X aplikacije na daljinu mini-KAKO

Vincent Zweije, *zweije@xs4all.nl* 19. studenog 1999.

Ovaj mini-KAKO opisuje kako pokretati udaljene X aplikacije. To jest, kako prikazati X program na drugom računalu, a ne onom na kojem radi. Ili obratno: kako natjerati X program da radi na drugom računalu, a ne onom pred kojim sjedite. Ovaj mini-KAKO usredotočen je na sigurnost. Također sadrži informacije o lokalnom pokretanju X aplikacija, ali kao drugi korisnik.

Sadržaj

1 Uvod

Ovo je (ili bi trebao biti) vodič u pokretanju X aplikacija preko mreže. Usredotočen je na sigurnost. Napisao sam ga iz nekoliko razloga.

1. Na USENET-u se stalno pojavljuju pitanja o pokretanju udaljenih X aplikacija.
2. Vidim mnogo, mnogo preporuka za korištenje `xhost +računalo` ili čak `xhost +` kako bi se omogućile X veze. **Njihova sigurnost je smiješna**, a ima boljih načina.
3. Ne znam za jednostavan dokument koji opisuje **te** načine. Ako vi znate, obavijestite me na *zweije@xs4all.nl*.

Ovaj dokument napisan je za Unix-olike sustave. I ako je vaš lokalni ili udaljeni operacijski sustav nešto drugo, ovdje ćete shvatiti kako stvari rade. Međutim, primjere ćete morati prevesti tako da odgovaraju vašem operacijskom sustavu.

Najnovija verzija ovog dokumenta uvijek se može dobiti WWW-om na `<http://www.xs4all.nl/~zweije/xauth.html>`. Također je dostupna kao *Linux Remote X Apps mini-HOWTO* na `<http://sunsite.unc.edu/LDP/HOWTO/mini/Remote-X-Apps>`. Linux (mini-)HOWTO-i dostupni su HTTP-om ili FTP-om sa *sunsite.unc.edu*.

Ovo je verzija 0.6.1. Nema jamstava, samo dobre namjere. Rado primam prijedloge, ideje, dodatke, korisne pokazivače, (pravopisne) ispravke, itd.... Želim da ovaj dokument ostane jednostavan i čitljiv, u najboljem KAKO stilu. Flejmovi u `/dev/null`.

Sadržaj zadnji put osvježio Vincent Zweije `<http://www.xs4all.nl/~zweije/index.html>` 19. studenog 1999.

2 Povezani dokumenti

Kevin Kenny je napisao srodan dokument, *What to do when Tk says that your display is insecure* (što učiniti kad Tk kaže da je vaš prikaz nesiguran). Nalazi se na `<http://ce-toolkit.crd.ge.com/tkxauth/>`. Predlaže slično rješenje X autorizaciji (`xauth`). No Kevin se više usmjerio na olakšavanje uporabe `xautha` preko `xdma`.

Također sam obavješten da je dobar izvor informacija *The X System Window System Vol. 8 X Window System Administrator's Guide* kojeg je izdao O'Reilly and Associates. Nažalost, nisam uspio provjeriti.

I još jedan dokument vrlo sličan onom kojeg čitate, naslova *Securing X Windows*, može se naći na `<http://ciac.llnl.gov/ciac/documents/ciac2316.html>`.

Također pogledajte usenet grupe kao što je *comp.windows.x*, *comp.os.linux.x* i *comp.os.linux.networking*.

3 Prizor

Koristite dva računala. Na prvom gledate i tipkate u X Window System. Na drugom obavljate neke važne grafičke poslove. Hoćete da drugo rezultate pokaže na prvom. X Window System to čini mogućim.

Naravno, prvo vam treba veza između njih. I to što brža; X protokol užasno je zahtjevan. Ali, uz malo strpljenja i odgovarajuće sažimanje prokola, aplikacije možete pokretati i preko modema. Ako vas zanima kompresija X protokola, pogledajte dxpc, `<http://ccwf.cc.utexas.edu/~zvonler/dxpc/>` ili LBX, `<http://www.ultranet.com/~pauld/faqs/LBX-HOWTO.html>` (također poznat kao *LBX mini-HOWTO* `<http://sunsite.unc.edu/LDP/HOWTO/mini/LBX>`).

