**JOBSHET 5**

**STRUKTUR DIREKTORI & OPERASI FILE**

Nama: Rizqi Bagus Andrean

Kelas: TI-1D

Absen: 25

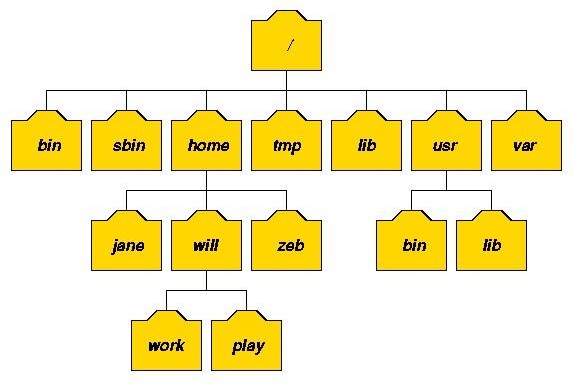
Capaian Pembelajaran

* Memahami organisasi file dalam direktory pada sistem operasi Linux
* Memahami konsep link dan Symbolic
* Menciptakan dan manipulasi directory

# DASAR TEORI:

1. **ORGANISASI FILE**

Sistem file pada Linux menyerupai pepohonan (tree), yaitu dimulai dari root, kemudian direktori dan sub dirrektori. Sistem file pada Linux diatur secara hirarkhikal, yaitu dimulai dari root dengan symbol “/” seperti Gambar 3.1.

User dapat menciptakan File dan Direktori mulai dari root ke bawah. Direktori adalah file khusus, yang berisi nama file dan INODE (pointer yang menunjuk ke data / isi file tersebut). Secara logika, Direktori dapat berisi File dan Direktori lagi (disebut juga Subdirektori).

Gambar 1.3 Struktur direktori pada Linux

# DIREKTORY STANDAR

|  |  |
| --- | --- |
| /etc | Berisi file administrative (konfigrasi dll) dan file executable atau script yang berguna untuk administrasi system. |
| /dev | Berisi file khusus yang merepresentasikan peralatan hardware seperti memori, disk, printer, tape, floppy, jaringan dll. |
| /bin | Berisi utilitas sistem level rendah (binary) . |
| /sbin | Berisi utilitas sistem untuk superuser (untuk membentuk administrasi sistem). |
| /usr/sbin | Berisi utilitas sistem dan program aplikasi level tinggi. |
| /usr/bin |
| /usr/lib | Berisi program library yang diperlukan untuk kompilasi program (misalnya C). Berisi instruksi (command) misalnya untuk Print Spooler (lpadmin) dll. |
| /tmp | Berisi file sementara, yang pada saat Bootstrap akan dihapus (dapat digunakan oleh sembarang user). |
| /boot | Berisi file yang sangat penting untuk proses bootstrap. Kernel vmlinuz disimpan di direktori ini. |
| /proc | Berisi informasi tentang kernel Linux, proses dan virtual system file. |
| /var | Direktori variable, artinya tempan penyimpanan LOG (catatan hasil output program), file ini dapat membengkak dan perlu dimonitor perkembangannya. |
| /home | Berisi direktori untuk pemakai Linux (pada SCO diletakkan pada /usr) |
| /mnt | Direktori untuk mounting system file |
| /root | Home direktori untuk superuser (root) |
| /usr/bin/X11 | Symbolic link ke /usr/X11R6/bin, program untuk X-Window |
| /usr/src | Source code untuk Linux |
| /opt | Option, direktori ini biasanya berisi aplikasi tambahan (“addon”) seperti Netscape Navigator, kde, gnome, applix dll. |
| /var | Direktori variable, artinya tempan penyimpanan LOG (catatan hasil output program), file ini dapat membengkak dan perlu dimonitor perkembangannya. |
| /home | Berisi direktori untuk pemakai Linux (pada SCO diletakkan pada /usr) |
| /mnt | Direktori untuk mounting system file |
| /root | Home direktori untuk superuser (root) |
| /usr/bin/X11 | Symbolic link ke /usr/X11R6/bin, program untuk X- Window |
| /usr/src | Source code untuk Linux |
| /opt | Option, direktori ini biasanya berisi aplikasi tambahan (“addon”) seperti Netscape Navigator, kde, gnome, applix dll. |

**Direktori /etc**

Berisi file yang berhubungan dengan administrasi system, maintenance script, konfigurasi, security dll. Hanya superuser yang boleh memodifikasi file yang berada di drektori ini. Subdirektori yang sering diakses pada direktori **/etc** antara lain :

* + httpd, apache web server.
  + ppp, point to point protocol untuk koneksi ke Internet.
  + rc.d atau init.d, inisialisasi (startup) dan terminasi (shutdown) proses di Linux dengan konsep runlevel.
  + cron.d, rincian proses yang dieksekusi dengan menggunakan jadwal(time dependent process)
  + FILES, file security dan konfigurasi meliputi : *passwd, hosts, shadow, ftpaccess, inetd.conf, lilo.conf, motd, printcap, profile, resolv.conf, sendmail.cf, syslog.conf, dhcp.conf, smb.conf, fstab*.

**Direktori** /dev

Konsep Unix dan Linux adalah memperlakukan peralatan hardware sama seperti penanganan file. Setiap alat mempunyai nama file yang disimpan pada direktori /dev.

|  |  |
| --- | --- |
| **Peralatan** | **Direktori** |
| **Floppy** | /dev/fd0 |
| **Harddisk** | IDE : /dev/had, /dev/hdb, /dev/hdc,  /dev/hdd  SCSI : /dev/sda, /dev/sdb, /dev/sdc |
| **CDROM** | SCSI : /dev/scd0, /dev/scd1  IDE : /dev/gscd, /dev/sonycd  Universal : /dev/cdrom (link dari actual cdrom ide atau scsi) |
| **Mouse** | PS2 : /dev/lp0  Universal : /dev/mouse |
| **Parallel Port** | LPT1 : /dev/lp0  LPT2 : /dev/lp1 |
| **Serial Port** | COM1 : /dev/ttyS0  COM2 : /dev/ttyS1  Universal : /dev/modem (link dari S0 atau S1) |

**Direktori** /proc

Direktori /proc adalah direktori yang dibuat diatas RAM (Random Access Memory) dengan system file yang diatur oleh kernel. /proc berisi nomor proses dari system dan nama driver yang aktif di system. Semua direktori berukuran 0 (kosong) kecuali file kcore dan self. Setiap nomor yang ada pada direktori tsb merepresentasikan PID (Process ID).

# TIPE FILE

Pada Linux terdapat 6 buah tipe file yaitu

* + Ordinary file
  + Direktori
  + Block Device (Peralatan I/O)

Merupakan representasi dari peralatan hardware yang menggunakan transmisi data per block (misalnya 1 KB block), seperti disk, floppy, tape.

* + Character Device (Peralatan I/O)

Merupakan representasi dari peralatan hardware yang menggunakan transmisi data karakter per karakter, seperti terminal, modem, plotter dll

* + Named Pipe (FIFO)

File yang digunakan secara intern oleh system operasi untuk komunikasi antar proses

* + Link File

# PROPERTI FILE

File mempunyai beberapa atribut, antara lain :

* + Tipe file : menentukan tipe dari file, yaitu:

|  |  |
| --- | --- |
| **Karakter** | **Arti** |
| **-** | File biasa |
| **d** | Direktori |
| **l** | Symbolic link |
| **b** | Block special file |
| **c** | Character special file |
| **s** | Socket link |
| **p** | FIFO |

* + Ijin akses : menentukan hak user terhadap file ini.
  + Jumlah link : jumlah link untuk file ini.
  + Pemilik (Owner) : menentukan siapa pemilik file ini
  + Group : menentukan group yang memiliki file ini
  + Jumlah karakter : menentukan ukuran file dalam byte
  + Waktu pembuatan : menentukan kapan file terakhir dimodifikasi
  + Nama file : menentukan nama file yang dimaksud.

# NAMA FILE

Nama file maksimal terdiri dari 255 karakter berupa alfanumerik dan beberapa karakter spesial yaitu garis bawah, titik, koma dan lainnya kecuali spasi dan karakter “&”, “;”, “|”, “?”, “`”, “””, “’”, “[“, “]”, “(“, “)”, “$”, “<”, “>”, “{“, “}”, “^”, “#”, “\”, “/”. Linux membedakan huruf kecil dengan huruf besar (case sensitive). Contoh nama file yang benar :

Abcde5434 3 prog.txt

PROG.txt Prog.txt,old report\_101,v2.0.1

5-01.web.html

# SIMBOLIC LINK

Link adalah sebuah teknik untuk memberikan lebih dari satu nama file dengan data yang sama.

