Remote X Apps Mini-HOWTO

Vincent Zweije, zweije@xs4all.nl Diterjemahkan oleh: Yaya Heryadi, yheryadi@necnusa.co.id v, 14 July 1998, Terj. 8 Jan 1999

Mini-HOWTO ini menjelaskan cara untuk menjalankan aplikasi-aplikasi X secara remote. Lebih jelasnya adalah bagaimana membuat tampilan dari sebuah program X untuk ditampilkan pada sebuah komputer lain yang berbeda dengan komputer tempat dijalankannya program tadi. Atau sebaliknya, bagaimana membuat sebuah program X berjalan pada komputer yang berbeda dengan komputer tempat Anda bekerja. Fokus dari mini-HOWTO ini adalah aspek keamanan.

Contents

1 Pendahuluan

Mini-HOWTO ini merupakan pedoman mengenai cara menjalankan aplikasi X secara remote. Dokumen ini ditulis dengan beberapa alasan yaitu:

1. Beberapa pertanyaan mengenai cara menjalankan aplikasi X secara remote telah muncul di usenet.

2. Saya melihat telah banyak, banyak saran mengenai "penggunaan `xhost +hostname`" atau bahkan "`xhost +`" untuk memberikan koneksi X. Cara-cara di atas sangat tidak aman, sedangkan ada metoda lain yang lebih baik.

3. Penulis tidak mengetahui keberadaan sebuah dokumen sederhana yang menjelaskan pilihan-pilihan yang Anda miliki. Silahkan memberi-tahu penulis zweije@xs4all.nl apabila Anda mengetahui lebih banyak lagi.

Dokumen ini ditulis dengan berpedoman kepada sistem-sistem yang mirip UNIX. Meskipun sistem operasi lokal maupun remote Anda sedikit berbeda, disini Anda akan mendapatkan cara yang dibutuhkan untuk melakukannya. Namun Anda harus menyesuaikan sendiri contoh-contoh untuk Anda gunakan pada sistem Anda.

Versi terakhir dari dokumen ini selalu tersedia di dalam WWW pada `<http://www.xs4all.nl/~zweije/xauth.html>`. Dokumen ini juga tersedia dengan judul Linux Remote X Apps mini-HOWTO pada `<http://sunsite.unc.edu/LDP/HOWTO/mini/Remote-X-Apps>`. Linux (mini-) HOWTO tersedia melalui http atau ftp dari `sunsite.unc.edu`.

Versi dokumen ini adalah 0.5.1. Tidak ada pemberian jaminan apapun kecuali adanya niat baik. Penulis terbuka untuk menerima berbagai saran, gagasan, penambahan, penunjuk yang berguna, koreksi penulisan (tipo), dsb...

Penulis menginginkan agar dokumen ini tetap merupakan sebuah dokumen yang mudah dibaca serta sederhana, merupakan gaya terbaik dari HOWTO. Semua flame akan dibuang ke dalam `/dev/null`.

Isi dokumen terakhir diperbaiki pada tanggal 14 Juli 1998 oleh Vincent Zweije.

2 Bahan bacaan yang terkait

Sebuah dokumen yang terkait dengan dokumen ini serta terdapat di dalam WWW berjudul "*What to do when Tk reports that your display is unsecure*" ("*Apa yang harus dilakukan jika Tk melaporkan bahwa tampilan Anda tidak aman*"), `<http://ce-toolkit.crd.ge.com/tkxauth/>`. Dokumen di atas ditulis oleh Kevin Kenny. Tulisan itu memberikan solusi yang mirip dengan otentikasi X yang diberikan di dalam dokumen ini. Namun, Kevin lebih bermaksud untuk menggunakan `xadm` untuk Anda gunakan dalam mengatur `xauth`.

Sebuah buku berjudul *The X System Window System Vol 8 X* "*Window System Administration's Guide*" terbitan *O'Reilly and Associates* telah diberitahukan kepada penulis sebagai satu sumber informasi yang baik. Sayang sekali, penulis belum sempat mendapatkannya.

Sebuah dokumen lain yang sangat mirip dengan dokumen yang Anda sedang baca berjudul "*Securing X Window*" ("*Mengamankan X Window*") tersedia di `<http://ciac.llnl.gov/ciac/documents/ciac2316.html>`.