Za sve to trebate napraviti dvije stvari:

1. Reći lokalnom prikazu (serveru) da prihvaća veze s udaljenog računala.
2. Reći udaljenoj aplikaciji (klijentu) da se pokaže na vašem prikazu.

4 Malo teorije

Čarobna je riječ DISPLAY (prikaz). Na X Window Systemu zaslon se (pojednostavljeno) sastoji od tipkovnice, miša i zaslona. O prikazu se brine program poznat kao X server. Server sposobnost prikaza daje drugim programima koji su na njega spojeni.

Prikaz ima svoje ime, kao što je:

- `DISPLAY=svemir.svjet.la:0`
- `DISPLAY=localhost:4`
- `DISPLAY=:0`

Ono se sastoji od imena računala (kao što je `svemir.svjet.la` ili `localhost`), dvotočke (`:`) i nastavka (kao što je `0` ili `4`). Ime računala označava računalo na kojem X server radi. Ako nije navedeno, misli se na lokalni stroj. Nastavak je obično `0` — on se mijenja ako je na jednom računalu više prikaza.

Ako ikad naletite na ime prikaza koje završava na `.b`, to je broj zaslona. Prikaz zapravo može imati i više zaslona. No obično postoji samo jedan, s brojem `0`.

Ima i drugih oblika DISPLAY-a, ali ovo je dovoljno za naše potrebe.

Za tehnički radoznale:

- `imeračunala:P.Z` znači zaslon `Z` na prikazu `P` računala `imeračunala`; X server za taj prikaz sluša na TCP portu `6000+P`.
- `računalo/unix:P.Z` znači zaslon `Z` na prikazu `P` računala `računalo`; X server za taj prikaz sluša na UNIX domain socketu `/tmp/.X11-unix/XP` (pa je dostupan samo s računala `računalo`).
- `:P.Z` je isto što i `računalo/unix:P.Z`, gdje je `računalo` ime lokalnog računala.

5 Objašnjavanje klijentu

Program klijent (vaša grafička aplikacija, na primjer) saznaje na koji će se prikaz spojiti preko DISPLAY varijable okružja. Nju možete nadjačati davanjem argumenta `-display računalo:0` pri pokretanju programa. Malo primjera razjasnit će stvari.

Jedno računalo poznato je kao `svemir`, a mi smo u domeni `svjet.la`. Ako radimo na normalnom X serveru, prikaz se zove `svemir.svjet.la:0`. Želimo pokrenuti `xfig`, program za crtanje, na udaljenom računalu, koje se zove `crna.ru.pa`, i prikazati ga ovdje, na `svemir`-u.

Ako na udaljenom računalu radite u cshu:

```
crna% setenv DISPLAY svemir.svjet.la:0
crna% xfig &
```

Ili:

```
crna% xfig -display svemir.svjet.la:0 &
```

Ako na udaljenom računalu radite u shu:

```
crna$ DISPLAY=svemir.svjet.la:0
crna$ export DISPLAY
crna$ xfig &
```

Ili:

```
dark$ DISPLAY=svemir.svjet.la:0 xfig &
```

Ili, naravno:

```
dark$ xfig -display svemir.svjet.la:0 &
```

Izgleda da neke verzije telneta varijablu DISPLAY automatski prenose na udaljeno računalo. Ako imate jedan od njih, imate sreće i ne morate ništa raditi. Ako nemate, većina verzija telneta prenose TERM varijablu okružja; uz malo petljanja moguće je priljepiti DISPLAY na varijablu TERM.

Zamisao priljepljivanja je da pomoću neke skripte postignete slijedeće: prije telneta verijabli TERM dodajte vrijednost varijable DISPLAY. Zatim pokrenite telnet. Na drugom kraju veze, u odgovarajućoj `.*shrc` datoteci pročitajte vrijednost DISPLAY iz TERM.

