

## NAVIGASI FILESYSTEM LINUX

### OBJEKTIF :

1. Mahasiswa Mampu Memahami Directory Structure dan Paths.
  2. Mahasiswa Mampu Menggunakan Listing Files in Directory.
- 

### PENDAHULUAN

Di Linux, semuanya dianggap sebagai file. File digunakan untuk menyimpan data seperti teks, grafik, dan program. Direktori adalah jenis file yang digunakan untuk menyimpan file lain; Pengguna Windows dan Mac OS X biasanya menyebutnya sebagai folder. Bagaimanapun, direktori digunakan untuk menyediakan struktur organisasi hierarki. Namun, struktur ini mungkin agak berbeda tergantung pada jenis sistem yang digunakan.

Saat bekerja di sistem operasi Linux, penting untuk mengetahui cara memanipulasi file dan direktori. Beberapa distribusi Linux memiliki aplikasi berbasis GUI yang memungkinkan Anda untuk mengelola file, tetapi bermanfaat untuk mengetahui cara melakukan operasi ini melalui *command line*.

## Did you know these companies use Linux?

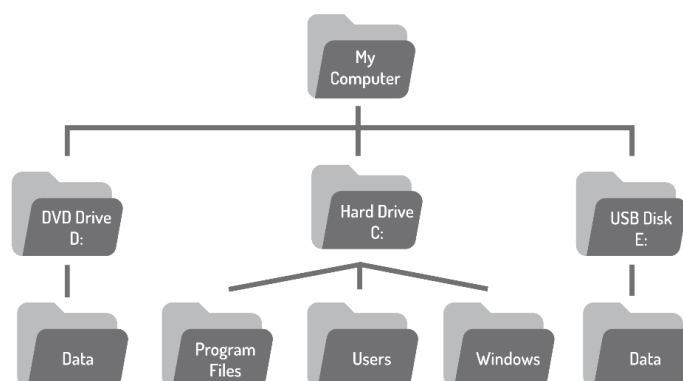


### 5.1 STRUKTUR DIREKTORI

Pada sistem Windows, tingkat teratas dari struktur direktori disebut My Computer. Perangkat fisik, seperti hard drive, drive USB, *network drive*, muncul di bawah *My Computer* dan masing-masing diberi *drive letter*, seperti C: atau D:.

Struktur direktori yang ditunjukkan di bawah ini disediakan hanya sebagai contoh. Direktori ini mungkin tidak ada dalam lingkungan mesin virtual kursus ini.

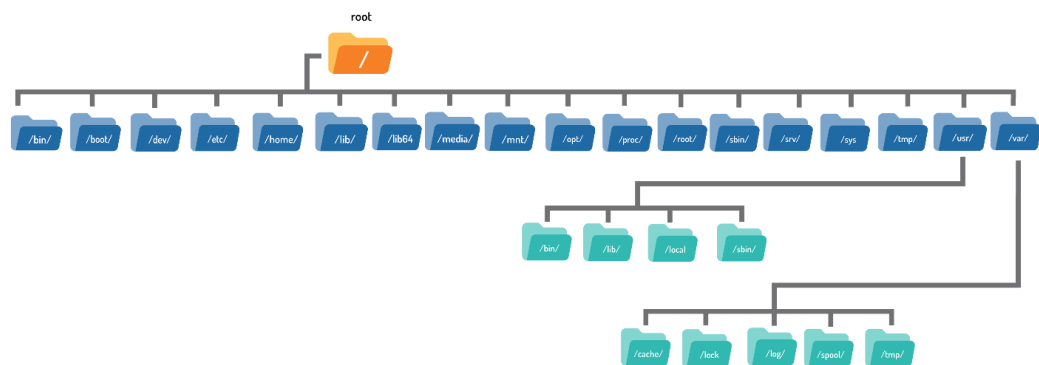
Representasi visual dari struktur direktori Windows:



Copyright © 2018 Network Development Group Inc.

Seperti Windows, struktur direktori Linux, biasanya disebut *filesystem*, juga memiliki *top level*. Namun alih-alih *My Computer*, ini disebut direktori root, dan dilambangkan dengan karakter *slash* /. Selain itu, tidak ada drive di Linux; setiap perangkat fisik dapat diakses di bawah direktori, bukan *drive letter*.

Gambar berikut menunjukkan representasi visual dari *filesystem* Linux yang khas:



Copyright © 2019 Network Development Group Inc.

Untuk melihat isi direktori root, gunakan command `ls` dengan karakter `'/'` sebagai argumen:

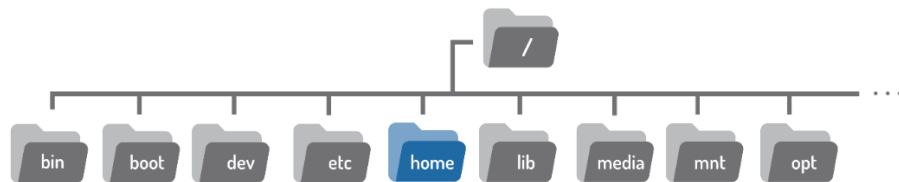
```

sysadmin@localhost:~$ ls /
bin    etc    lib    mnt    root  'sbin'$'\342\200\214'  tmp
boot   home   lib64  opt    run    srv                      usr
dev    init   media  proc   sbin   sys                      var
  
```

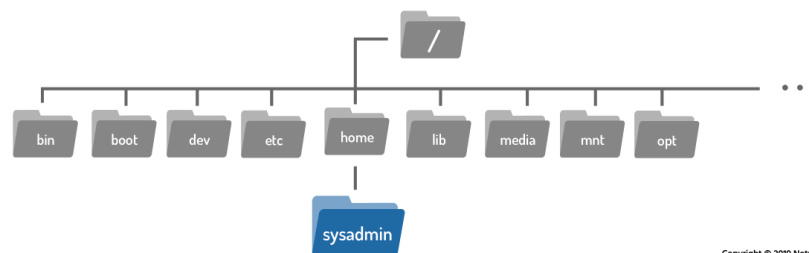
Perhatikan bahwa ada banyak direktori dengan nama deskriptif termasuk `/boot`, yang berisi file untuk *boot* komputer.

