

Web Server Sederhana dengan thttpd

Untuk melayani CGI atau html biasa, Anda tidak perlu harus selalu menginstal Apache. Web Server thttpd saja barangkali sudah mencukupi. Hemat, kecil, dan luar biasa.

Tergantung kebutuhan Anda dalam mengatur web server, Anda bisa memilih web server besar seperti Apache atau berbagai web server kecil lainnya. Dari sekian banyak web server kecil yang menarik, thttpd adalah salah satu yang paling menarik. Bagi Anda yang pertama kali mencoba thttpd, dijamin akan terkaget-kaget dan sekaligus terkagum-kagum.

Sesuai namanya, thttpd (t untuk tiny) adalah web server yang berukuran kecil dan cepat. Thttpd sangatlah sederhana, *portable* dan juga aman. Dengan ukurannya yang kecil, dalam kondisi tertentu, thttpd sangatlah cepat, bahkan dapat mengalahkan apache. Thttpd dikatakan sederhana karena hanya menangani spesifikasi HTTP/1.1 yang diperlukan. Namun, pengertian yang diperlukan di sini umumnya sudah dapat memenuhi semua kebutuhan untuk melayani data statis, ditambah dengan sedikit data dinamis. Jadi, sudah lebih dari cukup.

Sementara, walaupun kecil, thttpd dapat berjalan pada sangat banyak platform, mulai dari FreeBSD, Linux sampai Solaris. Terakhir, untuk semua fasilitas yang disediakan tersebut, thttpd terhitung cukup aman karena berjalan pada modus chroot.

Untuk menambah kekaguman Anda, thttpd dapat pula bekerja dengan server side include, aplikasi CGI, proteksi halaman web, halaman web per user dan advanced feature seperti *redirecting* dan *URL-traffic-based throttling* di mana yang terakhir masih belum dimiliki oleh web server lain.

Artikel kali ini akan membahas bagaimana mengatur web server thttpd, mulai dari instalasi, pembuatan script CGI, halaman per user, dan *password protection*.

Untuk instalasi, thttpd terbukti sungguh sederhana. Tidak macam-macam pustaka

yang dibutuhkan. Apabila Anda menggunakan sistem berbasis RPM (contoh: SUSE atau Fedora), maka instalasi akan lebih mudah lagi. Bagi Anda yang ingin melakukan kompilasi dari source code, dapatkan sourcenya di <http://www.acme.com/software/thttpd/>.

Setelah instalasi, Anda bahkan tidak membutuhkan konfigurasi lebih lanjut. Thttpd bisa mempergunakan file konfigurasi, ataupun langsung memberikan opsi pada saat menjalankan thttpd. Satu catatan penting. Seperti yang kita ketahui bahwa thttpd berjalan pada modus chroot, di mana membutuhkan hak root, maka thttpd harus dijalankan oleh user root. Namun, setelah semua hak root yang diperlukan selesai digunakan, maka thttpd akan kembali dijalankan oleh user biasa. Jadi, tetap aman. Mekanisme ini digunakan oleh cukup banyak aplikasi yang pada awalnya membutuhkan hak root, namun tidak membutuhkan lagi setelah beberapa rutin selesai dijalankan. Salah satu contoh aplikasi dengan cara kerja demikian adalah ntop (network monitor). Chroot adalah *system call* yang membatasi program untuk bekerja pada suatu tree direktori seolah-olah direktori tersebut adalah root directory (/). Dengan demikian, program tidak dapat lagi mengakses tree filesystem yang sebenarnya karena hirarki tertinggi (/) telah diatur pada direktori aktif tersebut.

Apabila mempergunakan SUSE, maka Anda bisa memberikan perintah berikut ini untuk menjalankan thttpd (sebagai root):

```
# rcthttpd start
```

Setelah thttpd sukses dijalankan, Anda bisa segera membuka web browser Anda dan mengarahkan ke alamat <http://localhost/>

untuk mendapatkan default home page dari web server ini. Apabila Anda menjumpai halaman web, maka instalasi thttpd Anda telah sukses, dan thttpd siap untuk digunakan lebih lanjut.