Bila file asli dihapus, maka data yang baru juga terhapus. Format dari Link :

ln *fileAsli fileDuplikat*

fileDuplikat disebut *hard link* dimana kedua file akan muncul identik (*link*

*count* = 2) Bila fileAsli atau fileDuplikat diubah perubahan akan terjadi pada file lainna.

Symbolic Link diperlukan bila file tersebut di “link” dengan direktori /file yang berada pada

partisi yang berbeda. Tipe file menjadi l (link) dan file tersebut menunjuk ke tempat asal. Format :

ln –s */FULLPATH/fileAsli /FULLPATH/fileDuplikat*

Pilihan –s (*shortcut*) merupakan bentuk *soft link* dimana jumlah *link count* pada file asal tidak akan berubah. Pada bentuk *soft link, symbolic link* dapat dilakukan pada file yang tidak ada, sedangkan pada *hard link* tidak dimungkinkan. Perbedaan lain, *symbolic link* dapat dibentuk melalui media disk atau partisi yang berbeda dengan *soft link,* tetapi pada *hard link* terbatas pada partisi disk yang sama.

# MELIHAT ISI FILE

Untuk melihat jenis file menggunakan format :

file *filename(s)*

Isi file akan dilaporkan dengan deskripsi level tinggin seperti contoh berikut

$ file myprog.c letter.txt webpage.html myprog.c: C program text letter.txt: ASCII text webpage.html: HTML document text

Perintah ini dapat digunakan secara luas untuk file yang kadang membingungkan, misalnya antara kode C++ dan Java.

# MENCARI FILE

Jika ingin melihat bagaimana pohon direktori dapat digunakan perintah

* + find

Format : find *directory* –name *targetfile* -print

Akan melihat file yang bernama *targetfile* (bisa berupa karakter wildcard)

* + which

Format : which *command*

Untuk mengetahui letak system utility

* + locate

Format : locate *string*

Akan mencari file pada semua directori dengan lebih cepat dan ditampilkan dengan path yang penuh.

# MENCARI TEXT PADA FILE

Untuk mencari text pada file digunakan perintah grep (*General Regular Expression Print*) dengan format perintah grep *option pattern files.* Grep akan mencari file yang bernama sesuai pattern yang diberikan dan akan menampilkan baris yang sesuai.

# TUGAS PENDAHULUAN:

Jawablah pertanyaan-pertanyaan di bawah ini :

1. Apa yang dimaksud perintah-perintah direktory : pwd, cd, mkdir, rmdir.
   * pwd : Perintah yang digunakan untuk melihat posisi di direktori mana user berada saat ini.
   * cd : Perintah yang digunakan untuk masuk ke dalam direktori, bisa juga digunakan untuk keluar dari direktori.
   * mkdir : Perintah yang digunakan untuk membuat direktori.
   * rmdir : Perintah yang digunakan untuk menghapus direktori.
2. Apa yang dimaksud perintah-perintah manipulasi file : cp, mv dan rm (Sertakan format yang digunakan)
   * cp : Perintah yang digunakan untuk mengcopy file. Format : cp <nama\_file> <direktori\_tujuan>
   * mv : Perintah yang digunakan untuk memindahkan / cut file.

Format : mv <nama\_file> <direktori\_tujuan>

* + rm : Perintah yang digunakan untuk menghapus file. Format : rm <nama\_file>

1. Jelaskan perbedaan *Symbolic link* menggunakan *hard link(direct)* dan *soft link (indirect)*.
   * *Hard Link* : Digunakan untuk membuat link pada sebuah file, dan hasil link dari file tersebut sama persis dengan yang aslinya. Dan bila file link/yang asli diubah/dihapus file lain yang terhubung juga akan berubah. Untuk linknya juga terbatas hanya dalam satu partisi.
   * *Soft Link* : Digunakan untuk membuat link namun dalam bentuk shortcut. Dan bila menggunakan soft link, Link Count file yang asli tidak akan berubah, berbeda dengan hard link. Untuk linknya bisa berbeda partisi.
2. Tuliskan maksud perintah-perintah : file, find, which, locate dan grep.
   * file : Digunakan untuk melihat jenis file.
   * find : Digunakan untuk mencari file.
   * which : Digunakan untuk mengetahui letak direktori yang user inginkan.
   * locate : Digunakan untuk mencari file pada semua direktori dengan lebih cepat dan ditampilkan dengan path yang penuh.
   * grep : Digunakan untuk mencari text pada file Merupakan representasi dari peralatan hardware yang menggunakan transmisi data per block (misalnya 1 KB block), seperti disk, floppy, tape.
   * Character Device (Peralatan I/O)

Merupakan representasi dari peralatan hardware yang menggunakan transmisi data karakter per karakter, seperti terminal, modem, plotter dll

* + Named Pipe (FIFO)

File yang digunakan secara intern oleh system operasi untuk komunikasi antar proses

* + Link File

# PROPERTI FILE

File mempunyai beberapa atribut, antara lain :

* + Tipe file : menentukan tipe dari file, yaitu:

|  |  |
| --- | --- |
| **Karakter** | **Arti** |
| **-** | File biasa |
| **d** | Direktori |
| **l** | Symbolic link |
| **b** | Block special file |
| **c** | Character special file |
| **s** | Socket link |
| **p** | FIFO |

* + Ijin akses : menentukan hak user terhadap file ini.
  + Jumlah link : jumlah link untuk file ini.
  + Pemilik (Owner) : menentukan siapa pemilik file ini
  + Group : menentukan group yang memiliki file ini
  + Jumlah karakter : menentukan ukuran file dalam byte
  + Waktu pembuatan : menentukan kapan file terakhir dimodifikasi
  + Nama file : menentukan nama file yang dimaksud.

# NAMA FILE

Nama file maksimal terdiri dari 255 karakter berupa alfanumerik dan beberapa karakter spesial yaitu garis bawah, titik, koma dan lainnya kecuali spasi dan karakter “&”, “;”, “|”, “?”, “`”, “””, “’”, “[“, “]”, “(“, “)”, “$”, “<”, “>”, “{“, “}”, “^”, “#”, “\”, “/”. Linux membedakan huruf kecil dengan huruf besar (case sensitive). Contoh nama file yang benar :

Abcde5434 3 prog.txt

PROG.txt Prog.txt,old report\_101,v2.0.1

5-01.web.html

# SIMBOLIC LINK

Link adalah sebuah teknik untuk memberikan lebih dari satu nama file dengan data yang sama.

Bila file asli dihapus, maka data yang baru juga terhapus. Format dari Link :

ln *fileAsli fileDuplikat*

fileDuplikat disebut *hard link* dimana kedua file akan muncul identik (*link*

*count* = 2) Bila fileAsli atau ileDuplikat diubah perubahan akan terjadi pada file lainna.

Symbolic Link diperlukan bila file tersebut di “link” dengan direktori /file yang berada pada

partisi yang berbeda. Tipe file menjadi l (link) dan file tersebut menunjuk ke tempat asal. Format :

ln –s */FULLPATH/fileAsli /FULLPATH/fileDuplikat*

Pilihan –s (*shortcut*) merupakan bentuk *soft link* dimana jumlah *link count* pada file asal tidak akan berubah. Pada bentuk *soft link, symbolic link* dapat dilakukan pada file yang tidak ada, sedangkan pada *hard link* tidak dimungkinkan. Perbedaan lain, *symbolic link* dapat dibentuk melalui media disk atau partisi yang berbeda dengan *soft link,* tetapi pada *hard link* terbatas pada partisi disk yang sama.

# MELIHAT ISI FILE

Untuk melihat jenis file menggunakan format :

file *filename(s)*

Isi file akan dilaporkan dengan deskripsi level tinggin seperti contoh berikut

$ file myprog.c letter.txt webpage.html myprog.c: C program text letter.txt: ASCII text webpage.html: HTML document text

Perintah ini dapat digunakan secara luas untuk file yang kadang membingungkan, misalnya antara kode C++ dan Java.

# MENCARI FILE

Jika ingin melihat bagaimana pohon direktori dapat digunakan perintah

* + find

Format : find *directory* –name *targetfile* -print

Akan melihat file yang bernama *targetfile* (bisa berupa karakter wildcard)

* + which

Format : which *command*

Untuk mengetahui letak system utility

* + locate

Format : locate *string*

Akan mencari file pada semua directori dengan lebih cepat dan ditampilkan dengan path yang penuh.

