sebagai terminal) yang telah diaktifkan oleh sistem operasi akan menampilkan *prompt* dan menunggu pemakai memasukkan perintah dari

keyboard. Ilustrasi tentang hubungan shell, pemakai dan komputer dapat dilihat pada gambar 1.

Kernel merupakan bagian utama dari sistem operasi yang mengendalikan perangkat keras sistem dan melaksanakan berbagai fungsi beraras rendah, seperti menampilkan tulisan di layar dan membaca data dari *disk*. Utiliti adalah program-program yang tersedia pada sistem operasi yang masing-masing mempunyai tugas spesifik seperti daftar *file* direktori atau melihat



Gambar 1. Shell Sebagai Perantara Antara Pemakai dan Komputer

isi *file*. Program ini akan dimuat ke memori dari *disk* saat digunakan. *Shell* sendiri termasuk sebagai utiliti. Aplikasi adalah program-program yang dibuat oleh pemakai. Program-program ini dapat dibuat dengan menggunakan sejumlah utiliti, perintah *shell* atau pun dibangun dengan bahasa pemrograman seperti C, COBOL ataupun 4GL. Aplikasi dapat juga berupa program paket yang dibeli dari pemasok perangkat lunak. [6]

Sistem operasi Microsoft Windows memiliki satu shell, yaitu *DOS prompt*, pada sistem operasi GNU/Linux atau Unix dikenal beberapa *shell*, diantaranya :

- Sh (Bourne Shell), dibuat oleh Stephen R Bourne merupakan *shell* tertua di lingkungan Unix varian.
- Csh (C Shell), dibuat oleh Bill Joy memiliki karakteristik yang lebih interaktif (seperti job-control, alias dan history mechanism) dan sintak pemrogramannya mendekati bahasa C. [7]

- c. Tcsh, csh dengan penambahan karakteristik
- d. Ksh (Korn Shell), dibuat oleh David Korn.
- e. Bash (Bourne Again Shell), dibuat oleh Brian Fox dan Chet Ramey merupakan pengembangan dari sh
- f. Bash merupakan *shell* yang paling banyak digunakan pada sistem operasi GNU/Linux. Bash kompatibel dengan *shell* sh dan ditambah dengan kemampuan atau karakteristik yang dimiliki oleh Korn Shell (ksh) dan C Shell (csh). Bash juga dibakukan sebagai sebuah *shell* yang mengimplementasikan spesifikasi IEEE POSIX Shell dan Tools (IEEE Working Group 1003.2). [7]

Bash merupakan suatu bahasa perintah (*command language*) yang dapat digunakan untuk mengontrol eksekusi program. Bash juga merupakan bahasa pemrograman (*programming language*) yang dapat digunakan untuk membuat program yang terstruktur. Sebagai bahasa perintah (*command language*) Bash memiliki karakteristik sebagai berikut.

- a. Eksekusi program, yang memungkinkan untuk menjalankan program secara berurutan dan paralel.
- b. Pembuatan *File*, dapat membuat *file* baru dan menambah ke *file* yang ada
- c. *Argument Passing*, yang memungkinkan melewati argumen ke program melalui baris perintah maupun *environment* variabel.
- d. Eksekusi program berkondisi, memungkinkan untuk menjalankan suatu program berdasar pada keberhasilan atau kegagalan dari program lainnya.
- e. *Predefined procedures*, dapat digunakan untuk menyusun suatu skrip yang mendefinisikan suatu urutan program yang akan dieksekusi

- f. *Parameterization*, memungkinkan untuk menulis skrip menggunakan variabel *shell* (*shell variable*) sehingga skrip dapat bersifat lebih *general*. [7]

Bash sebagai bahasa pemrograman (*programming language*) memiliki beberapa karakteristik sebagai berikut.

- a. *Variables*, dapat mendefinisikan variabel dan melakukan operasi terhadapnya.
- b. *Structured Language Constructs*, menyediakan struktur *sequence*, *selection* dan iterasi (perulangan).
- c. *Scope*, dapat membatasi ruang lingkup tempat antara variabel dan program dapat saling mengenal.
- d. *Macro Substitution*, Bash menyediakan *macro* dan *header file*
- e. *Subroutines*, dapat ditulis dan memanggil suatu subrutin dan memungkinkan untuk membuat suatu rekursi di dalamnya. [7]

METODOLOGI PENELITIAN

Metode penelitian yang dilakukan dalam perancangan dan pembuatan aplikasi ini adalah sebagai berikut.