Periksalah usenet newsgroup, seperti `<news:comp.windows.x>`, `<news:comp.os.linux.x>`, dan `<news:comp.os.linux.networking>`.

3 Situasi

Anda sedang menggunakan dua buah komputer. Anda sedang menggunakan sistem X Window di dalam komputer pertama untuk menulis dan melihat hasilnya. Anda juga sedang menggunakan komputer kedua untuk mengerjakan pekerjaan grafik yang penting. Anda menginginkan agar komputer kedua menampilkan outputnya ke dalam tampilan komputer pertama. Sistem X Window memungkinkan hal itu.

Tentu saja, Anda membutuhkan sebuah koneksi terhadap jaringan untuk maksud ini. Lebih disukai jaringan yang berkecepatan tinggi; Protokol X merupakan pemakai berat jaringan. Namun dengan sedikit kesabaran serta penggunaan kompresi protokol yang tepat, Anda bahkan dapat menjalankan aplikasi-aplikasi melalui sebuah modem. Mengenai kompresi protokol X, Anda mungkin perlu memeriksa `dxpc` `<http://ccwf.cc.utexas.edu/~zvonler/dxpc>` atau LBX `<http://www.ultranet.com/~pauld/faqs/LBX-HOWTO.html>` (juga dikenal sebagai LBX mini-HOWTO)

Anda harus melakukan dua hal untuk mencapai semua itu

1. Memerintahkan tampilan lokal (server) untuk menerima koneksi dari komputer remote.
2. Memerintahkan aplikasi remote (klien) untuk mengarahkan outputnya ke tampilan lokal.

4 Sedikit teori

Kata ajaibnya adalah `DISPLAY`. Di dalam sistem X Window, sebuah tampilan (display) terdiri dari (secara sederhana) sebuah keyboard, mouse, dan layar. Sebuah tampilan dikelola oleh sebuah program server, dikenal sebagai X Server. X Server tadi melayani program-program lain yang terhubung dengannya dalam hal berbagai kemampuan penampilan.

Sebuah tampilan ditunjukkan dengan sebuah nama, misalnya

- `DISPLAY=light.uni.verse:0`
- `DISPLAY=localhost:4`
- `DISPLAY=:0`

Sebuah tampilan terdiri dari sebuah hostname (misalnya `light.uni.verse` dan `localhost`), titik dua (`:`), dan nomor urut (misalnya `0` atau `4`). Hostname bagi sebuah tampilan adalah nama komputer tempat berjalannya X Server. Ketiadaan hostname berarti komputer lokal. Nomor urut yang biasanya 0 - nomor itu dapat bervariasi jika terdapat beberapa tampilan yang terhubung dengan satu komputer.

Jika Anda pernah menemukan sebuah penunjukkan tampilan dengan sebuah tambahan .n melekat padanya, tambahan itu menunjukkan nomor layar. Sebuah tampilan sesungguhnya dapat memiliki layar berganda. Meskipun biasanya hanya terdapat satu layar saja, dengan nomor n = 0, sehingga ini merupakan nilai default.

Terdapat bentuk-bentuk lain dari `DISPLAY` namun bentuk di atas sudah memadai bagi keperluan kita.

5 Memerintahkan klien

Program klien (misalnya aplikasi grafik Anda) mengetahui tampilan mana untuk dihubungi dengan memeriksa variabel environment `DISPLAY`. Pengaturan ini dapat di atasi dengan memberikan argumen perintah baris `-display hostname:0` pada klien pada saat klien mulai berjalan. Nanti akan disajikan beberapa contoh untuk memperjelas hal ini.

Komputer kita diketahui oleh fihak luar dengan nama light, dan kita terdapat di dalam suatu domain `uni.verse`. Jika kita menjalankan sebuah X Server secara normal, tampilannya dikenal dengan nama `light.uni.verse:0`. Kita ingin menjalankan program pembuatan gambar `xfig` pada sebuah komputer remote, yang disebut `dark.matt.er`, dan menampilkan outputnya disini pada komputer light.

Dalam hal ini Anda dianggap telah melakukan telnet ke dalam komputer remote, `dark.matt.er`.