### 5.1.1 HOME DIRECTORY

Istilah *home directory* sering membingungkan pengguna baru Linux. Pertama-tama, pada kebanyakan distribusi Linux ada direktori bernama `home` di bawah direktori root `/`.

Copyright © 2019 Network Development Group Inc.

Di bawah direktori `/home` ini terdapat direktori untuk setiap pengguna di sistem. Nama direktori sama dengan nama pengguna, jadi pengguna bernama `sysadmin` akan memiliki direktori *home* bernama `/home/sysadmin`.

Copyright © 2019 Network Development Group Inc.

Direktori *home* adalah direktori penting. Pertama-tama, ketika pengguna membuka shell, mereka secara otomatis ditempatkan di direktori *home* mereka, karena biasanya di sinilah mereka melakukan sebagian besar pekerjaan mereka.

Selain itu, direktori *home* adalah salah satu dari beberapa direktori di mana pengguna memiliki kontrol penuh untuk membuat dan menghapus file dan direktori tambahan. Di sebagian besar distribusi Linux, satu-satunya pengguna yang dapat mengakses file di direktori *home* adalah pemilik dan administrator sistem. Sebagian besar direktori lain dalam *filesystem* Linux dilindungi dengan *file permissions*.

Direktori *home* memiliki simbol khusus yang digunakan untuk mewakilinya; karakter tilde `~`. Jadi jika pengguna `sysadmin login`, karakter tilde `~` dapat digunakan di direktori `/home/sysadmin`.

Dimungkinkan juga untuk merujuk ke direktori *home* pengguna lain dengan menggunakan karakter tilde *~* diikuti dengan nama akun pengguna. Misalnya, *~bob* sama dengan */home/bob*.

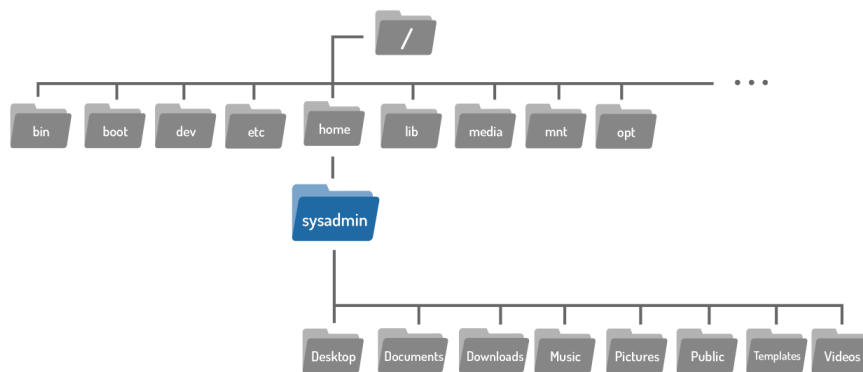
### 5.1.2 CURRENT DIRECTORY

Untuk menentukan di mana pengguna saat ini berada dalam sistem file, *command* **pwd** (*print working directory*) dapat digunakan:

```
pwd [OPTIONS]
```

```
sysadmin@localhost:~$ pwd
/home/sysadmin
```

*Command* **pwd** mencetak direktori kerja, yang merupakan lokasi pengguna saat ini dalam *filesystem*. Output dari *command* di atas menunjukkan bahwa pengguna **sysadmin** saat ini berada di direktori *home* mereka, ditunjukkan pada *filesystem* di bawah ini:

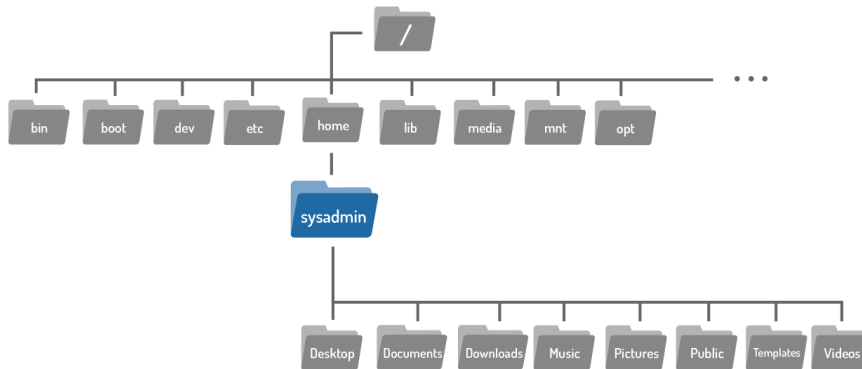


Copyright © 2019 Network Development Group Inc.

### 5.1.3 MENGUBAH DIREKTORI

Saat pengguna membuka *shell*, biasanya mereka mulai di direktori *home*-nya. Saat Anda memulai mesin virtual baru dalam kursus kami, baik dengan

membuka kursus atau setelah menggunakan tombol reset, Anda *login* sebagai pengguna *sysadmin*, dan Anda mulai di direktori *home* untuk pengguna tersebut, disorot pada gambar di bawah.