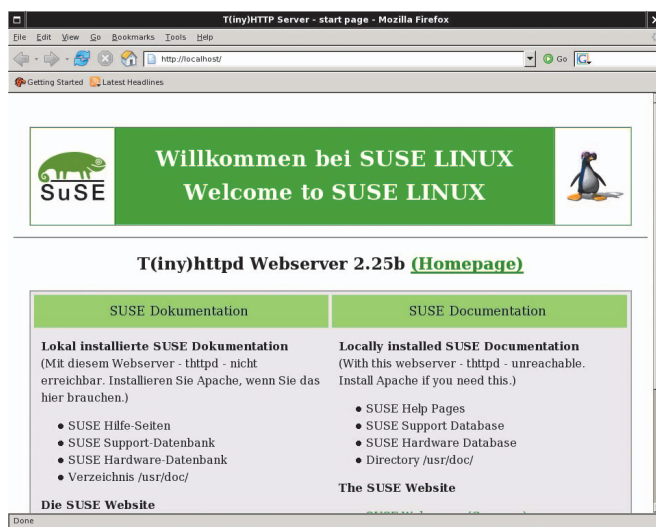
Anda mungkin ingin segera mengubah default homepage tersebut. Masalah ke document root thttpd Anda. Lokasi document root tergantung pada konfigurasi awal pada saat instalasi ataupun dari konfigurasi Anda. Di SUSE, document root untuk thttpd terletak di `/srv/www/htdocs/` seperti halnya Apache. Apabila Anda hanya ingin melayani data statis, Anda bisa segera meng-copy-kan data Anda di dalam direktori tersebut.

CGI di thttpd

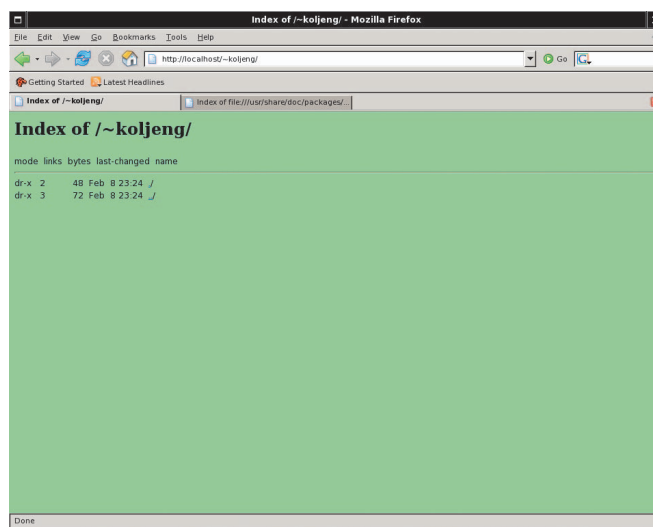
Satu hal yang hebat dari thttpd adalah kemampuannya dalam menangani CGI walaupun menyebut dirinya sebagai web server sederhana. Satu hal yang harus diingat, thttpd berjalan di modus chroot. Oleh karena itu, Anda harus memastikan bahwa program CGI Anda harus mampu dijalankan pada lingkungan chroot tersebut. Atau, program Anda tidak akan berjalan. Pastikan Anda mengopikan semua file yang dibutuhkan ke dalam direktori yang di-chroot (!), atau Anda membangun aplikasi CGI statis.

Sebagai contoh, Anda ingin membangun aplikasi CGI Anda dengan shell. Interpreter yang digunakan adalah `/bin/sh`. Maka, Anda harus memastikan pustaka apa yang diperlukan agar `/bin/sh` dapat berjalan ketika sedang dalam kondisi chroot. Gunakan program `ldd` seperti perintah berikut ini:

```
$ ldd /bin/sh
linux-gate.so.1 => (0xffffe000)
libreadline.so.4 => /lib/
libreadline.so.4 (0x4001e000)
libhistory.so.4 => /lib/
```



Homepage default thttpd SUSE.



Directory listing thttpd.

```
libhistory.so.4 (0x4004a000)
libncurses.so.5 => /lib/
libncurses.so.5 (0x40051000)
libdl.so.2 => /lib/libdl.so.2
(0x40097000)
libc.so.6 => /lib/tls/libc.
so.6 (0x4009a000)
/lib/ld-linux.so.2 => /lib/
ld-linux.so.2 (0x40000000)
```

Setelah itu, Anda bisa mulai meng-copy-kan pustaka-pustaka tersebut sesuai dengan nama direktori dan hirarkinya dalam direktori yang di-chroot oleh thttpd.

Anda keberatan? Begitu juga dengan penulis. Meng-copy-kan dengan cara seperti ini adalah konyol. Apalagi kalau Anda hanya ingin membangun CGI sederhana. Tidak akan sebanding dengan usahanya. Namun, kalau Anda ingin mempergunakan PHP misalnya, mau tidak mau, Anda harus meng-copy-kan PHP ke direktori chroot thttpd. Tidak mungkin membangun interpreter secanggih PHP sendiri. Dan, satu saran, apabila Anda sudah membutuhkan PHP, barangkali thttpd sudah tidak tepat bagi Anda. Gunakan saja Apache.