# MENCARI TEXT PADA FILE

Untuk mencari text pada file digunakan perintah grep (*General Regular Expression Print*) dengan format perintah grep *option pattern files.* Grep akan mencari file yang bernama sesuai pattern yang diberikan dan akan menampilkan baris yang sesuai.

# TUGAS PENDAHULUAN:

Jawablah pertanyaan-pertanyaan di bawah ini :

1. Apa yang dimaksud perintah-perintah direktory : pwd, cd, mkdir, rmdir.
   * pwd : Perintah yang digunakan untuk melihat posisi di direktori mana user berada saat ini.
   * cd : Perintah yang digunakan untuk masuk ke dalam direktori, bisa juga digunakan untuk keluar dari direktori.
   * mkdir : Perintah yang digunakan untuk membuat direktori.
   * rmdir : Perintah yang digunakan untuk menghapus direktori.
2. Apa yang dimaksud perintah-perintah manipulasi file : cp, mv dan rm (Sertakan format yang digunakan)
   * cp : Perintah yang digunakan untuk mengcopy file. Format : cp <nama\_file> <direktori\_tujuan>
   * mv : Perintah yang digunakan untuk memindahkan / cut file.

Format : mv <nama\_file> <direktori\_tujuan>

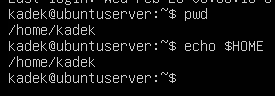
* + rm : Perintah yang digunakan untuk menghapus file. Format : rm <nama\_file>

1. Jelaskan perbedaan *Symbolic link* menggunakan *hard link(direct)* dan *soft link (indirect)*.
   * *Hard Link* : Digunakan untuk membuat link pada sebuah file, dan hasil link dari file tersebut sama persis dengan yang aslinya. Dan bila file link/yang asli diubah/dihapus file lain yang terhubung juga akan berubah. Untuk linknya juga terbatas hanya dalam satu partisi.
   * *Soft Link* : Digunakan untuk membuat link namun dalam bentuk shortcut. Dan bila menggunakan soft link, Link Count file yang asli tidak akan berubah, berbeda dengan hard link. Untuk linknya bisa berbeda partisi.
2. Tuliskan maksud perintah-perintah : file, find, which, locate dan grep.
   * file : Digunakan untuk melihat jenis file.
   * find : Digunakan untuk mencari file.
   * which : Digunakan untuk mengetahui letak direktori yang user inginkan.
   * locate : Digunakan untuk mencari file pada semua direktori dengan lebih cepat dan ditampilkan dengan path yang penuh.
   * grep : Digunakan untuk mencari text pada file

PRAKTIKUM

## Percobaan 1 : Direktory

* 1. Melihat direktori HOME



Analisa:

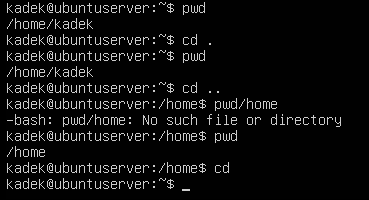
$ pwd  Digunakan untuk melihat direktori yang bekerja pada saat ini. Pada percobaan didapatkan hasil /home/kadek artinya saat ini user bekerja pada direktori kadek yang berada didalam direktori home.



$ echo $HOME  Digunakan untuk melihat direktori HOME. Pada hasil percobaan didapatkan bahwa di dalam direktori home terdapat direktori kadek.



* 1. Melihat direktori aktual dan parent direktori



Analisa:

$ pwd  Digunakan untuk melihat direktori yang bekerja pada saat ini. Pada percobaan didapatkan hasil /home/kadek artinya saat ini user bekerja pada direktori /kadek yang berada didalam direktori /home.

$ cd .  Perintah cd digunakan untuk berpindah ke direktori lain. Namun jika hanya diikuti dengan tanda satu titik saja maka perintah ini tidak akan mengeksekusi apapun. Untuk membuktikannya mari lihat hasil perintah $ pwd.

$ pwd  Didapatkan hasil /home/kadek seperti hasil sebelumnya artinya perintah $ cd

. tidak memberikan eksekusi apapun.

$ cd ..  Perintah cd digunakan untuk berpindah ke direktori lain. Ketika perintah ini diikuti tanda dua titik maka akan mengeksekusi perintah yaitu kembali ke direktori sebelumnya.

$ pwd  Didapatkan hasil /home artinya user telah meninggalkan direktori /kadek dan menuju direktori sebelumnya yaitu /home.

$ cd  Perintah cd digunakan untuk berpindah ke direktori lain. Pada hasil percobaan menunjukkan bahwa user telah pindah dari /home ke direktori awal tempat user bekerja yaitu

/kadek.

* 1. Membuat satu direktori, lebih dari satu direktori atau sub director

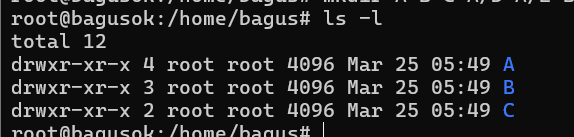
Analisa:

$ pwd  Digunakan untuk melihat direktori yang bekerja pada saat ini. Pada percobaan didapatkan hasil /home/kadek artinya saat ini user bekerja pada direktori /kadek yang berada didalam direktori /home.

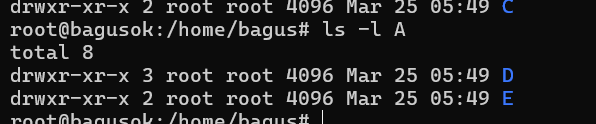
$ mkdir A B C A/D A/E B/F A/D/A  Perintah mkdir digunakan untuk membuat direktori. Pada percobaan dibuat tiga direktori yaitu /A, /B, dan /C. Kemudian didalam direktori /A dibuat subdirektori /D dan subdirektori /E. Lalu, didalam direktori /B dibuat subdirektori /F. Terakhir direktori /A yang terdapat subdirektori /D dibuat subdirektori lagi yaitu /A.



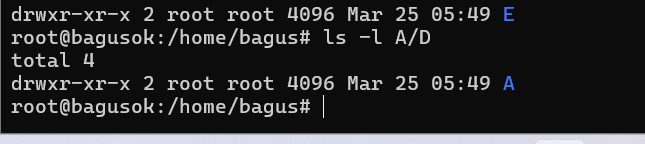
$ ls -l  Digunakan untuk menampilkan seluruh isi dari direktori /home. Dan terlihat bahwa tiga direktori yaitu /A, /B, dan /C sudah terdapat disana.



$ ls -l A  Digunakan untuk melihat isi direktori /A, dan terlihat bahwa dua subdirektori yaitu /D dan /E sudah terdapat disana.



$ ls -l A/D  Digunakan untuk menampilkan isi subdirektori /D pada direktori /A dan terlihat bahwa subdirektori /A sudah terdapat disana.



Pohon dari struktur file dan direktori :



Home



kadek

A

B

C

D

E

F

A

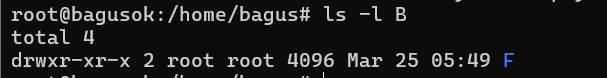
* 1. Menghapus satu atau lebih direktori hanya dapat dilakukan pada direktori kosong dan hanya dapat dihapus oleh pemiliknya kecuali bila diberikan ijin aksesnya

Analisa:

$ rmdir B (Terdapat pesan error, mengapa ?)  Perintah rmdir digunakan untuk menghapus direktori. rmdir B error karena direktori tidak kosong sehingga menyebabkan direktori tidak dapat dihapus.

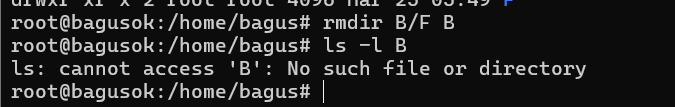


$ ls -l B  Digunakan untuk melihat isi direktori /B, dan terlihat bahwa direktori /F sudah terdapat disana.



$ rmdir B/F B  Digunakan untuk menghapus direktori B dan subdirektori F.

$ ls -l B (Terdapat pesan error, mengapa ?)  Terdapat pesan error karena direktori B sudah dihapus pada perintah sebelumnya.



* 1. Navigasi direktori dengan instruksi cd untuk pindah dari satu direktori ke direktori lain.

Analisa:

$ pwd  Digunakan untuk melihat direktori yang bekerja pada saat ini. Pada percobaan didapatkan hasil /home/kadek artinya saat ini user bekerja pada direktori /kadek yang berada didalam direktori /home.