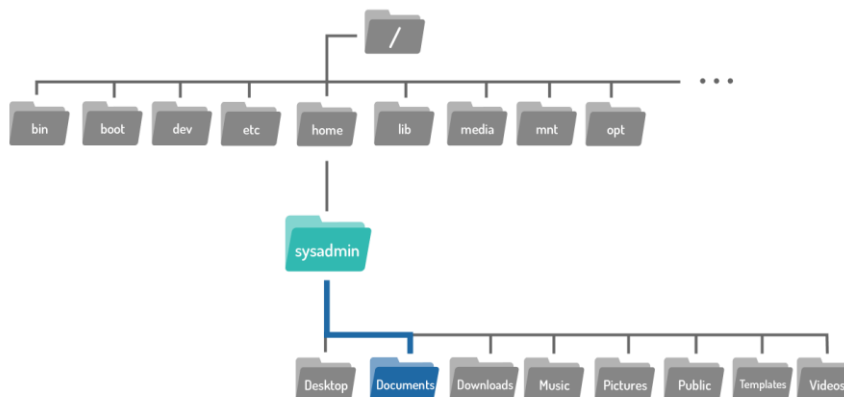
Copyright © 2019 Network Development Group Inc.

Untuk menavigasi sistem file, gunakan *command* **cd** (*change directory*).

```
cd [options] [path]
```

Ada direktori bernama *Documents* yang terletak di direktori *home* dari pengguna *sysadmin*. Untuk berpindah dari direktori *home* ke direktori *Documents*, gunakan nama direktori sebagai argumen ke *command* **cd**:

```
sysadmin@localhost:~$ cd Documents
sysadmin@localhost:~/Documents$
```



Copyright © 2019 Network Development Group Inc.

Jika digunakan tanpa argumen, *command* `cd` akan membawa pengguna ke direktori *home* mereka.

```
sysadmin@localhost:~/Documents$ cd
sysadmin@localhost:~$
```

Perhatikan mesin virtual kami menggunakan *prompt* yang menampilkan direktori kerja saat ini, ditekankan dengan warna biru. Pada prompt pertama, karakter tilde `~` sama dengan `/home/sysadmin`, mewakili direktori home pengguna.

```
sysadmin@localhost:~$
```

Setelah mengubah direktori, lokasi baru `~/Documents` juga dapat dikonfirmasi di *prompt* baru, lagi-lagi ditampilkan dengan warna biru.

```
sysadmin@localhost:~$ cd Documents
sysadmin@localhost:~/Documents$
```

#### Pertimbangkan ini

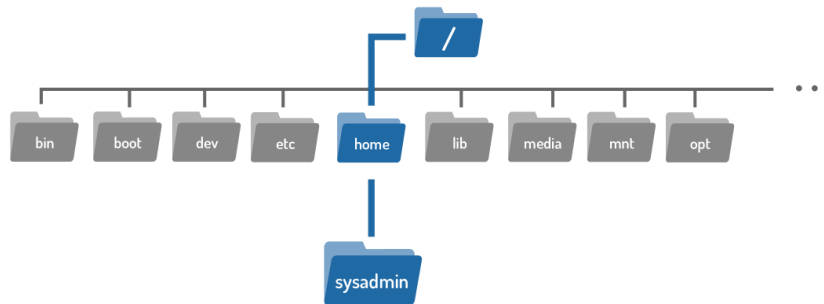
Untuk beberapa *command*, tidak ada berita yang berarti kabar baik; tidak ada keluaran jika *command* `cd` berhasil. Jika pengguna mencoba mengubah ke direktori yang tidak ada, *command* mengembalikan pesan kesalahan:

```
sysadmin@localhost:~$ cd Junk
-bash: cd: Junk: No such file or directory
```

## 5.2 PATHS

Argumen untuk *command* `cd` lebih dari sekedar nama direktori, itu sebenarnya jalan. Path adalah daftar direktori yang dipisahkan oleh karakter `'/'`. Jika Anda menganggap sistem file sebagai peta, jalurnya adalah alamat direktori, yang mencakup arah navigasi langkah demi langkah; mereka dapat digunakan untuk menunjukkan lokasi file apa pun dalam sistem file.

Sebagai contoh, `/home/sysadmin` adalah *path* ke direktori *home*:



Copyright © 2019 Network Development Group Inc.

Ada dua jenis jalur: *absolut* dan *relatif*.

### 5.2.1 ABSOLUTE PATHS

Path absolut memungkinkan pengguna untuk menentukan lokasi direktori yang tepat. Itu selalu dimulai pada direktori root, dan karena itu selalu dimulai dengan karakter '/'. *Path* `/home/sysadmin` adalah *path* absolut; ia memberitahu sistem untuk memulai pada direktori root `/`, pindah ke direktori *home*, dan kemudian ke direktori *sysadmin*.

Jika *path* `/home/sysadmin` digunakan sebagai argumen ke *command* `cd`, ia akan memindahkan pengguna ke direktori *home* untuk pengguna *sysadmin*.

```
sysadmin@localhost:~/Documents$ cd /home/sysadmin
```

Sekali lagi, tidak ada output berarti *command* berhasil. Ini bisa dikonfirmasi dengan melihat prompt, atau menggunakan *command* `pwd`:

```
sysadmin@localhost:~$ pwd
/home/sysadmin
```