Bagi Anda yang keberatan, kita akan mempergunakan cara kedua, yaitu membangun aplikasi statis, yang tidak membutuhkan pustaka apapun ketika aplikasi dijalankan. Semua pustaka yang dibutuhkan sudah dikompilasi ke dalam binary aplikasi itu sendiri. Untuk mengetahui statis atau tidaknya suatu aplikasi, Anda bisa kembali mempergunakan program ldd. Apabila ke-

luaran dari ldd adalah seperti keluaran ketika kita memberikan ldd /bin/sh sebelumnya, maka aplikasi adalah aplikasi yang dikompilasi secara dinamis. Namun, apabila keluaran ldd adalah seperti berikut ini, maka aplikasi tersebut dikompilasi secara statis.

```
$ ldd /usr/bin/ldd
not a dynamic executable
```

Kita akan membangun sebuah program CGI yang akan mencetak tabel perkalian 5x5 ke web browser, dan aplikasi ini akan kita kompilasi secara statis. Berikut ini adalah *source code* aplikasi dalam bahasa C:

```
#include <stdio.h>

int main()
{
    int i=0;
    int j=0;
    printf("content-type: text/
plain\n\n");
    printf("Hello, ini cgi yang
dijalankan oleh thttpd\n");
    printf("Berikut ini adalah
tabel perkalian 1 sampai
5\n\n");
    for (i=1; i<=5; i++)
    {
        for (j=1; j<=5; j++)
        {
            printf(" %02d x %02d =
%02d ", i, j, i*j);
        }
        printf("\n");
    }
}
```

```
return 0;
```

```
}
```

Untuk contoh, simpanlah di direktori cgi-bin thttpd Anda (di SUSE, direktori cgi-bin thttpd adalah di /srv/www/htdocs/cgi-bin) dengan nama test-cgi.c. Setelah itu, lakukanlah kompilasi dengan perintah berikut ini:

```
$ gcc -static -o test-cgi test-
cgi.c
```

Opsi -static akan menyebabkan aplikasi kita dikompilasi secara statis. Semua pustaka yang dibutuhkan akan dimasukkan ke dalam binari aplikasi. Tentu saja, binari aplikasi kemudian akan menjadi besar. Di sistem penulis, contoh program tersebut akan berukuran lebih dari 2 MB setelah kompilasi. Anda bisa melakukan stripping dengan program objcopy apabila diinginkan.

Setelah kompilasi selesai, Anda bisa menjalankan test-cgi dengan mengarahkan browser Anda ke <http://localhost/cgi-bin/test-cgi>. Anda akan mendapatkan sebuah tabel perkalian 5x5. Apabila ingin lebih rapi, sebenarnya kita bisa mengganti content-type dari text/plain ke text/html dan memberikan kode yang Anda inginkan.

Tentu saja aplikasi CGI yang kita bangun tidak dapat mencerminkan CGI yang sesungguhnya dalam hidup sehari-hari. Apabila Anda termasuk salah seorang yang nekad dan menjadi sangat fanatik dengan CGI, Anda bisa mempelajari pemrograman

CGI dari berbagai tutorial di internet. Penulis pernah mendapatkan satu program CGI yang sangat menarik beberapa tahun lalu: sebuah webmail komplit yang hanya membutuhkan 1 buah file! Dibangun dengan C++!

User directory di thttpd

Ketika kali pertama berkenalan dengan Apache dan berhasil mengonfigurasi user dir (setelah puluhan kali gagal mengonfigurasi Apache), betapa senangnya penulis ketika bisa menyediakan juga alamat `http://server/~user/` yang selama ini hanya bisa penulis dapatkan di Internet. Nah, kemampuan untuk menghadirkan *user directory* di web server ini bukan hanya milik Apache. Namun, thttpd pun bisa. Dengan demikian, user meletakkan file direktori sendiri tanpa harus menulis ke document root.

thttpd datang dengan sebuah program dengan nama makeweb, yang dijalankan oleh setiap user yang ingin memiliki user dir sendiri. Setelah dijalankan, program ini akan membuat direktori dengan nama user di bawah `/users/` di document root Anda. Dengan demikian, apabila user nop menjalankan makeweb, maka akan direktori `/srv/www/htdocs/users/nop/` akan dibuat. Kepemilikan akan diserahkan kepada user nop dan di dalam home direktori nop, sebuah symlink dengan nama `public_html` akan diarahkan ke `/srv/www/htdocs/users/nop/` tersebut. Hebat, bukan?

Anda tidak harus mengarahkan ke `http://localhost/users/<nama_user>` untuk dapat

mengakses webpage user. Anda dapat menggunakan notasi `~` sehingga webpage user dapat diakses dengan pola alamat `http://server/~user`. Apabila browser diarahkan ke alamat user, maka *directory listing* akan ditampilkan. Namun, tampilannya tidak sama persis dengan Apache. Lebih sederhana dan cukup unik.

Basic Authentication

Ini adalah salah satu fitur thttpd yang luar biasa. Walaupun kecil dan sederhana, thttpd mengizinkan pula basic authentication seperti halnya Apache dan file `.htaccess`nya yang mempergunakan Basic Auth. Caranya mudah sekali dan tidak mengutak-atik sedikitpun konfigurasi.