$ ls -l  Digunakan untuk menampilkan seluruh isi dari direktori /home. Dan terlihat bahwa dua direktori yaitu /A dan /C masih terdapat disana, sedangkan direktori /B sudah dihapus.

$ cd A  Digunakan digunakan untuk berpindah kedirektori /A.

$ pwd  Digunakan untuk melihat direktori yang bekerja pada saat ini. Pada percobaan didapatkan hasil /home/kadek/A artinya saat ini user bekerja pada subdirektori /A yang berada didalam subdirektori /kadek yang berada didalam direktori /home.

$ cd ..  Perintah cd digunakan untuk berpindah ke direktori lain. Ketika perintah ini diikuti tanda dua titik maka akan mengeksekusi perintah yaitu kembali ke direktori sebelumnya.

$ pwd  Digunakan untuk melihat direktori yang bekerja pada saat ini. Pada percobaan didapatkan hasil /home/kadek artinya saat ini user bekerja pada direktori /kadek yang berada didalam direktori /home.

$ cd /home/<user>/C  Digunakan digunakan untuk berpindah ke subdirektori /C yang berada didalam subdirektori /kadek yang berada didalam direktori /home.

$ pwd  Digunakan untuk melihat direktori yang bekerja pada saat ini. Pada percobaan didapatkan hasil /home/kadek/C artinya saat ini user bekerja pada subdirektori /C yang berada didalam subdirektori /kadek yang berada didalam direktori /home.

$ cd /<user/C (Terdapat pesan error, mengapa ?)  Terjadi error karena pada tidak ada nama atau direktori /kadek/C tempat user akan pindah.

$ pwd  Digunakan untuk melihat direktori yang bekerja pada saat ini. Pada percobaan didapatkan hasil /home/kadek/C artinya saat ini user bekerja pada subdirektori /C yang berada didalam subdirektori /kadek yang berada didalam direktori /home.

## Percobaan 2 : Manipulasi file

1. Perintah cp untuk mengkopi file atau seluruh direktori

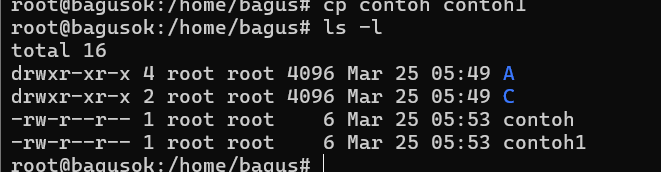
Analisa:

$ cat > contoh Membuat sebuah file[Ctrl-d]  Digunakan untuk menunjukkan output dari file contoh yang isinya inputnya dulu oleh user. Isi dari file contoh yaitu kalimat “Membuat sebuah file”.

$ cp contoh contoh1  Digunakan untuk menyalin isi dari contoh ke contoh1.



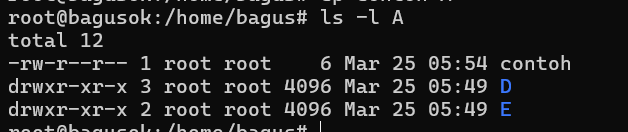
$ ls -l  Digunakan untuk menampilkan seluruh isi dari direktori /home. Dan terlihat bahwa dua file yaitu contoh dan contoh1 sudah terdapat disana.



$ cp contoh A  Digunakan untuk menyalin file contoh ke direktori /A.

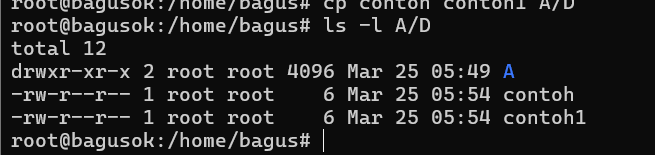


$ ls –l A  Digunakan untuk menampilkan seluruh isi dari direktori /A. Dan terlihat bahwa file yaitu contoh sudah terdapat disana.



$ cp contoh contoh1 A/D  Digunakan untuk menyalin dua file yaitu contoh dan contoh1 ke subdirektori /D pada direktori /A.

$ ls –l A/D  Digunakan untuk menampilkan seluruh isi dari direktori A/D. Dan terlihat bahwa dua file yaitu contoh dan contoh1 sudah terdapat disana.

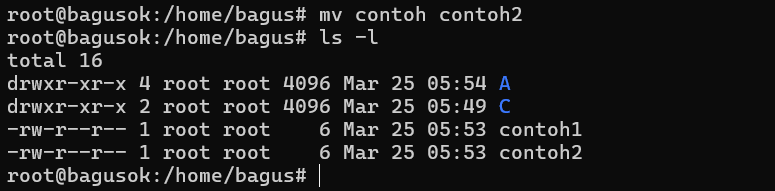


1. Perintah mv untuk memindah file

Analisa:

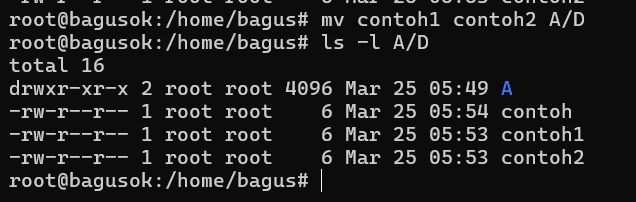
$ mv contoh contoh2  Digunakan untuk memindahkan file contoh ke file contoh2

$ ls -l  Digunakan untuk menampilkan seluruh isi dari direktori /home. Dan terlihat bahwa file yaitu contoh sudah berubah menjadi contoh2.

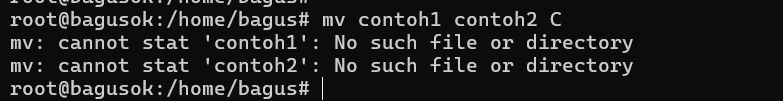


$ mv contoh1 contoh2 A/D  Digunakan untuk memindahkan file contoh1 dan contoh2 dari subdirektori /D di direktori /A.

$ ls –l A/D  Digunakan untuk menampilkan seluruh isi dari direktori A/D. Dan terlihat bahwa dua file yaitu contoh1 dan contoh2 sudah terdapat disana.

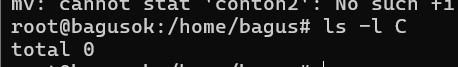


$ mv contoh contoh1 C  Terdapat pesan error karena tidak ada file apapun pada direktori /C.



$ ls –l C  Digunakan untuk menampilkan seluruh isi dari direktori /C. Dan terlihat

bahwa “total 0” artinya tidak ada file disana.

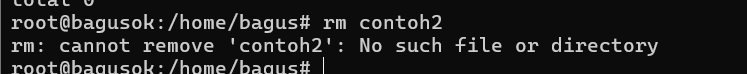


1. Perintah rm untuk menghapus file

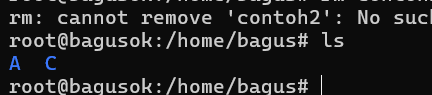
Analisa:

$ rm contoh2  Terdapat pesan error karena file contoh2 tidak terdapat di direktori

/home, hal ini disebabkan file contoh2 telah dipindahkan ke direktori A/D.

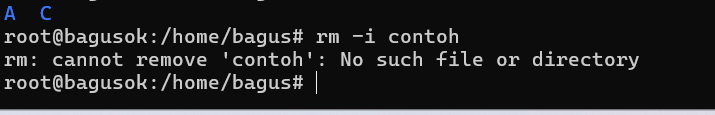


$ ls -l  Digunakan untuk menampilkan seluruh isi dari direktori /home. Dan terlihat bahwa tidak ada file contoh2.

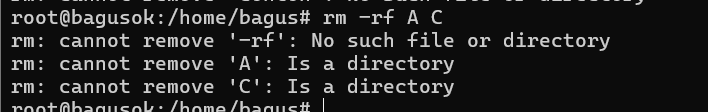


$ rm –i contoh  Terdapat pesan error karena file contoh tidak terdapat di direktori

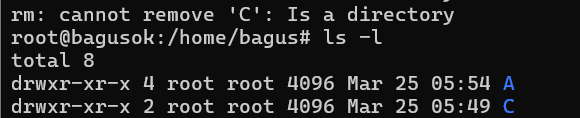
/home, hal ini disebabkan file contoh telah diubah namanya menjadi contoh2 dan dipindahkan ke direktori A/D.



$ rm –rf A C  Digunakan untuk menghapus direktori /A dan /C.



$ ls -l  Digunakan untuk menampilkan seluruh isi dari direktori /home. Dan terlihat bahwa tidak ada direktori /A dan /C.