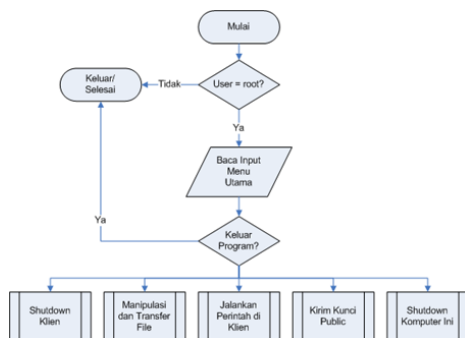
- a. Mengumpulkan bahan atau materi penelitian, berupa :
 - a. Halaman manual dari nama perintah yang digunakan
 - b. Artikel dan jurnal yang terkait dengan Openssh, Bash, skrip *shell* dan keamanan jaringan.
 - c. Buku referensi tentang skrip shell, sistem operasi GNU/Linux atau Unix.
- b. Perangkat keras yang digunakan
 - a. Komputer pengendali yang memiliki sistem operasi GNU/Linux secara fisik atau dalam mesin *virtual* (menggunakan

software VMware).

- b. Komputer klien yang memiliki sistem operasi GNU/Linux yang sudah terpasang dan menjalankan *daemon* ssh.
- c. Perangkat lunak yang digunakan
 - a. Openssh yang terpasang pada komputer pengendali dan *daemon* Openssh pada komputer klien
 - b. *Shell* Bash yang dipasang pada komputer pengendali dan direkomendasikan juga terpasang pada klien
- d. Jalannya penelitian.
 - a. Mempelajari literatur
 - b. Analisa sistem
 - c. Perancangan dan desain sistem
 - d. Implementasi dari perancangan sistem
 - e. Testing dan implementasi system
 - f. Menganalisa hasil penelitian, evaluasi dan pembahasan

HASIL DAN PEMBAHASAN

1. Proses Sistem



Gambar 8. Diagram Alur Sistem

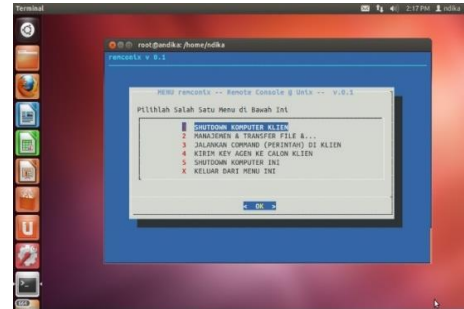
Untuk menjalankan sistem diperlukan otoritas super user. Dalam linux untuk super user adalah user root untuk masing-masing klien dan server pengendali

g. Antarmuka Sistem

Perancangan antar muka pengguna (*user*

interface) dalam pembangunan perangkat lunak sebagai kelanjutan dari implementasi program yang dibangun. Berikut adalah *user interface* yang akan dibuat dalam sistem ini.

a. Menu Awal



Gambar 8. Menu Awal

Script bash untuk membuat menu awal aplikasi ini adalah sebagai berikut :

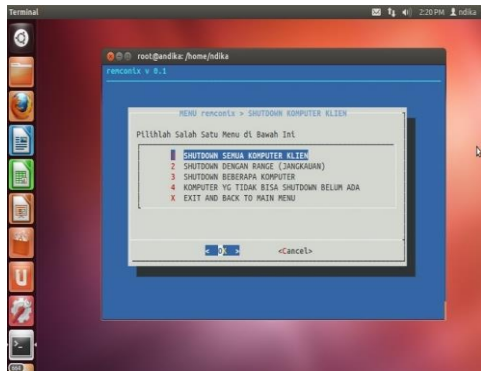
```

JUDUL="MENU $APPL"
OMAN="$DESKAPP v.0.1"
MTEKS="\nPilihlah Salah Satu
Menu di Bawah Ini"
PILIH="1"
PIL1="SHUTDOWN KOMPUTER KLIEN"
PIL2="MANAJEMEN & TRANSFER FILE
&..."
PIL3="JALANKAN COMMAND
(PERINTAH) DI KLIEN"
PIL4="KIRIM KEY AGEN KE CALON
KLIEN"
PIL5="SHUTDOWN KOMPUTER INI"
PIL6="KELUAR DARI PROGRAM"
DIMEN1="17 65"
DIMEN2="15 65"
DIMEN3="10 65"
Main_Menu()
{
    dialog --backtitle "$BT" --
    default-item $PILIH --nocancel --
    title "$JUDUL$OMAN" --menu "$MTEKS"
    $DIMEN2 6 \
        1 "$PIL1" 2 "$PIL2" 3
    "$PIL3" 4 "$PIL4" 5 "$PIL5" X "$PIL6"
    2> $TEMP/tmpmenu

    if [ ! $? = 0 ];then      #---
Mengecek Penekanan Tombol Cancel
        Out_Menu
    fi
}