### 5.2.2 RELATIVE PATHS

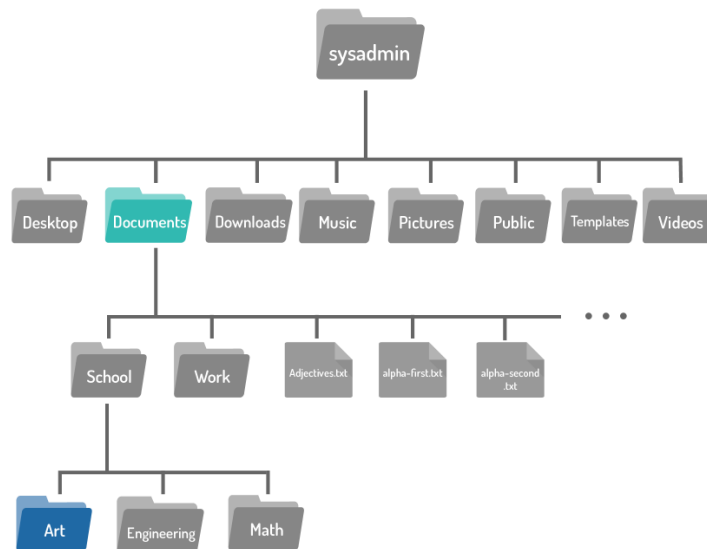
Path relatif dimulai dari direktori saat ini. Path relatif memberikan arah ke file relatif ke lokasi saat ini di sistem file. Mereka tidak dimulai dengan karakter '/'. Sebagai gantinya, mereka mulai dengan nama direktori. Lebih khusus, jalur relatif dimulai dengan nama direktori yang terdapat dalam direktori saat ini.



Lihatlah contoh *command* `cd` pertama. Argumen adalah contoh dari jalur relatif paling sederhana: nama direktori dalam direktori kerja saat ini.

```
sysadmin@localhost:~$ cd Documents
sysadmin@localhost:~/Documents$
```

Jika pengguna berada di direktori *Documents*, pindah ke direktori *Art* dapat dilakukan dengan beberapa cara.



Copyright © 2019 Network Development Group Inc.

Path absolut ke direktori *Art* dapat digunakan:

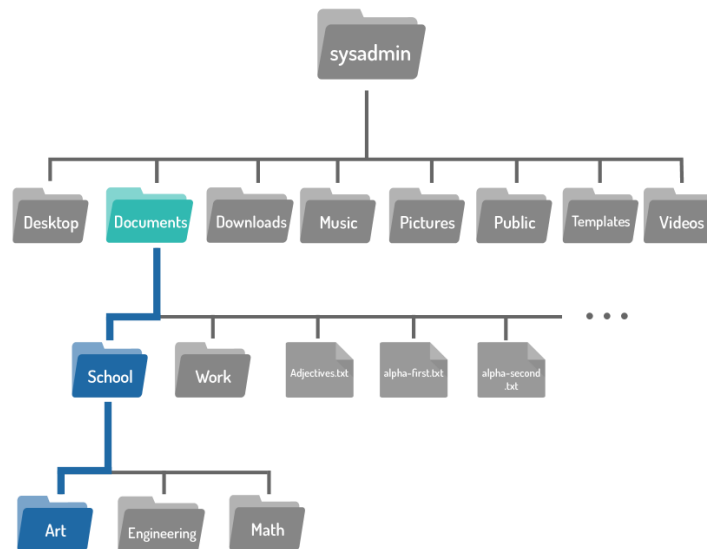
```
sysadmin@localhost:~/Documents$ cd /home/sysadmin/Documents/School/Art
sysadmin@localhost:~/Documents/School/Art$
```

Beberapa jalur relatif dapat digunakan:

```
sysadmin@localhost:~/Documents$ cd School
sysadmin@localhost:~/Documents/School$ cd Art
sysadmin@localhost:~/Documents/School/Art$
```

Namun, metode paling sederhana adalah dengan menggunakan jalur relatif tunggal yang mencakup perjalanan dari asal ke direktori tujuan:

```
sysadmin@localhost:~/Documents$ cd School/Art
sysadmin@localhost:~/Documents/School/Art$
```



Copyright © 2019 Network Development Group Inc.

Gunakan *command* `pwd` untuk mengonfirmasi perubahan:

```
sysadmin@localhost:~/Documents/School/Art$ pwd
/home/sysadmin/Documents/School/Art
```

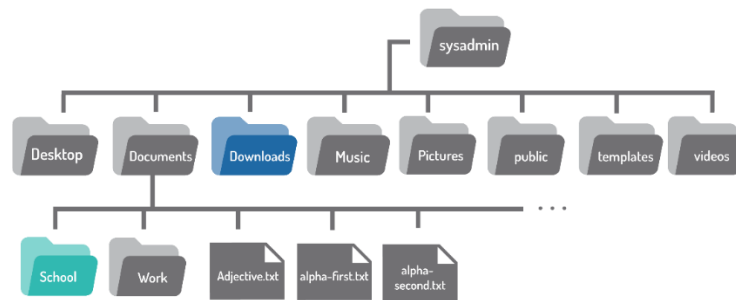
### 5.2.3 SHORTCUTS

#### Karakter “..”

Terlepas dari direktori mana pengguna berada, dua periode ‘..’ karakter selalu mewakili satu direktori lebih tinggi relatif terhadap direktori saat ini, kadang-kadang disebut sebagai direktori induk. Untuk pindah dari direktori *Art* kembali ke direktori *School*:

```
sysadmin@localhost:~/Documents/School/Art$ cd ..
sysadmin@localhost:~/Documents/School$
```

Titik ganda juga dapat digunakan di jalur yang lebih panjang juga. Jalur relatif berikut dapat digunakan untuk berpindah dari direktori *School* ke direktori *Downloads* (keduanya disorot pada gambar di bawah):



Copyright © 2015 Network Development Group Inc.

```
sysadmin@localhost:~/Documents/School$ cd ../../Downloads
sysadmin@localhost:~/Downloads$
```

### Karakter “.”

Terlepas dari direktori mana pengguna berada, satu periode ‘.’ karakter selalu mewakili direktori saat ini. Untuk `cd` pintasan ini tidak terlalu berguna, tetapi berguna untuk *command* yang tercakup dalam bagian selanjutnya.