## Percobaan 3 : Symbolic Lin

1. Membuat shortcut (file link)

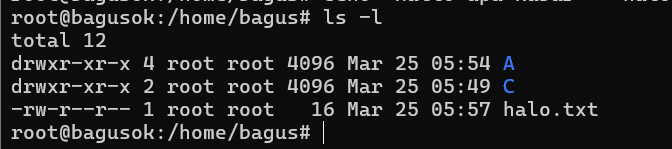
Analisa:

$ echo "Hallo apa kabar" > halo.txt  Digunakan untuk membuat kalimat

“Hallo apa kabar” yang akan dimasukan ke dalam file halo.txt



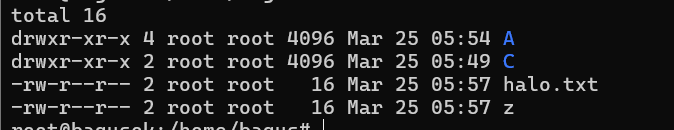
$ ls -l  Digunakan untuk menampilkan seluruh isi dari direktori /home. Dan terlihat bahwa file halo.txt sudah terdapat disana.



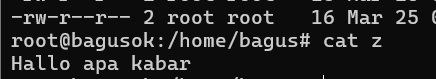
$ ln halo.txt z  Digunakan untuk membuat link pada direktori home yang berfungsi untuk memberikan dari satu nama file dengan data yang sama. Dimana file isi halo.txt akan sama file z.



$ ls -l  Digunakan untuk menampilkan seluruh isi dari direktori /home. Dan terlihat bahwa file z sudah terdapat disana.

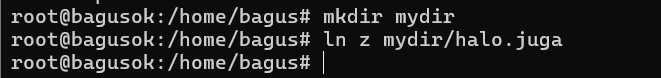


$ cat z  Digunakan untuk melihat isi file z

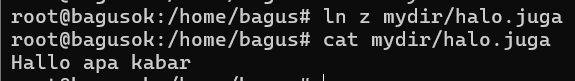


$ mkdir mydir  Perintah mkdir digunakan untuk membuat direktori. Pada percobaan dibuat direktori yaitu mydir.

$ ln z mydir/halo.juga  Digunakan untuk membuat link pada direktori home. Dimana isi file z akan sama dengan file halo.juga, kemudian file halo.juga akan disimpan ke dalam direktori /mydir.

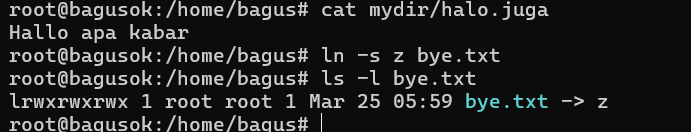


$ cat mydir/halo.juga  Digunakan untuk melihat isi file halo.juga yang terdapat dalam direktori /mydir.

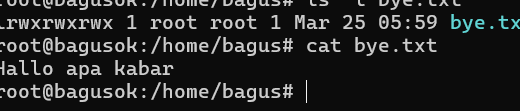


$ ln -s z bye.txt  Digunakan untuk membuat link pada direktori home yang berfungsi untuk memberikan dari satu nama file dengan data yang sama. Dimana file isi bye.txt akan sama file z.

$ ls -l bye.txt  Digunakan untuk menampilkan seluruh isi dari file bye.txt



$ cat bye.txt  Digunakan untuk melihat isi file bye.txt

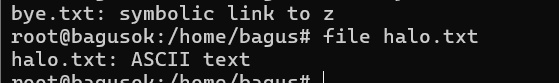


## Percobaan 4 : Melihat Isi File

Analisa:

$ ls –l  Digunakan untuk menampilkan seluruh isi dari direktori /home.

$ file halo.txt  Digunakan untuk melihat informasi file halo.txt berupa ASCII text.



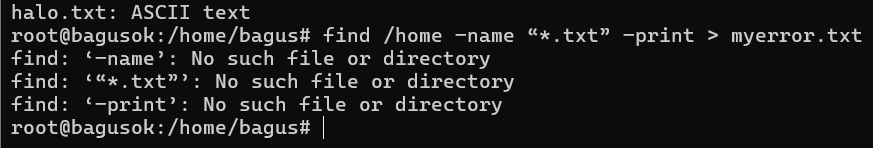
$ file bye.txt  Digunakan untuk melihat informasi bye.txt yang merupakan symbolic link to z.



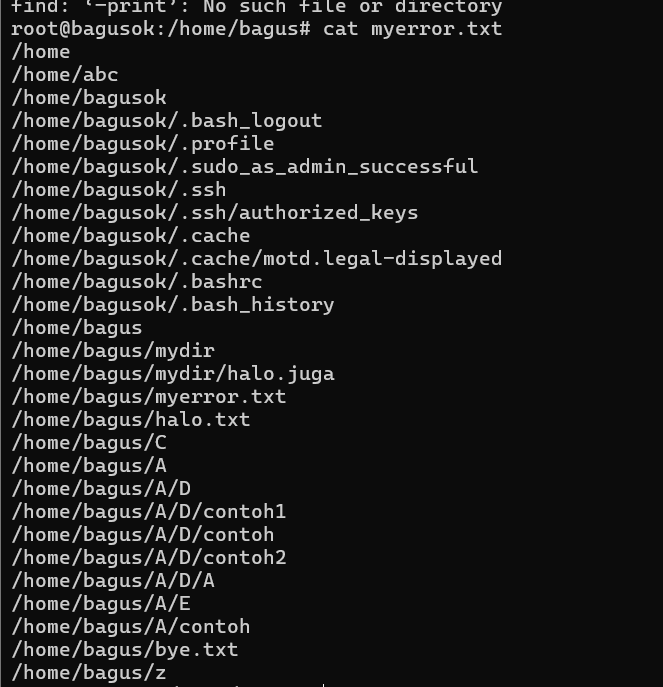
## Percobaan 5 : Mencari file

Analisa:

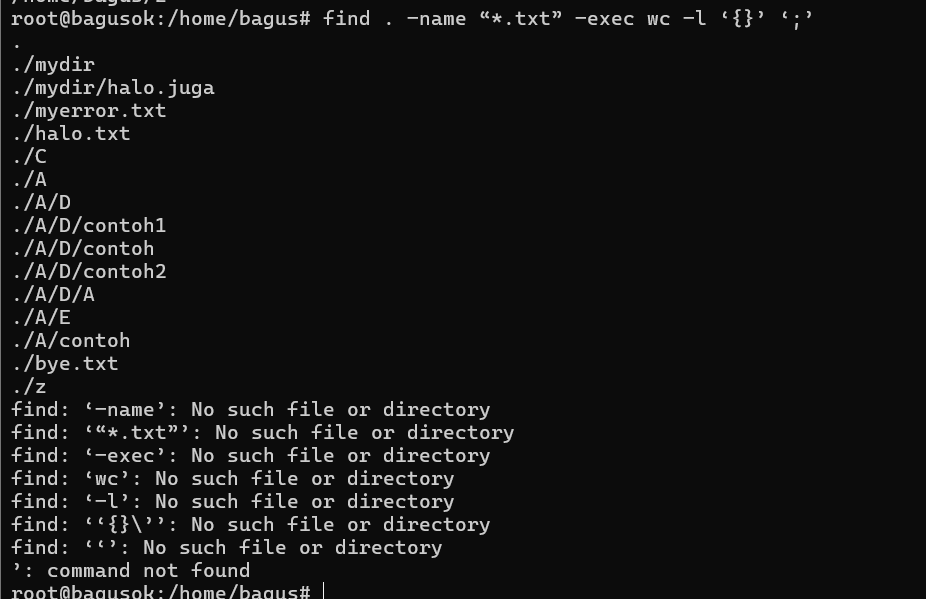
$ find /home –name “\*.txt” –print > myerror.txt  Digunakan untuk mencari letak file sesuai dengan ektensi file. Perintah tersebut akan mencari letak file dengan ektensi “\*.txt” yang terdapat pada direktori /home kemudian hasil pencarian tersebut akan dimasukan ke dalam file bernama “myerror.txt”.



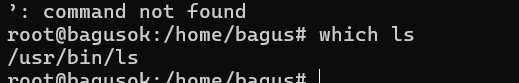
$ cat myerror.txt  Digunakan untuk melihat isi file myerror.txt.



$ find . –name “\*.txt” –exec wc –l ‘{}’ ‘;’  Digunakan untuk mencari letak file sesuai dengan ektensi file.