6 Objašnjavanje serveru

Server ne prihvaća veze od bilo kamo. Ne želite da svi prikazuju prozore na vašem zaslonu. Ili čitaju ono što tipkate — nemojte zaboraviti da je tipkovnica dio prikaza!

Čini se da premalo ljudi shvaća da je dopuštanje pristupa prikazu sigurnosni rizik. Netko tko može pristupiti vašem prikazu može čitati i mijenjati vaše ekrane, čitati vaše tipke i pratiti kretanje miša.

Većina servera poznaje dva načina autorizacije veza: mehanizam popisa računala (`xhost`) i mehanizam čarobnih riječi (`xauth`). Tu je i ssh, sigurna ljuska, koji može prenositi X veze.

6.1 xhost

`xhost` dopušta pristup na osnovu imena računala. Server ima popis računala koja se na njega smiju spojiti. Provjeru računala može i potpuno isključiti. Pažnja: to znači da provjera nema, pa se može spojiti **bilo koje** računalo!

Serverov popis računala kontrolirate programom `xhost`. U prethodnom primjeru učinite ovo:

```
svemir$ xhost +crna.ru.pa
```

Ovo dopušta sve veze s računala `crna.ru.pa`. Čim se vaš X klijent spoji i prikaže prozor, radi sigurnosti onemogućite daljnja povezivanja pomoću:

```
svemir$ xhost -crna.ru.pa
```

Provjeru računala možete ugasiti s:

```
svemir$ xhost +
```

Tako se isključuje provjera pristupa što omogućuje **svima** da se spoje. Ovo **nikad** nemojte napraviti na mreži na kojoj ne vjerujete **svim** korisnicima (kao što je Internet). Ponovo uključite provjeru računala pomoću:

```
svemir$ xhost -
```

`xhost -` sam po sebi **neće** izbrisati sva računala iz popisa pristupa (to bi bilo prilično beskorisno — ne biste se mogli spojiti od bilo gdje, čak ni sa svog lokalnog računala).

xhost je vrlo nesiguran mehanizam. Ne razlikuje pojedine korisnike na udaljenom računalu. Također, imena računala (zapravo adrese) mogu se krivotvoriti. To je loše ako ste na neprovjerenoj mreži (na primjer već i PPP pristupom na Internet).

6.2 xauth

`xauth` omogućava pristup svakome tko zna pravu tajnu. Takva tajna zove se autorizacijski zapis, odnosno čarobni kolačić. Taj način autorizacije formalno se zove MIT-MAGIC-COOKIE-1.

Kolačići za razne prikaze spremljeni su zajedno u `~/.Xauthority/`. Kolačićima se bavi program `xauth`, odakle dolazi i nadimak ove sheme. Vašem `~/.Xauthority` ne smiju moći pristupiti drugi korisnici.

Na početku rada server će pročitati kolačić iz datoteke navedene uz argument `-auth`. Nakon toga, server dopušta veze samo onim klijentima koji znaju isti kolačić. Kada se kolačić u `~/.Xauthority` promijeni, **server neće prihvatiti promjenu**.

Noviji serveri kolačiće mogu generirati u hodu za klijente koji to zatraže. No kolačići su još uvijek pohranjeni u serveru; ne završavaju u `~/.Xauthority`, osim ako ih klijent tamo stavi. Prema Davidu Wigginsu:

> U X11R6.3 dodana je još jedna mogućnost koja bi vas mogla zanimati. Preko novog proširenja SECURITY sam X server može generirati i vraćati kolačiće u hodu. Također, kolačići se mogu odrediti "nepovjerljivima" tako da su aplikacije koje se spajaju s takvim kolačićima ograničene u svom djelovanju. Na primjer, neće moći krasti ulaz s tipkovnice/miša ili sadržaj prozora od drugih, povjerljivih klijenata. `xauth`u je dodana nova podnaredba `generate` kako bi ta mogućnost bila barem iskoristiva, ako ne jednostavna.