Jika Anda memiliki `csh` yang berjalan pada komputer remote:

```
dark% setenv DISPLAY light.uni.verse
dark% xfig &
```

atau alternatifnya adalah:

```
dark% xfig -display light.uni.verse:0 &
```

Jika Anda memiliki sh berjalan pada komputer remote:

```
dark% DISPLAY=light.uni.verse
dark% export DISPLAY
dark% xfig &
```

Atau alternatifnya adalah:

```
dark% DISPLAY=light.uni.verse:0 xfig &
```

Atau, tentu saja:

```
dark% xfig -display light.uni.verse:0
```

Nampaknya beberapa versi telnet dapat mengirimkan variabel `DISPLAY` ke remote host secara otomatis. Jika Anda memiliki versi ini, Anda beruntung, sehingga Anda tidak perlu mengaturnya secara manual. Jika tidak, hampir semua versi dari telnet mengirimkan variabel environment `TERM`; dengan cara hacking yang jitu maka pengiriman variabel `DISPLAY` dapat dilakukan dengan menumpang ke dalam variabel `TERM`.

Ide untuk mengikut-sertakan nilai variabel tadi adalah dengan menulis sebuah script untuk melakukan hal berikut: sebelum melakukan telnet, lekatkan nilai variabel `DISPLAY` ke dalam variabel `TERM`. Kemudian dilakukan telnet keluar. Pada sisi komputer remote, di dalam suatu file `.*shrc`, pisahkan nilai `DISPLAY` dari variabel `TERM`.

6 Memerintahkan server

X Server tidak akan menerima permintaan koneksi dari sembarang tempat. Anda tidak menginginkan setiap orang menampilkan windows pada layar Anda. Ataupun membaca setiap yang Anda tulis - perlu diingat bahwa keyboard Anda merupakan bagian dari tampilan!

Terlalu sedikit orang yang menyadari bahwa pemberian akses terhadap tampilan Anda mengandung suatu risiko keamanan. Seseorang yang mampu melakukan akses ke dalam tampilan Anda akan dapat membaca dan menulis ke dalam layar Anda, membaca tulisan Anda, bahkan membaca pergerakan mouse Anda.

Hampir semua server memiliki dua cara untuk melakukan otentikasi terhadap koneksi terhadapnya, yaitu: mekanisma daftar host (`xhost`) dan mekanisma cookie ajaib (`xauth`). Terdapat pula ssh, yaitu shell yang aman, yang mampu mengarahkan koneksi X.

6.1 Xhost

Xhost memberikan akses berdasarkan nama komputer (host). Server memiliki sebuah daftar berisi setiap host yang diizinkan melakukan koneksi terhadapnya. Xhost dapat pula menon-aktifkan pemeriksaan host secara keseluruhan. Mohon berhati-hati: dengan menon-aktifkan pemeriksaan host berarti pemeriksaan ditiadakan, sehingga setiap host akan dapat melakukan koneksi!

Anda dapat melakukan kontrol terhadap daftar host yang dimiliki server dengan menggunakan program xhost. Untuk menggunakan mekanisme ini pada contoh di atas, lakukan:

```
light$ xhost  +dark.matt.er
```

Perintah di atas mengizinkan semua koneksi yang berasal dari host `dark.matt.er`. Tidak lama sesudah klien X membuat koneksi dan menampilkan sebuah window, untuk kepentingan keamanan, tarik kembali pemberian izin untuk membuat lebih banyak koneksi dengan cara:

```
        light$  xhost  -dark.matt.er
```

Anda dapat menon-aktifkan pemeriksaan host dengan cara:

```
        light$ xhost +
```

Perintah di atas menon-aktifkan pemeriksaan akses terhadap host sehingga mengizinkan semua orang untuk membuat koneksi. Anda seharusnya tidak melakukan hal ini di dalam suatu jaringan yang Anda sendiri tidak mempercayai semua usernya (seperti pada Internet). Anda dapat mengaktifkan kembali pemeriksaan host dengan cara:

```
        light$ xhost -
```

`xhost` - tidak dapat menghilangkan sendiri semua host dari daftar akses (menjadi sangat tidak bermanfaat - Anda tidak dapat melakukan koneksi tempat manapun, bahkan dari komputer lokal Anda sekalipun).