$ which ls  Digunakan untuk mencari direktori mana yang terdapat ls.



$ locate “\*.txt”  Digunakan untuk mencari file pada semua direktori yang memiliki ektensi .txt

## Percobaan 6 : Mencari text pada file

$ grep Hallo \*.txt  Digunakan untuk melihat file .txt yang memiliki isi “Hallo”.

# LATIHAN:

1. Cobalah urutan perintah berikut :

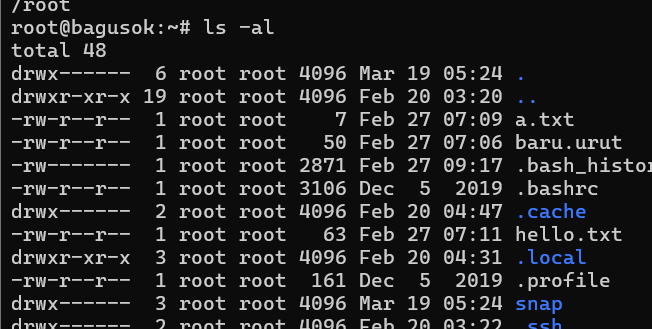
$ cd



$ pwd



$ ls –al



$ cd .



$ pwd



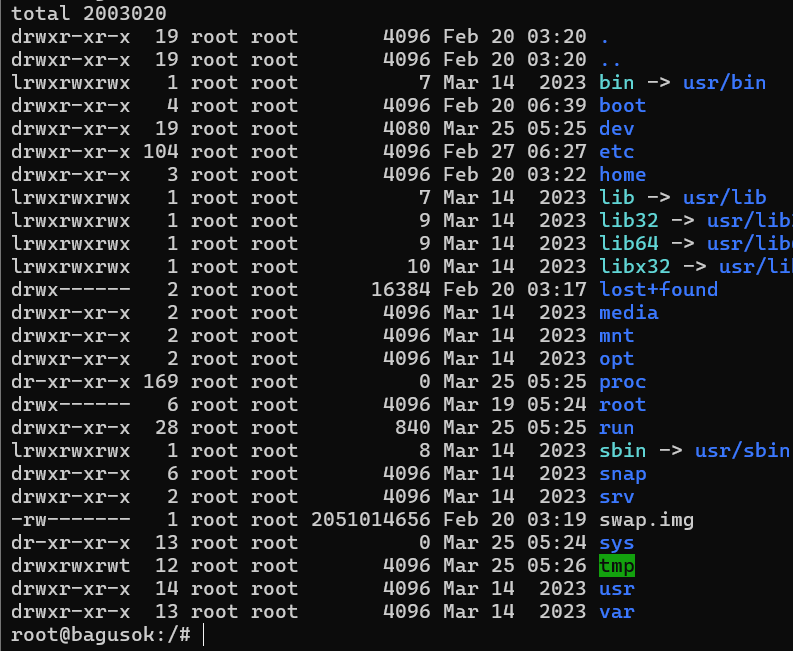
$ cd ..



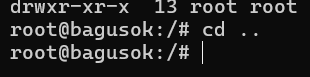
$ pwd



$ ls -al



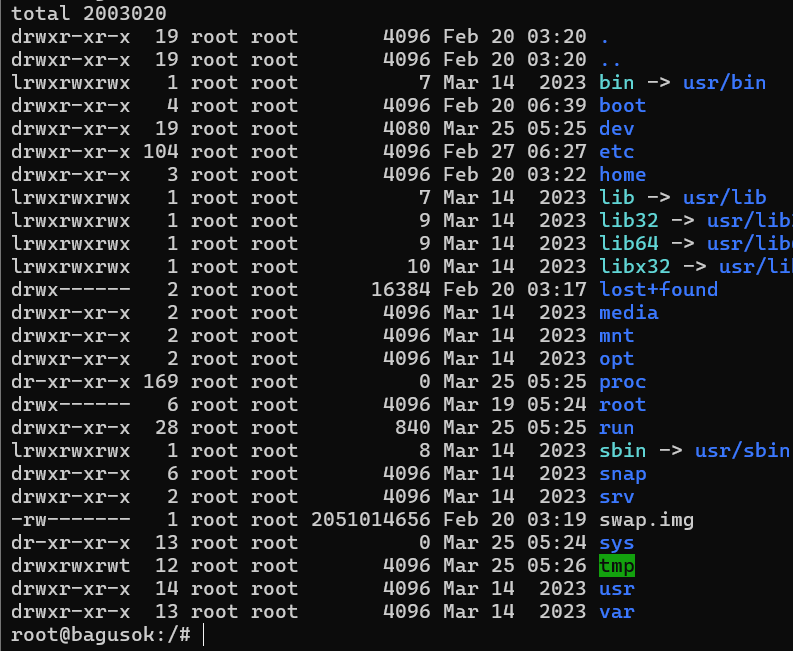
$ cd ..