## 5.3 MENAMPILKAN DAFTAR FILE DI DIREKTORI

Untuk contoh sebelumnya, gambar disediakan untuk menunjukkan tata letak *filesystem*. Dalam praktiknya, peta seperti ini tidak disediakan, dan pengguna harus mengandalkan apa yang tersedia di *command line*, membuat command `ls` (*list*) salah satu yang paling *powerful* untuk menavigasi *filesystem*.

```
ls [OPTION]... [FILE]...
```

Command `ls` digunakan untuk menampilkan isi direktori dan dapat memberikan informasi rinci tentang file. Secara default, ketika digunakan tanpa opsi atau argumen, itu membuat daftar file dalam direktori saat ini:

```
sysadmin@localhost:~$ ls
Desktop Documents Downloads Music Pictures Public Templates Videos
```

Command `ls` juga dapat digunakan untuk membuat daftar isi dari direktori manapun dalam sistem berkas. Berikan jalur ke direktori sebagai argumen:

```
sysadmin@localhost:~$ ls /var
backups  cache  lib  local  lock  log  mail  opt  run  spool  tmp
```

### Pertimbangkan ini

Di banyak distribusi Linux, termasuk yang digunakan di mesin virtual kami, command `ls` menggunakan warna untuk membedakan dengan jenis file. Misalnya, direktori dapat ditampilkan dengan warna biru, file yang dapat dijalankan dapat ditampilkan dalam warna hijau, dan tautan simbolik dapat ditampilkan dalam warna cyan.

Keluaran berwarna bukanlah perilaku default untuk command `ls`, melainkan efek dari option `--color`. `ls` tampaknya melakukan pewarnaan ini secara otomatis karena ada alias untuk command `ls`, sehingga berjalan dengan option `-color`.

```
sysadmin@localhost:~$ type ls
ls is aliased to `ls --color=auto`
```

Untuk menghindari penggunaan alias, letakkan karakter garis miring terbalik `\`` di depan command Anda:

```
sysadmin@localhost:~$ ls
Desktop  Documents  Downloads  Music  Pictures  Public  Templates  Videos
sysadmin@localhost:~$ \ls
Desktop  Documents  Downloads  Music  Pictures  Public  Templates  Videos
```

### 5.3.1 DAFTAR FILE TERSEMBUNYI

Command `ls` digunakan untuk menampilkan isi direktori, tidak semua file ditampilkan secara otomatis. Command `ls` menghilangkan *hidden files*

secara default. *Hidden file* adalah file (atau direktori) apa pun yang dimulai dengan karakter *dot* `'.'`.

Untuk menampilkan semua file, termasuk *hidden files*, gunakan option `-a` ke command `ls`:

```
sysadmin@localhost:~$ ls -a
.          .bashrc    .selected_editor  Downloads  Public
..         .cache     Desktop           Music      Templates
.bash_logout .profile  Documents         Pictures   Videos
```

Mengapa file disembunyikan sejak awal? Sebagian besar *hidden files* adalah *customization files*, yang dirancang untuk menyesuaikan cara kerja Linux, shell atau program Anda. Misalnya, file `.bashrc` di direktori home menyesuaikan fitur *shell*, seperti membuat atau mengubah variabel dan alias.

*Customization files* ini bukan yang Anda kerjakan secara rutin, dan menampilkannya akan mempersulit pencarian file lain.

### 5.3.2 LONG DISPLAY LISTING

Setiap file memiliki detail yang terkait dengannya yang disebut metadata. Ini dapat mencakup informasi seperti *size*, *ownership*, atau *timestamps*. Untuk melihat informasi ini, gunakan option `-l` ke command `ls`. Di bawah ini, daftar direktori `/var/log` digunakan sebagai contoh, karena menyediakan berbagai keluaran:

```
sysadmin@localhost:~$ ls -l /var/log/
total 900
-rw-r--r-- 1 root  root  15322 Dec 10 21:33 alternatives.log
drwxr-xr-x 1 root  root   4096 Jul 19 06:52 apt
-rw-r----- 1 syslog adm    371 Dec 15 16:38 auth.log
-rw-r--r-- 1 root  root  35330 May 26  2018 bootstrap.log
-rw-rw---- 1 root  utmp     0 May 26  2018 btmp
-rw-r----- 1 syslog adm    197 Dec 15 16:38 cron.log
-rw-r--r-- 1 root  adm   85083 Dec 10 21:33 dmesg
-rw-r--r-- 1 root  root  351960 Jul 19 06:52 dpkg.log
-rw-r--r-- 1 root  root   32064 Dec 10 21:33 faillog
drwxr-xr-x 2 root  root   4096 Jul 19 06:51 journal
-rw-rw-r-- 1 root  utmp 292584 Dec 15 16:38 lastlog
-rw-r----- 1 syslog adm   14185 Dec 15 16:38 syslog
-rw----- 1 root  root   64128 Dec 10 21:33 tallylog
-rw-rw-r-- 1 root  utmp    384 Dec 15 16:38 wtmp
```

Pada keluaran di atas, setiap baris menampilkan metadata tentang satu file. Berikut ini menjelaskan masing-masing bidang data dalam output dari command `ls -l`:

- **File Type**

```
-rw-r--r-- 1 root  root 15322 Dec 10 21:33 alternatives.log
drwxr-xr-x 1 root  root  4096 Jul 19 06:52 apt
```

Karakter pertama dari setiap baris menunjukkan jenis file. Jenis file tersebut adalah:

Simbol	Jenis File	Deskripsi
d	direktori	File yang digunakan untuk menyimpan file lain.
-	file biasa	Termasuk file yang dapat dibaca, file gambar, file biner, dan file terkompresi.
l	tautan simbolik/ <i>symbolic link</i>	Menunjuk ke file lain.
s	<i>socket</i>	Memungkinkan komunikasi antar proses.
p	<i>pipe</i>	Memungkinkan komunikasi antar proses.
b	<i>block file</i>	Digunakan untuk berkomunikasi dengan <i>hardware</i> .