```

b. Menu Shutdown komputer Klien



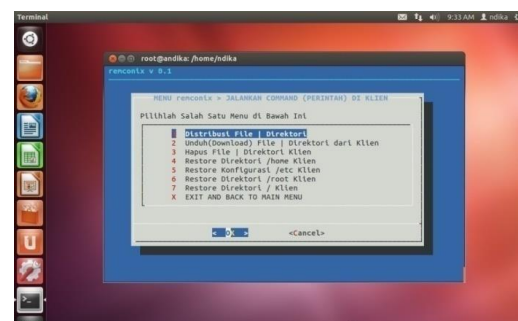
Gambar 9. Menu Shutdown Komputer Klien

Script bash untuk membuat Menu Shutdown Komputer Klien adalah sebagai berikut :

```
PILIH=`cat $TEMP/tmpmenu`
case "$PILIH" in
    1) ### SHUTDOWN KOMPUTER
        KLIEN
        >|$TEMP/errshut
        #redirecting output
        SHUT="1"
        Shut_Client() {
            PILa="SHUTDOWN SEMUA KOMPUTER
        KLIEN"
            PILb="SHUTDOWN DENGAN RANGE
        (JANGKAUAN)"
            PILc="SHUTDOWN BEBERAPA
        KOMPUTER"
            PILd="KOMPUTER YG TIDAK BISA
        SHUTDOWN BELUM ADA"
            PILE="EXIT AND BACK TO MAIN
        MENU"
            dialog --backtitle "$BT" --
            default-item $SHUT --title "$JUDUL >
            $PIL1" --menu "$MTEKS" $DIMEN2 5 1
            "$PILa" 2 "$PILb" 3 "$PILc" \
            4 "$PILd" X "$PILE" 2>
            $TEMP/tmpmenu
            if [ ! $? = 0 ];then
                Main_Menu
            fi
            SHUT=`cat $TEMP/tmpmenu`
            case "$SHUT" in
                1) ### SHUTDOWN SEMUA KOMPUTER
                    KLIEN
                    >|$TEMP/errshut
                    TEKS="\nKomputer yang akan
                    dishutdown adalah komputer no ini ke
                    bawah.\n
                    \nMasukkan NOMOR
                    KOMPUTER TERBESAR "
                    dialog --backtitle "$BT" -
                    -title "$PILa" --inputbox "$TEKS"
                    $DIMEN2 2> $TEMP/bignum
                    if [ ! $? = 0 ];then #---
                    Mengecek Penekanan Tombol Cancel
```

```
Shut_Client
fi
ALL=`cat $TEMP/bignum`
echo "Komputer Yang Gagal
di Shutdown" > $TEMP/allshut
for ((i=1;i<=$ALL;i++));
do
    ssh root@192.168.7.$i
    halt -p >> $TEMP/allshut 2>&1 &
    #
    if [ ! $? = 0
];then
    #
    cat $TEMP/allshut
| cut -f 5 -d ' ' >> $TEMP/errshut
    #
    echo -e
    "Komputer No $i" >> $TEMP/errshut
    #
    fi
done
dialog --backtitle "$BT" -
-sleep 2 --title "Warning" --msgbox
"Harap Menunggu Sebentar" $DIMEN3
Shut_Client
;;
#
echo "Komputer Yang
Gagal di Shutdown / ssh"
#
tail -f
$TEMP/allshut | cut -f 5 -d ' ' 2>
/dev/null
#
Shut_Client
#
if [ ! $? = 0 ];then
#
Shut_Client
#
fi
#
INFO="Daftar
Komputer Yang Tidak Bisa di Shutdown"
#
dialog --
backtitle "$BT" --title "$INFO" --
tailbox $TEMP/errshut $DIMEN2
#
sleep 5s
#
TROUB=`tail -f
$TEMP/shutdown`
#
dialog --backtitle
"$BT" --title "$INFO" --msgbox "`tail
-f $TEMP/errshut`" $DIMEN2
#
dialog --backtitle
"$BT" --title "$INFO" --tailbox
$TEMP/errshut 17 100
```