$ pwd

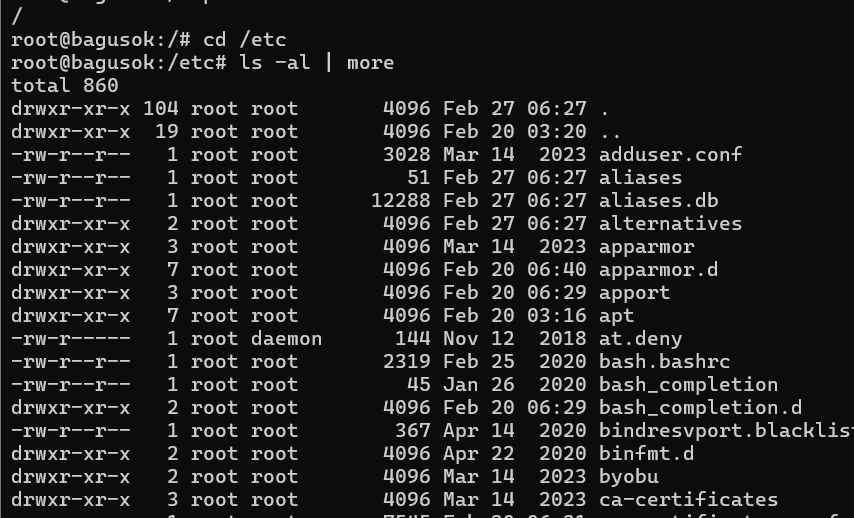


$ ls -al

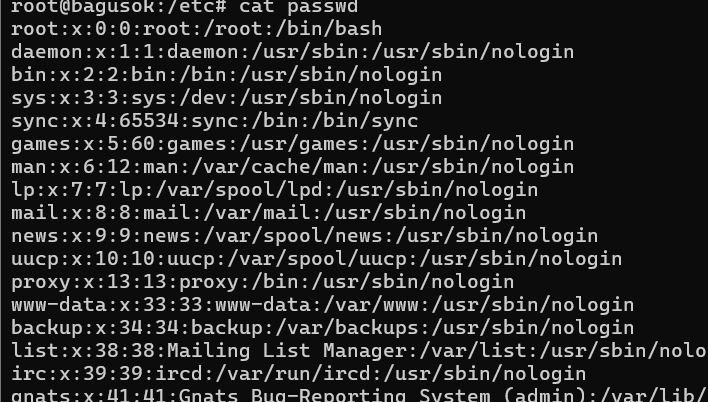


$ cd /etc

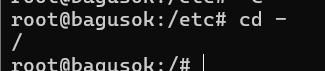
$ ls –al | more



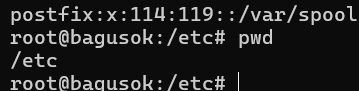
$ cat passwd



$ cd –



$ pwd

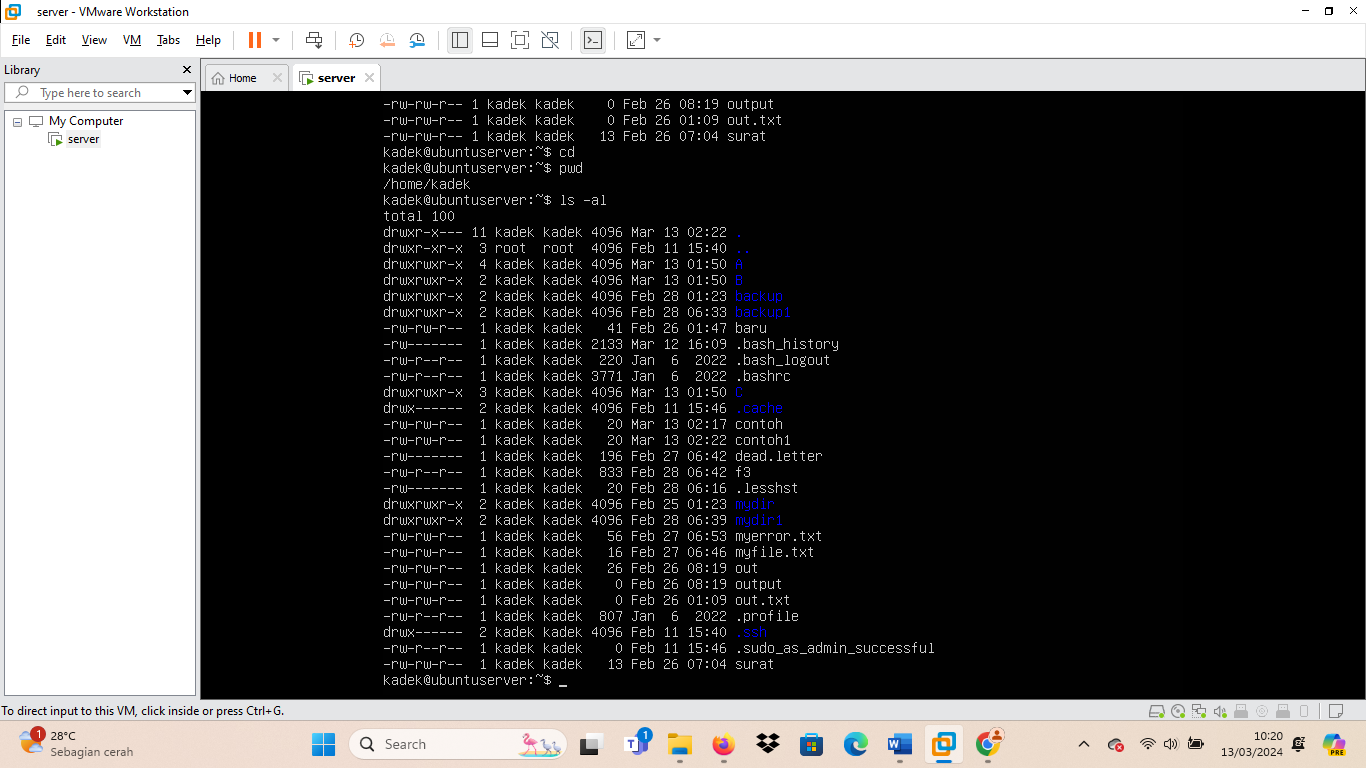


Analisa:

$ cd  Digunakan untuk berpindah ke direktori lain.

$ pwd  Digunakan untuk melihat posisi user.

$ ls –al  Digunakan untuk melihat semua file secara lengkap.



$ cd ..  Digunakan untuk kembali ke direktori sebelumnya.

$ cat  Digunakan untuk melihat isi dari sebuah file.

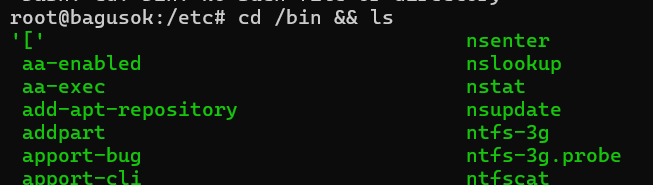
$ ls –al | more  Digunakan untuk melihat semua file secara satu layar penuh.

$ cd –  Digunakan untuk kembali ke root.

1. Lanjutkan penelusuran pohon pada sistem file menggunakan cd, ls, pwd dan cat.

Telusuri direktory /bin, /usr/bin, /sbin, /tmp dan /boot.

*/bin  Berisi utilitas sistem level rendah (binary).*/usr/bin  Berisi utilitas sistem dan program aplikasi level tinggi.

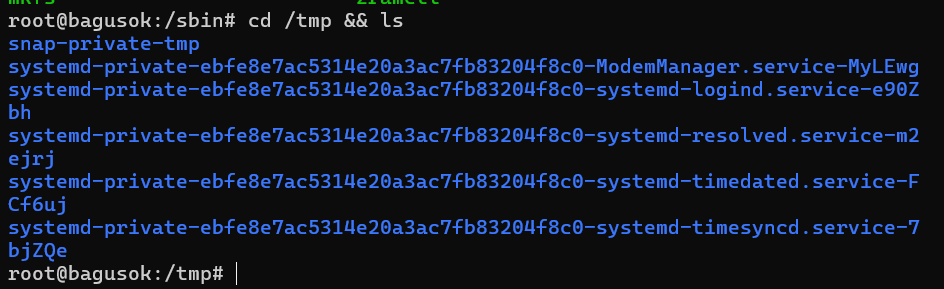


*/sbin  Berisi utilitas sistem untuk superuser (untuk

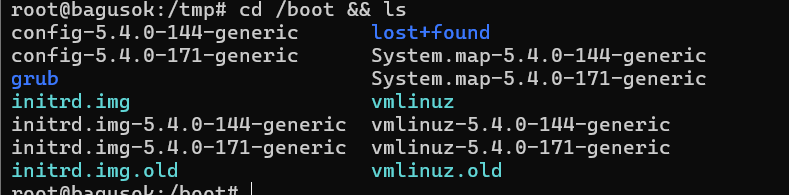
membentuk administrasi sistem).

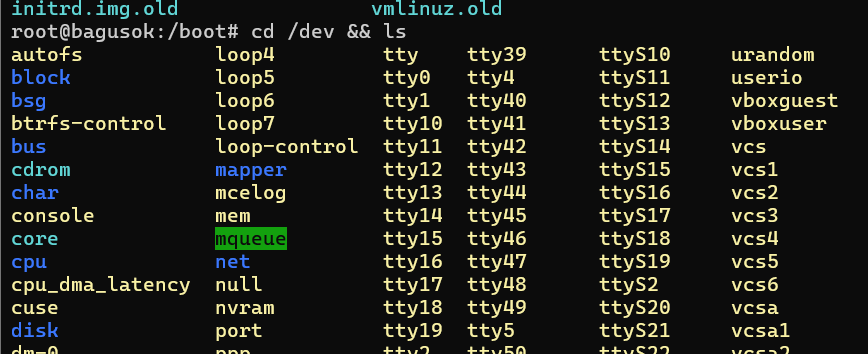


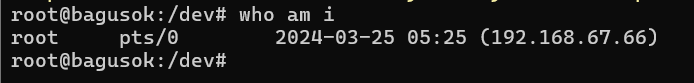
*/tmp  Berisi file sementara, yang pada saat Bootstrap akan dihapus (dapat digunakan oleh sembarang user).



*/boot  Berisi file yang sangat penting untuk proses bootstrap. Kernel vmlinuz disimpan di direktori ini.

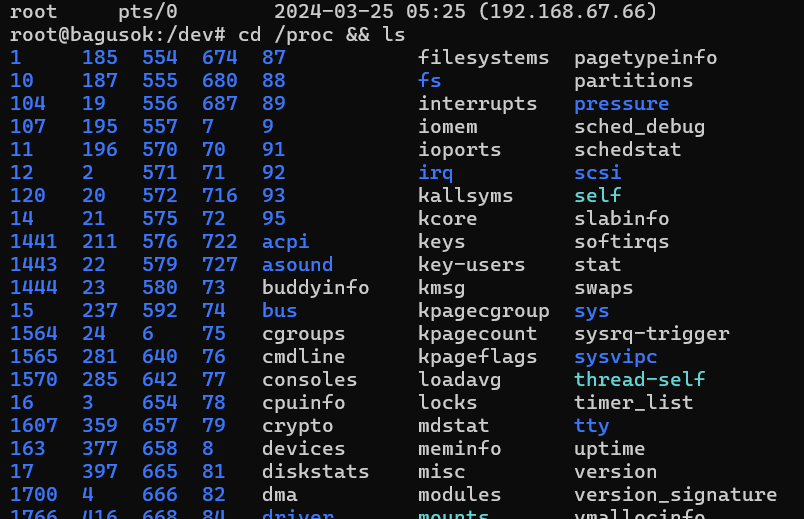


1. Telusuri direktory /dev. Identifikasi perangkat yang tersedia. Identifikasi tty (termninal) Anda (ketik who am i); siapa pemilih tty Anda (gunakan ls –l).
2. 