Simbol	Jenis File	Deskripsi
--------	------------	-----------

c	file karakter	Digunakan untuk berkomunikasi dengan <i>hardware</i> .
---	---------------	--

File pertama `alternatives.log` adalah file biasa (-), sedangkan file kedua `apt` adalah direktori (d).

- **Permissions**

```
d rwxr-xr-x 2 root root 4096 Jul 19 06:51 journal
```

Sembilan karakter berikutnya menunjukkan *permissions* file. *Permissions* menunjukkan bagaimana pengguna tertentu dapat mengakses file.

- **Hard Link Count**

```
-rw-r----- 1 syslog adm 371 Dec 15 16:38 auth.log
drwxr-xr-x 2 root root 4096 Jul 19 06:51 journal
```

Angka ini menunjukkan berapa banyak *hard link* yang mengarah ke file ini.

- **User Owner**

```
-rw-r----- 1 syslog adm 197 Dec 15 16:38 cron.log
```

Setiap file dimiliki oleh akun pengguna. Ini penting karena pemilik memiliki hak untuk mengatur *permissions* pada file.

- **Group Owner**

```
-rw-rw-r-- 1 root utmp 292584 Dec 15 16:38 lastlog
```

Menunjukkan grup mana yang memiliki file ini. Ini penting karena setiap anggota grup ini memiliki sekumpulan *permissions* pada file tersebut.

- **File Size**

```
-rw-r----- 1 syslog adm 14185 Dec 15 16:38 syslog
```

Menampilkan ukuran file dalam byte.

Untuk direktori, nilai ini tidak menggambarkan ukuran total direktori, melainkan berapa banyak *byte* yang dicadangkan untuk melacak nama file di direktori. Dengan kata lain, abaikan bidang ini untuk direktori.

- **Timestamp**

```
-rw-rw---- 1 root utmp 0 May 26 2018 btmp
```

Menunjukkan waktu terakhir kali konten file diubah. Untuk direktori, stempel waktu ini menunjukkan terakhir kali file ditambahkan atau dihapus dari direktori.

- **File Name**

```
-rw-r--r-- 1 root root 35330 May 26 2018 bootstrap.log
```

Bidang terakhir berisi nama file atau direktori.

Dalam kasus *symbolic links*, nama link ditampilkan bersama dengan panah dan nama jalur dari file asli.

```
lrwxrwxrwx. 1 root root 22 Nov 6 2018 /etc/grub.conf -> ../boot/grub/grub.conf
```

### 5.3.3 UKURAN YANG DAPAT DIBACA

Option `-l` ke command `ls` menampilkan ukuran file dalam byte. Untuk file teks, satu *byte* adalah 1 karakter. Untuk file yang lebih kecil, ukuran *byte* sudah cukup. Namun, untuk file yang lebih besar, sulit untuk memahami seberapa besar file tersebut. Sebagai contoh, perhatikan *output* dari *command* berikut:

```
sysadmin@localhost:~$ ls -l /var/log/lastlog
-rw-rw-r-- 1 root utmp 292584 Dec 15 16:38 /var/log/lastlog
```



Ukuran file sulit ditentukan dalam byte. Apakah 292584 file besar atau kecil? Tampaknya cukup besar, tetapi sulit untuk ditentukan menggunakan byte.

Anggap saja seperti ini: jika seseorang memberikan jarak antara Boston dan New York menggunakan inci, nilai itu tidak ada artinya. Kebanyakan orang berpikir dalam mil atau kilometer.

Terkadang lebih baik menampilkan ukuran file dalam ukuran yang lebih dapat dibaca manusia, seperti *megabyte* atau *gigabyte*. Untuk mencapai ini, tambahkan option `-h` ke *command* `ls`:

```
sysadmin@localhost:~$ ls -lh /var/log/lastlog
-rw-rw-r-- 1 root utmp 286K Dec 15 16:38 /var/log/lastlog
```

**Penting:** option `-h` harus digunakan dengan option `-l`.

#### 5.3.4 DAFTAR DIREKTORI

Ketika option `-d` digunakan, ini merujuk ke direktori saat ini, dan bukan konten di dalamnya. Tanpa option lain, ini agak tidak berarti. Ingat bahwa direktori saat ini selalu disebut dengan karakter satu period/titik “.”:

```
sysadmin@localhost:~$ ls -d
.
```

Untuk menggunakan option `-d` dengan cara yang bermakna membutuhkan penambahan option `-l`. Dalam hal ini, perhatikan bahwa command berikut mencantumkan rincian konten di direktori `/home/sysadmin`:

```
sysadmin@localhost:~$ ls -l
total 32
drwxr-xr-x 2 sysadmin sysadmin 4096 Dec 10 21:33 Desktop
drwxr-xr-x 4 sysadmin sysadmin 4096 Dec 10 21:33 Documents
drwxr-xr-x 2 sysadmin sysadmin 4096 Dec 10 21:33 Downloads
drwxr-xr-x 2 sysadmin sysadmin 4096 Dec 10 21:33 Music
drwxr-xr-x 2 sysadmin sysadmin 4096 Dec 10 21:33 Pictures
drwxr-xr-x 2 sysadmin sysadmin 4096 Dec 10 21:33 Public
drwxr-xr-x 2 sysadmin sysadmin 4096 Dec 10 21:33 Templates
drwxr-xr-x 2 sysadmin sysadmin 4096 Dec 10 21:33 Videos
```

Sebagai perbandingan, *command* berikutnya mencantumkan direktori /home/sysadmin itu sendiri:

```
sysadmin@localhost:~$ ls -ld
drwxr-xr-x 1 sysadmin sysadmin 224 Nov  7 17:07 .
```

Catat period tunggal di akhir daftar panjang. Ini menunjukkan bahwa direktori saat ini sedang terdaftar, dan bukan isinya.