Xhost merupakan suatu mekanisme yang tidak aman. Dalam hal ini xhost tidak membedakan pemakai yang berbeda di dalam komputer remote. Demikian pula, hostname (yang sesungguhnya merupakan alamat) dapat dipalsukan (spoofing). Keadaan ini sangat buruk jika Anda berada di dalam suatu jaringan yang tidak memiliki jaminan keamanan (misalnya sesudah membuat dialup PPP untuk mengakses Internet).

6.2 Xauth

Xauth memberikan akses kepada setiap orang yang mengetahui rahasia yang dibutuhkan. Rahasia yang dimaksud adalah rekord otorisasi, atau cookie ajaib. Skema otorisasi ini secara formal disebut sebagai `MIT-MAGIC-COOKIE-1`.

Semua cookie dari tampilan-tampilan yang berbeda disimpan di dalam file `~/.Xauthority`. File `~/.Xauthority` yang Anda miliki tidak boleh dapat diakses oleh group/pemakai lain manapun. Program xauth mengelola cookie tsb, itulah sebabnya nama kependekan xauth digunakan untuk skema ini.

Pada sessi permulaan, server membaca sebuah cookie dari sebuah file yang ditunjukkan oleh argumen `-auth`. Sesudah itu, server tadi hanya mengizinkan koneksi dari klien-klien yang mengetahui cookie yang sama. Jika cookie di dalam file `~/.Xauthority` berubah, server tidak akan menerima perubahan yang terjadi.

Server-server baru dapat menghasilkan langsung cookie pada saat klien-klien memintanya. Semua cookie tetap disimpan di dalam server; Cookie tadi tidak berakhir di dalam `~/.Xauthority` kecuali apabila seorang klien meletakkan ke dalamnya. Seperti yang telah disampaikan oleh David Wiggins:

Perkembangan lanjutan telah ditambahkan ke dalam X11R6.3 yang mungkin akan menarik perhatian Anda. Dengan penambahan SECURITY baru, X Server sendiri sambil berjalan dapat membuat dan mengembalikan cookies baru. Lebih lanjut, cookies tadi dapat ditandai sebagai "tidak dipercaya" sehingga aplikasi-aplikasi yang membuat koneksi menggunakan cookie tadi akan dibatasi operasinya. Misalnya, aplikasi tadi tidak mampu untuk mencuri input keyboard/mouse, atau isi window, dari klien lain yang tidak dipercayai. Sebuah sub-perintah dari xauth yang baru dibuat memungkinkan penggunaan fasilitas ini, jika tidak mudah.

Xauth memiliki suatu kelebihan yang jelas dalam hal keamanan dibandingkan xhost. Anda dapat membatasi akses user-user tertentu ke dalam komputer tertentu. Xauth tidak terancam dengan pemalsuan alamat seperti pada xhost. Dan jika Anda menginginkannya, Anda masih dapat menggunakan xhost bersama-sama dengan xauth untuk memberikan koneksi.

6.2.1 Pembuatan cookie

Jika anda ingin menggunakan xauth, Anda harus menjalankan X Server menggunakan argumen -auth authfile. Jika Anda menggunakan script startx, script itu merupakan tempat yang tepat untuk melakukannya. Buatlah rekord otorisasi seperti yang diberikan berikut ini di dalam startx yang Anda miliki.

Merupakan bagian yang diambil dari `/usr/X11R6/bin/startx`:

```
mcookie | sed  -e  's/^add  :0  . /' | xauth -q
xinit -- -auth "$HOME/X.authority"
```

`mcookie` merupakan sebuah program kecil dari paket util-linux, situs utamanya adalah `<ftp://ftp.math.uio.no/pub/linux/>`. Secara alternatif, Anda dapat menggunakan **md5sum** untuk mengambil beberapa data secara acak (misalnya dari `/dev/urandom` atau `ps -ax`) dan memasukkannya ke dalam format cookie:

```
dd  if=/dev/urandom  count=1|md5sum|sed -e 's/^/add  :0  . /'|xauth -q
xinit -- -auth  "$HOME/.Xauthority"
```