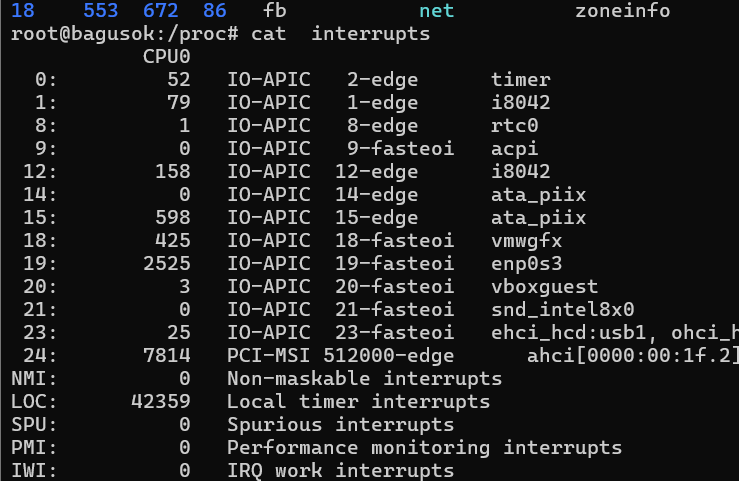
Setelah menelusuri direktori /dev dengan perintah cd, di dalam /dev ada daftar perangkat yang tersedia. Kemudian mengidentifikasi terminal yang digunakan menggunakan perintah whoami yang disana tertera user yaitu damelia. Selanjutnya menggunakan perintah ls –l untuk menampilkan/memunculkan semua isi direktori /dev.

1. Telusuri derectory /proc. Tampilkan isi file interrupts, devices, cpuinfo, meminfo dan uptime menggunakan perintah cat. Dapatkah Anda melihat mengapa directory /proc disebut *pseudo-filesystem* yang memungkinkan akses ke struktur data kernel ?

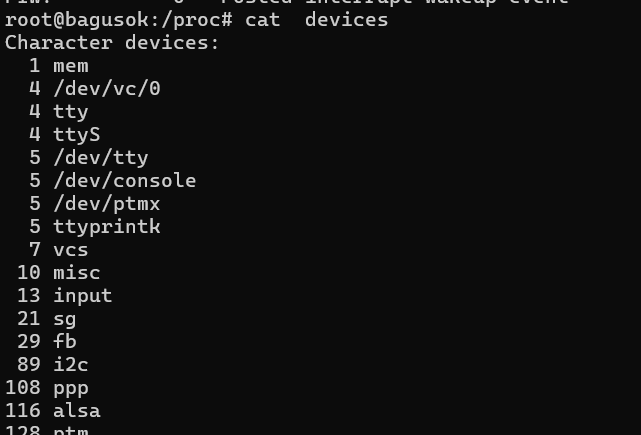


Direktori /proc disebut pseudo-filesystem dikarenakan direktori /proc dibuat diatas RAM dengan sistem file yang diatur oleh kernel itu sendiri, jadi user bisa melihat info memory, info processor, dsb.

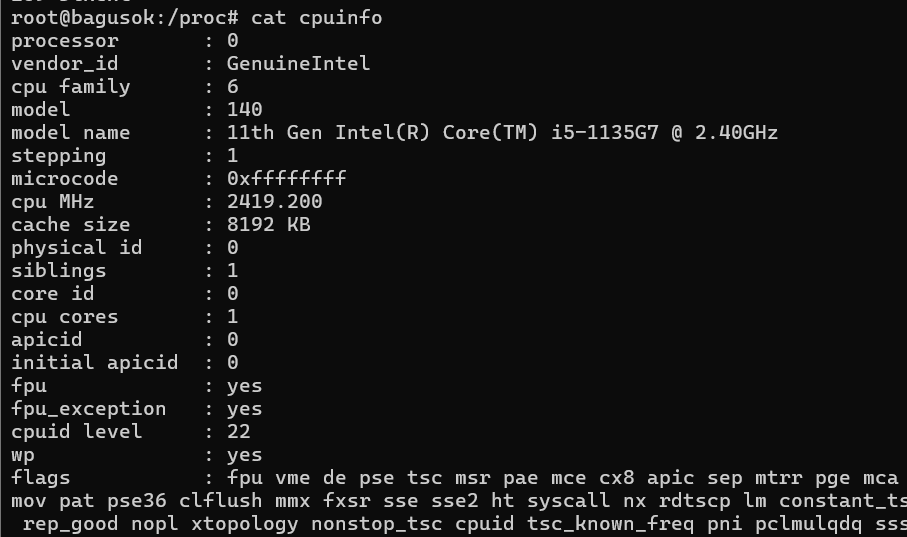
*interrupts  Berisi tentang proses apa saja yang terjadi saat ini.



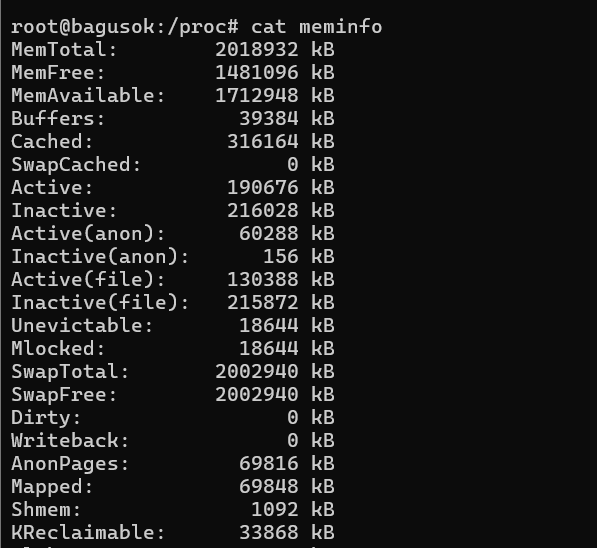
*devices  Berisi tentang devices apa saja yang diproses (aktif) pada saat ini.



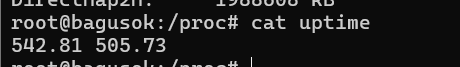
*cpuinfo  Berisi tentang info cpu dan bagaimana prosesnya saat ini.



*meminfo  Berisi tentang info memory dan bagaimana prosesnya saat ini.



*uptime  Mengetahui informasi tentang lama system berjalan setelah terakhir reboot atau mati.



1. Ubahlah direktory home ke user lain secara langsung menggunakan cd ~username.



1. Ubah kembali ke direktory home Anda.



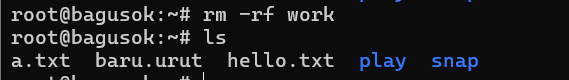
1. Buat subdirektory work dan play.

Dalam membuat subdirektory /work dan /play, saya membuat terlebih dahulu direktori

/damelia yang nantinya diisi subdirektory /work dan /play.



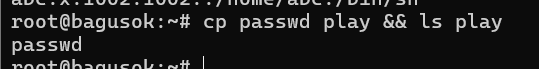
1. Hapus subdirektory work.



1. Copy file /etc/passwd ke direktory home Anda.



1. Pindahkan ke subirectory play.



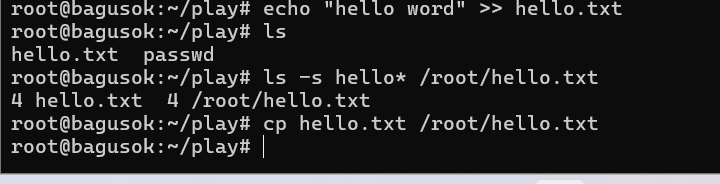
1. Ubahlah ke subdirektory play dan buat symbolic link dengan nama terminal yang menunjuk ke perangkat tty. Apa yang terjadi jika melakukan hard link ke perangkat tty ?



Ketika membuat symbolic link dengan hardlink terjadi error karena tty adalah file system, namun jika menggunakan softlink dengan perintah ln -s berhasil.

1. Buatlah file bernama hello.txt yang berisi kata”hello word”. Dapatkah Anda gunakan

”cp” menggunakan ”terminal” sebagai file asal untuk menghasilkan efek yang sama ?

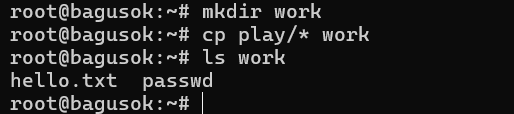


1. Copy hello.txt ke terminal. Apa yang terjadi?

Yang terjadi adalah error dikarenakan terlalu banyak level dari symbolic link.

1. Masih direktory home, copy keseluruhan direktory play ke direktory bernama work

menggunakan symbolic link.



1. Hapus direktory work dan isinya dengan satu perintah