### 5.3.5 DAFTAR RECURSIVE

Ada kalanya Anda ingin menampilkan semua file dalam direktori serta semua file di semua subdirektori di bawah direktori itu. Ini disebut daftar rekursif.

Untuk melakukan daftar berulang, gunakan option **-R** ke command **ls**:

```
sysadmin@localhost:~$ ls -R /etc/ppp
/etc/ppp:
ip-down.d ip-up.d

/etc/ppp/ip-down.d:
bind9

/etc/ppp/ip-up.d:
bind9
```

Perhatikan bahwa pada contoh sebelumnya, file-file di direktori /etc/ppp terdaftar terlebih dahulu. Setelah itu, isi subdirektori /etc/ppp/ip-down.d dan /etc/ppp/ip-up.d terdaftar.

Hati-hati dengan option ini; misalnya, menjalankan command pada direktori root akan mencantumkan setiap file pada sistem file, termasuk semua file pada perangkat USB dan DVD apa pun yang terpasang dalam sistem. Batasi penggunaan daftar rekursif untuk struktur direktori yang lebih kecil.

### 5.3.6 MENGURUTKAN DAFTAR FILE

Secara default, *command* `ls` mengurutkan file secara alfabet berdasarkan nama file. Terkadang, mungkin berguna untuk mengurutkan file menggunakan kriteria yang berbeda.

Untuk mengurutkan file berdasarkan ukuran, kita dapat menggunakan option `-S`. Perhatikan perbedaan dalam *output* dari dua *command* berikut. File dan direktori yang sama terdaftar, tetapi dalam urutan yang berbeda:

```
sysadmin@localhost:~$ ls /etc/ssh
moduli          ssh_host_ecdsa_key.pub  ssh_host_rsa_key  sshd_config
ssh_config      ssh_host_ed25519_key   ssh_host_rsa_key.pub
ssh_host_ecdsa_key  ssh_host_ed25519_key.pub  ssh_import_id
```

```
sysadmin@localhost:~$ ls -S /etc/ssh
moduli          ssh_host_ed25519_key  ssh_host_ecdsa_key.pub
sshd_config     ssh_host_rsa_key.pub  ssh_host_ed25519_key.pub
ssh_host_rsa_key  ssh_import_id
ssh_config      ssh_host_ecdsa_key
```

#### Catatan

Option di atas menggunakan huruf kapital s.

Sementara option `-S` bekerja dengan sendirinya, ini sangat berguna ketika digunakan dengan option `-l` sehingga ukuran file terlihat. *Command* berikut mencantumkan file dari terbesar ke terkecil dan menampilkan ukuran sebenarnya dari file tersebut.

```
sysadmin@localhost:~$ ls -lS /etc/ssh
total 580
-rw-r--r-- 1 root root 553122 Feb 10 2018 moduli
-rw-r--r-- 1 root root 3264 Feb 10 2018 sshd_config
-rw----- 1 root root 1679 Jul 19 06:52 ssh_host_rsa_key
-rw-r--r-- 1 root root 1580 Feb 10 2018 ssh_config
-rw----- 1 root root 411 Jul 19 06:52 ssh_host_ed25519_key
-rw-r--r-- 1 root root 399 Jul 19 06:52 ssh_host_rsa_key.pub
-rw-r--r-- 1 root root 338 Jul 19 06:52 ssh_import_id
-rw----- 1 root root 227 Jul 19 06:52 ssh_host_ecdsa_key
-rw-r--r-- 1 root root 179 Jul 19 06:52 ssh_host_ecdsa_key.pub
-rw-r--r-- 1 root root 99 Jul 19 06:52 ssh_host_ed25519_key.pub
```

Mungkin juga berguna untuk menggunakan option `-h` untuk menampilkan ukuran file yang dapat dibaca manusia:

```

sysadmin@localhost:~$ ls -lSh /etc/ssh
total 580K
-rw-r--r-- 1 root root 541K Feb 10 2018 moduli
-rw-r--r-- 1 root root 3.2K Feb 10 2018 sshd_config
-rw----- 1 root root 1.7K Jul 19 06:52 ssh_host_rsa_key
-rw-r--r-- 1 root root 1.6K Feb 10 2018 ssh_config
-rw----- 1 root root 411 Jul 19 06:52 ssh_host_ed25519_key
-rw-r--r-- 1 root root 399 Jul 19 06:52 ssh_host_rsa_key.pub
-rw-r--r-- 1 root root 338 Jul 19 06:52 ssh_import_id
-rw----- 1 root root 227 Jul 19 06:52 ssh_host_ecdsa_key
-rw-r--r-- 1 root root 179 Jul 19 06:52 ssh_host_ecdsa_key.pub
-rw-r--r-- 1 root root 99 Jul 19 06:52 ssh_host_ed25519_key.pub