Jika Anda tidak dapat mengedit script startx (karena Anda bukan root), mintalah administrator sistem Anda untuk melakukan setup terhadap startx sesuai dengan yang diinginkan, atau meminta dia untuk mensetup xdm. Jika dia tidak dapat atau dia tidak mau melakukannya, Anda dapat membuat sebuah script `~/.xserverrc`. Jika Anda memilikinya, script ini dijalankan oleh xinit dan bukan oleh X Server yang sebenarnya. Setelah itu, Anda dapat mulai menjalankan X Server yang sebenarnya dari script ini menggunakan argumen yang tepat. Untuk dapat melakukannya, buatlah agar `~/.xserverrc` Anda menggunakan baris cookie ajaib di atas untuk membuat sebuah cookie lalu mengeksekusi X Server yang sebenarnya:

```
#!/bin/sh
mcookie|sed -e 's/^/add  :0  . /'|xauth -q
exec  /usr/X11R6/bin/X  "$@" -auth "$HOME/.Xauthority"
```

Jika Anda menggunakan xdm untuk mengelola sessi X Anda, Anda dapat menggunakan xauth dengan mudah. Definisikan resource untuk `DisplayManager.authDir` di dalam `/etc/X11/xdm/xdm-config`. Xdm akan memberikan argumen `-auth` tadi ke X Server pada saat mulai dijalankan. Pada saat Anda login sebagai xdm, xdm akan meletakkan cookie di dalam `~/.Xauthority` Anda. Silahkan baca *xadm(1)* untuk mendapatkan informasi lebih lanjut. Sebagai contoh `/etc/X11/xdm/xdm-config` milik penulis memiliki baris berikut:

```
DisplayManager.authDir: /var/lib/xdm
```

6.2.2 Pengiriman cookie

Sekarang Anda telah menjalankan sessi X di dalam server host `light.uni.verse` dan telah memiliki cookie di dalam `~/.Xauthority`, Anda harus mengirimkan cooke tadi ke client host, `dark.matt.er`.

Cara termudah adalah ketika direktori-direktori home di dalam komputer light dan dark digunakan secara bersama-sama. File `~/.Xauthority` memiliki isi yang sama, sehingga cookie akan dikirimkan segera. Namun ada suatu kendala: pada saat Anda meletakkan sebuah cookie untuk `:0` di dalam `~/.Xauthority`, komputer dark akan mengira bahwa cookie itu disediakan untuknya dan bukan untuk komputer light. Anda harus memberikan hostname secara eksplisit ketika Anda membuat sebuah cookie; Anda tidak dapat mengosongkannya. Anda dapat menginstal cookie yang sama untuk `:0` dan `light:0` dengan cara:

```
#!/bin/sh
cookie=`mcookie`
xauth  add  :0  .  $cookie
xauth  add  "$HOST:0" . $cookie
exec  /usr/X11R6/bin/X "$@" -auth "$HOME/.Xauthority"
```

Jika direktori-direktori home tidak dipakai secara bersama-sama, Anda dapat mengirimkan cookie tadi dengan menggunakan rsh, remote shell:

```
light$ xauth  nlist  :0  | rsh  dark.matt.er  xauth  nmerge  -
```

1. Ambillah cookie dari file lokal Anda `~/.Xauthority` (`xauth nlist :0`)
2. Transfer cookie tadi ke `dark.matt.er` (`| rsh dark.matt.er`)
3. Letakkan di dalam file `~/.Xauthority` (`xauth nmerge -`)

rsh mungkin tidak berjalan sesuai dengan keinginan Anda. Selain itu, rsh juga memiliki sebuah kelemahan dalam keamanan (pemalsuan hostname, jika ingatan penulis tidak salah). Jika Anda tidak akan atau tidak mau menggunakan rsh, Anda dapat mengirimkan cookie secara manual sbb:

```
light$ echo $DISPLAY
:0
```

```
light$ xauth list $DISPLAY
light/unix:0 MIT-MAGIC-COOKIE-1 076aaecfd370fd2af6bb9f5550b26926
light$ rlogin dark.matt.er
Password:
dark% setenv DISPLAY light.uni.verse:0
dark% xauth add $DISPLAY . 076aaecfd370fd2af6bb9f5550b26926
dark% xfig &
[15332]
dark% logout
light$
```

Silahkan juga baca rsh(1) dan xauth(1) untuk mendapat informasi selanjutnya.