Pertama-tama, tujuhan direktori yang ingin diprotek. Sebagai contoh, penulis memiliki direktori `/srv/www/htdocs/users/koljeng/secret/`. Maka, pertama-tama, penulis akan aktif di direktori tersebut.

Setelah itu, gunakan program `htpasswd` untuk membuat file database user. Sebagai contoh:

```
$ htpasswd -c .htpasswd nop
Adding password for nop.
New password:
Re-type new password:
```

Dalam contoh tersebut, penulis membuat file database user dengan nama file `.htpasswd` dan membuat user baru `nop`. Kini, halaman `~/koljeng/secret/` hanya dapat diakses oleh user `nop` apabila password yang benar disertakan.

Advanced Feature

Kita akan membahas beberapa fitur lanjutan thttpd. Salah satunya adalah fitur *redirect*. Anda dapat melakukan redirect ke alamat lain dengan cara yang mudah. Bagi Anda yang berminat, Anda bisa membaca manual program *redirect* yang disertakan bersama web server ini.

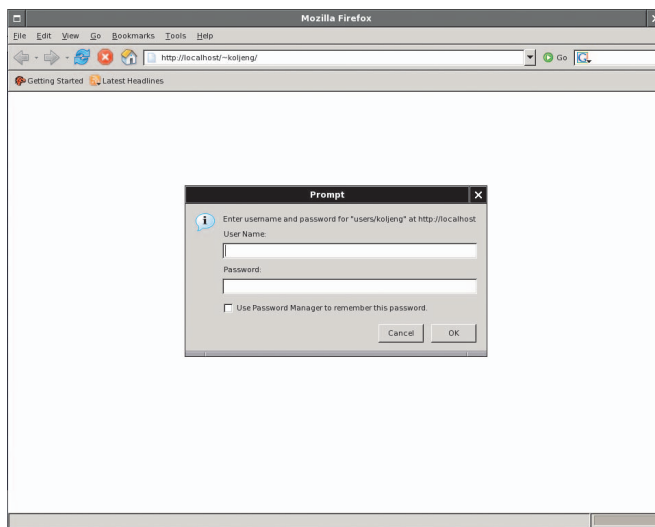
Fitur *advanced* lainnya adalah *throttling*. Dengan adanya fitur ini, kita bisa mengatur byte rate maksimal untuk URL atau grup URL. Kita juga dapat mengatur ukuran minimal apabila diinginkan. Throttling bekerja dengan memeriksa URL yang dilayani oleh thttpd. Setelah itu, pencocokan ke dalam throttle file akan dilakukan dan perhitungan bandwidth yang telah dikonsumsi akan dilakukan.

Selain itu, Anda juga dapat menyediakan virtual host dengan web server kecil-kecil cabe rawit ini. Luar biasa, bukan?

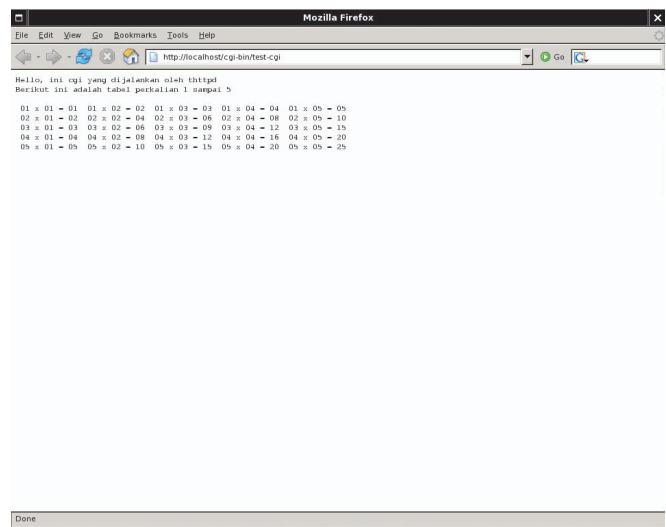
Apabila diinginkan, Anda juga dapat membangun *custom error* untuk kesalahan yang terjadi. Dengan demikian, halaman web Anda akan terlihat lebih ramah. Dan, hal ini bisa dilakukan dengan mudah, sesuai petunjuk di manual.

Demikianlah, kita telah sampai pada akhir pembahasan. Ketika kali pertama mencoba thttpd, penulis tidak habis kagum akan kemampuan server kecil ini. Penulis yakin bahwa banyak dari Anda pun akan merasakan yang sama setelah mencoba web server ini. Cobalah sendiri, dan rasakan kehebatannya! 🐱

Noprianto (noprianto@infolinux.co.id)



Basic Authentication thttpd.



Tabel perkalian 5x5 aplikasi CGI.