c. Menu Manajemen Direktori & Transfer File



Gambar 14. Menu Manajemen Direktori dan Transfer File

Script bash untuk membuat Menu Manajemen Direktori dan Transfer File adalah sebagai berikut :

```
MFile="1"
Manaj_File(){
  PILa="Distribusi File |
Direktori"
  PILb="Unduh(Download) File |
Direktori dari Klien"
  PILc="Hapus File | Direktori
Klien"
  PILd="Restore Direktori /home
Klien"
  PILE="Restore Konfigurasi /etc
Klien"
  PILf="Restore Direktori /root
Klien"
  PILg="Restore Direktori /
Klien"
  PILh="EXIT AND BACK TO MAIN
MENU"
  dialog --backtitle "$BT" --
default-item $MFile --title "$JUDUL >
$PIL3" --menu "$MTEKS" $DIMEN1 8 1
"$PILa" \
      2 "$PILb" 3 "$PILc" 4
"$PILd" 5 "$PILE" 6 "$PILf" 7 "$PILg"
X "$PILh" 2> $TEMP/tmpmenu
  MFile=`cat $TEMP/tmpmenu`
  if [ ! $? = 0 ];then
    Main_Menu
  fi
  case "$MFile" in
    1) ### Distribusi File |
Direktori
      TEKS="Masukkan path nama
file atau direktori yang akan di
distribusikan ke klien"
      dialog --backtitle "$BT" -
-title "$PILa" --fselect ~ $DIMEN3
2> $TEMP/tmpfile
      if [ ! $? = 0 ];then
        Manaj_File
      fi
      Manaj_File
      ;;
    2) ### Unduh(Download) File |
Direktori dari Klien
      TEKS="Masukkan path nama
file / direktori klien yang akan di
copy/download"
      dialog --backtitle "$BT" -
-title "$PILb" --inputbox "$TEKS"
$DIMEN2 "/home/" 2> $TEMP/tmpfile
      if [ ! $? = 0 ];then
        Manaj_File
      fi
      SRC=`cat $TEMP/tmpfile`
      TEKS2="Masukkan path
direktori tempat hasil download"
      dialog --backtitle "$BT" -
-title "$TEKS2" --fselect ~ $DIMEN2
2> $TEMP/tmpfile
      if [ ! $? = 0 ];then
        Manaj_File
      fi
    fi
  DST=`cat $TEMP/tmpfile`
  Manaj_File
  ;;
}
```

```
fi
DST=`cat $TEMP/tmpfile`
Manaj_File
;;
```

h. Instalasi Program

Untuk instalasi program ke komputer server langkah yang dilakukan adalah dengan cara berikut ini:

- Ubah atribut file skrip kendali menjadi executable (agar dapat dieksekusi) dengan perintah. Contoh perintahnya:

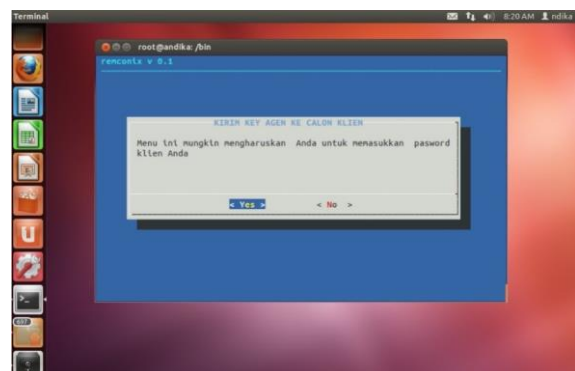
```
# chmod +x remconix
```

- Pindah atau salin file yang sudah diubah atributnya ke salah satu direktori /bin, /sbin, /usr/bin atau /usr/sbin. Contoh perintahnya:

```
# cp remconix /sbin
```

i. Membuka koneksi antara server dengan klien

Setelah aplikasi diinstall di komputer yang berfungsi sebagai server, langkah selanjutnya adalah menghubungkan komputer server dengan komputer klien dengan metode pengiriman key agen dari komputer server ke komputer klien. Untuk menghubungkan dengan komputer klien maka dipilih menu Kirim Key Agen.