Memasukkan cookie ke dalam variabel-variabel `TERM` ataupun `DISPLAY` pada saat Anda melakukan telnet ke host remote merupakan suatu hal yang mungkin dilakukan. Cara yang dilakukannya sama dengan cara untuk memasukkan variabel `DISPLAY` ke dalam variabel `TERM`. Bacalah Bagian 5: Memerintahkan klien. Dalam pandangan penulis Anda harus menghadapi resiko sendiri, namun penulis berminat untuk mengetahui jika ada yang bermaksud memberi konfirmasi ataupun sanggahan.

6.2.3 Penggunaan cookie

Sebuah aplikasi di dalam komputer dark.matt.er, seperti xfig di atas, akan secara otomatis memeriksa file /.Xauthority untuk mencari cookie guna melakukan otentikasi bagi dirinya.

6.3 Ssh

Rekord otoritas akan dikirim tanpa menggunakan enkripsi. Jika anda khawatir kalau ada orang yang akan mencium data itu, gunakanlah ssh, secure shell. Ssh akan melakukan pengiriman X melalui koneksi berenkripsi. Disamping itu, ssh juga berguna untuk hal-hal. Ssh merupakan peningkatan struktur yang baik bagi sistem Anda. Kunjungilah homepage ssh di `<http://www.cs.hut.fi/ssh/>`.

Siapa yang mengetahui skema otentikasi lainnya untuk melakukan enkripsi koneksi X? Mungkinkan digunakan kerberos?

7 Troubleshooting

Pertama kali Anda mencoba menjalankan sebuah aplikasi X secara remote biasanya tidak akan berjalan baik. Inilah beberapa pesan error yang umum, kemungkinan penyebabnya, dan solusi untuk membantu Anda.

```
xterm  Xt  error:  Can't open display:
```

Variabel `DISPLAY` tidak terdapat di dalam environment, demikian pula Anda tidak memberikan flag -display. Aplikasi mengasumsikannya sebagai kosong namun secara sintaks tidak dapat diterima. Untuk mengatasinya, pastikan bahwa Anda telah men-set variable `DISPLAY` di dalam environment dengan benar (menggunakan setenv atau export tergantung shell yang Anda gunakan).

```
_X11TransSocketINETConnect:  Can't connect: errno = 101
xterm  Xt  error; Can't open display: love.dial.xs4all.nl:0
```

Errno 101 adalah "Network is unreachable" ("Jaringan tidak dapat dicapai"). Aplikasi tidak dapat membuat suatu koneksi terhadap server melalui jaringan. Periksalah untuk mengetahui apakah Anda memiliki set `DISPLAY` yang benar, serta apakah komputer server dapat dicapai dari klien Anda (seharusnya dapat dicapai, karena Anda mungkin telah login ke dalam server tadi dan melakukan telnet terhadap klien).

```
_X11TransSocketINETConnect:  Can't connect: errno = 111
xterm  Xt  error; Can't open display: love.dial.xs4all.nl:0
```

Errno 111 adalah "Connection refused" ("Koneksi ditolak"). Komputer server yang sedang Anda coba buat koneksi dapat dicapai namun server yang ditunjukkan di atas tidak terdapat di dalamnya. Periksalah apakah Anda telah menggunakan nama host dan nomor tampilan yang benar.

```
Xlib: connection to ":0.0" refused by server
Xlib: Client is not authorized to connect to Server
xterm  Xt error: Can't open display: love.dial.xs4all.nl:0.0
```

Klien berhasil membuat koneksi dengan server namun server tidak mengizinkan klien tadi untuk menggunakannya (tidak berwenang). Pastikan bahwa Anda telah mengirimkan klien cookie ajaib yang tepat, dan cookie tadi belum kadaluarsa (server menggunakan sebuah cookie baru pada saat memulai sebuah sessi baru).

8 Catatan dari penterjemah

Penterjemah sudah berusaha menterjemahkan tulisan aslinya sedapat mungkin, namun tidak mustahil masih ada kekurangan dalam penterjemahan di sana-sini. Apabila ada saran-saran untuk memperbaiki penterjemahan tulisan ini silahkan disampaikan via e-mail ke penterjemah, yheryadi@necnusa.co.id, atau koordinator id LDP, mdamt@linux.or.id.