```

Option `-t` mengurutkan file berdasarkan waktu mereka dimodifikasi. Ini akan mendaftar file yang paling baru diubah terlebih dahulu. Option ini dapat digunakan sendiri, tetapi sekali lagi, biasanya lebih bermanfaat ketika dipasangkan dengan option `-l`:

```

sysadmin@localhost:~$ ls -tl /etc/ssh
total 580
-rw----- 1 root root 227 Jul 19 06:52 ssh_host_ecdsa_key
-rw-r--r-- 1 root root 179 Jul 19 06:52 ssh_host_ecdsa_key.pub
-rw----- 1 root root 411 Jul 19 06:52 ssh_host_ed25519_key
-rw-r--r-- 1 root root 99 Jul 19 06:52 ssh_host_ed25519_key.pub
-rw----- 1 root root 1679 Jul 19 06:52 ssh_host_rsa_key
-rw-r--r-- 1 root root 399 Jul 19 06:52 ssh_host_rsa_key.pub
-rw-r--r-- 1 root root 338 Jul 19 06:52 ssh_import_id
-rw-r--r-- 1 root root 553122 Feb 10 2018 moduli
-rw-r--r-- 1 root root 1580 Feb 10 2018 ssh_config
-rw-r--r-- 1 root root 3264 Feb 10 2018 sshd_config

```

Penting untuk diingat bahwa tanggal yang diubah pada direktori mewakili terakhir kali file ditambahkan atau dihapus dari direktori.

Jika file dalam direktori diubah beberapa hari atau bulan yang lalu, mungkin lebih sulit untuk mengetahui dengan pasti kapan mereka dimodifikasi, karena hanya tanggal yang disediakan untuk file yang lebih lama. Untuk informasi waktu modifikasi yang lebih rinci, Anda dapat menggunakan option `--full-time` untuk menampilkan cap waktu lengkap (termasuk jam, menit, detik). Ini akan menganggap option `-l` secara otomatis:

```

sysadmin@localhost:~$ ls -lt --full-time /etc/ssh
total 580
-rw----- 1 root root    227 2018-07-19 06:52:16.000000000 +0000 ssh_host_ecdsa_key
-rw-r--r-- 1 root root    179 2018-07-19 06:52:16.000000000 +0000 ssh_host_ecdsa_key.pub
-rw----- 1 root root    411 2018-07-19 06:52:16.000000000 +0000 ssh_host_ed25519_key
-rw-r--r-- 1 root root     99 2018-07-19 06:52:16.000000000 +0000 ssh_host_ed25519_key.pub
-rw----- 1 root root   1679 2018-07-19 06:52:16.000000000 +0000 ssh_host_rsa_key
-rw-r--r-- 1 root root    399 2018-07-19 06:52:16.000000000 +0000 ssh_host_rsa_key.pub
-rw-r--r-- 1 root root    338 2018-07-19 06:52:16.000000000 +0000 ssh_import_id
-rw-r--r-- 1 root root 553122 2018-02-10 02:31:46.000000000 +0000 moduli
-rw-r--r-- 1 root root   1580 2018-02-10 02:31:46.000000000 +0000 ssh_config
-rw-r--r-- 1 root root   3264 2018-02-10 02:31:46.000000000 +0000 sshd_config

```

Dimungkinkan untuk melakukan sortir terbalik dengan menggunakan option `-r`. Ini dapat digunakan sendiri, atau dikombinasikan dengan option `-S` atau `-t`. *Command* berikut akan mengurutkan file berdasarkan ukuran, terkecil hingga terbesar:

```

sysadmin@localhost:~$ ls -lrs /etc/ssh
total 580
-rw-r--r-- 1 root root     99 Jul 19 06:52 ssh_host_ed25519_key.pub
-rw-r--r-- 1 root root    179 Jul 19 06:52 ssh_host_ecdsa_key.pub
-rw----- 1 root root    227 Jul 19 06:52 ssh_host_ecdsa_key
-rw-r--r-- 1 root root    338 Jul 19 06:52 ssh_import_id
-rw-r--r-- 1 root root    399 Jul 19 06:52 ssh_host_rsa_key.pub
-rw----- 1 root root    411 Jul 19 06:52 ssh_host_ed25519_key
-rw-r--r-- 1 root root   1580 Feb 10 2018 ssh_config
-rw----- 1 root root   1679 Jul 19 06:52 ssh_host_rsa_key
-rw-r--r-- 1 root root   3264 Feb 10 2018 sshd_config
-rw-r--r-- 1 root root 553122 Feb 10 2018 moduli

```

Command berikut akan mencantumkan file berdasarkan tanggal modifikasi, terlama ke terbaru:

```

sysadmin@localhost:~$ ls -lrt /etc/ssh
total 580
-rw-r--r-- 1 root root   3264 Feb 10 2018 sshd_config
-rw-r--r-- 1 root root   1580 Feb 10 2018 ssh_config
-rw-r--r-- 1 root root 553122 Feb 10 2018 moduli
-rw-r--r-- 1 root root    338 Jul 19 06:52 ssh_import_id
-rw-r--r-- 1 root root    399 Jul 19 06:52 ssh_host_rsa_key.pub
-rw----- 1 root root   1679 Jul 19 06:52 ssh_host_rsa_key
-rw-r--r-- 1 root root     99 Jul 19 06:52 ssh_host_ed25519_key.pub
-rw----- 1 root root    411 Jul 19 06:52 ssh_host_ed25519_key
-rw-r--r-- 1 root root    179 Jul 19 06:52 ssh_host_ecdsa_key.pub
-rw----- 1 root root    227 Jul 19 06:52 ssh_host_ecdsa_key

```