Gambar 5. Menu Kirim Key Agen

script untuk pengiriman key agen dari komputer server ke komputer klien adalah sebagai berikut :

```
ETCKONF="/etc/$APPL"
RSAKEY="$ETCKONF/remconix-rsa"
```

```

SSSHDIR="/root/.ssh"

Make_Keys() {
    ssh-keygen -t rsa -N "" -C
    remconix-by-oman -f $RSAKEY
    cp $RSAKEY.pub
    $ETCKONF/authorized_keys
    md5sum `ls $ETCKONF` >
    $ETCKONF/keysum
    cp $RSAKEY $SSSHDIR/id_rsa
    chmod 644 $SSSHDIR/id_rsa
    TEKS="Jangan Mengubah file
Berikut ini \n
    $RSAKEY \n
    $RSAKEY.pub \n
    $ETCKONF/authorized_keys \n
    $ETCKONF/keysum
    \nKarena dapat berakibat Tidak
Berjalannya program $APPL Sebagaimana
Mestinya"
    echo -e $TEKS >
    $ETCKONF/README
}

#Check aplikasi sudah
terinstall di komputer atau belum
Check_App() {
    if [ ! -d $ETCKONF ] ; then
        TEKS="\nUntuk menjalankan
program ini Anda perlu menginstallnya
dan membuat private key
pada komputer ini, Apakah Anda
menginginkan Komputer ini sebagai
Pusat Kendali dari program ini"
        if [ $? = 0 ] ; then
            dialog --backtitle "$BT" --
            title "Need Installing This
Application" --yesno "$TEKS" 10 60
            TEKS="\nUntuk menjalankan
program ini Anda perlu menginstall
aplikasi dialog terlebih dahulu"
            echo -e "$TEKS"
            if [ $? = 0 ] ; then
                mkdir $ETCKONF
                Make_Keys
            fi
        else
            exit
        fi
    else
        md5sum -c $ETCKONF/keysum
        2> /dev/null
        if [ ! $? = 0 ] ; then
            TEKS=""
            rm -fr $TEMP/*
            fi
        fi
    }
}

```

KESIMPULAN DAN SARAN

Kesimpulan

Berdasarkan hasil yang dicapai saat ini berkaitan dengan permasalahan administrasi data

dan mekanisme menghidupkan dan mematikan komputer melalui jaringan maka dapat disimpulkan bahwa :

- a. Adanya manfaat yang cukup signifikan dengan adanya aplikasi yang mendukung untuk menghidupkan dan mematikan komputer yang berbasis sistem operasi Linux melalui jaringan komputer sehingga menghemat waktu.
- b. Dengan memanfaatkan fasilitas internal yaitu SSH dan perintah-perintah yang sudah ada didalam sistem operasi Linux yang dirangkai menjadi sebuah kode dapat dimanfaatkan untuk membangun sebuah aplikasi yang sifatnya open source.

Saran

Fasilitas dan perintah-perintah yang terdapat didalam sistem operasi Linux sudah lengkap seharusnya dapat dieksplorasi lebih dalam agar dapat diciptakan aplikasi yang lebih banyak manfaatnya lagi dalam membantu kelancaran proses perawatan laboratorium.

DAFTAR PUSTAKA

- [1] Ginting, "Analisis Kestabilan Sistem Kendali Jaringan Terhadap Tundaan," Universitas Diponegoro : Fakultas Matematika dan Ilmu Pengetahuan Alam, Semarang, 2009.
- [2] M. L. Iswan, "Implementasi virtual private network (VPN) remote access dengan linux openswan," UIN Syarif Hidayatullah Jakarta : Teknik Informatika Fakultas Sains dan Teknologi, Jakarta, 2010.
- [3] Y. N. Kunang, "Laboratorium Komputer Virtual Berbasis Linux," *Jurnal Ilmiah Matrik*, vol. 11, no. 1 April 2009, pp. 10-23, 2009.
- [4] Noprianto, *Secure Shell dan Openssh*, Jakarta: Info Linux, 2004, pp. 65-67.
- [5] Cooper, "The Linux Documentation Project," 10 March 2014. <http://tldp.org/LDP/abs/abs-guide.pdf>. [Diakses pada tanggal 23 Mei 2014].

- [6] A. Kadir, Seri Panduan Belajar Unix: Korn Shell, Jakarta: Elex Media Komputindo, 1995.
- [7] B. Susanto, Unix & Pemrograman Script, Yogyakarta: J & J Learning, 2001.