`xauth` ima jasnu sigurnosnu prednost nad `xhost`om. Možete ograničiti pristup na određene korisnike na određenim računalima. Nije ranjiv na krivotvorene adrese kao `xhost`. A ako želite, možete ga koristiti zajedno s `xhost`om za dopuštanje veza.

6.2.1 Pripremanje kolačića

Ako želite koristiti `xauth`, X server morate pokrenuti s argumentom `-auth datoteka`. Ako X server pokrećete skriptom `startx`, tamo je pravo mjesto za to. Ovako pripremite svoj autorizacijski zapis u svojoj `startx` skripti.

Izvadak iz `/usr/X11R6/bin/startx`:

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

mcookie je programčić iz paketa util-linux koji se distribuira s <ftp://ftp.math.uio.no/pub/linux/>. Možete koristiti i `md5sum` za pretvaranje nekih slučajnih podataka (na primjer, iz `/dev/urandom` ili `ps -axl`) u format kolačića:

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

Ako ne možete promijeniti `startx` skriptu (jer niste root), neka to učini vaš administrator, ili neka postavi `xdm`. Ako neće ili ne želi, možete napraviti `~/.xserverrc` skriptu. Ako ona postoji, umjesto pravog X servera pokreće ju xinit. Tako iz te skripte možete pokretati pravi X server s potrebnim argumentima. Da bi to radilo, stavite gornji primjer u svoj `~/.xserverrc` za stvaranje kolačića, a zatim izvršite pravi X server:

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

Ako za rad sa X-om koristite `xdm`, `xauth` možete lako koristiti. Definirajte DisplayManager.authDir u `/etc/X11/xdm/xdm-config`. `xdm` će argument `-auth` prenijeti X serveru kada ga bude pokretao. Kada se logirate preko `xdma`, on za vas kolačić stavlja u vaš `~/.Xauthority`. Za više informacija pogledajte man stranicu *xdm*(1). Na primjer, moj `/etc/X11/xdm/xdm-config` sadrži ovaj red:

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

6.2.2 Prenošenje kolačića

Sad, kad ste počeli rad u X-u na poslužitelju `svemir.svjet.la` i svoj kolačić imate u `~/.Xauthority`, trebate prenijeti kolačić klijentu `crna.ru.pa`.

To je najjednostavnije ako se vaš home direktorij dijeli između `crna` i `svemir`. `~/.Xauthority` datoteke su jednake, pa se kolačić prenosi automatski. Međutim, tu je zamka: kada u `~/.Xauthority` stavite kolačić za `:0`, `crna` će misliti da je to kolačić za njega, a ne za `svemir`. Kod stvaranja kolačića morate koristiti eksplicitno ime računala; ne možete ga izostaviti. Isti kolačić za `:0` i `svemir:0` možete instalirati pomoću:

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

Ako se home direktoriji ne dijele, kolačić možete prenijeti pomoću rsha, udaljene ljuske:

```
svemir$ xauth nlist "${HOST}:0" | rsh crna.ru.pa xauth nmerge -
```

Što će reći:

1. Izvuci kolačić iz lokalnog `~/.Xauthority` (`xauth nlist :0`).
2. Prenesi ga na `crna.ru.pa` (`| rsh crna.ru.pa`).
3. Stavi ga u tamošnji `~/.Xauthority` (`xauth nmerge -`).

Primijetite kako se koristi `${HOST}`. Morate prenijeti kolačić izričito povezan s lokalnim računalom. Udaljena X aplikacija bi mislila da se ime prikaza `:0` odnosi na udaljeno računalo, a to nije ono što želite!

Moguće je da rsh neće raditi. Osim toga, rsh ima i sigurnosnih mana (opet lažna imena računala, ako se dobro sjećam). Ako ne možete ili ne želite koristiti rsh, kolačić možete prenijeti i ručno, na primjer ovako:

```
svemir$ echo $DISPLAY
:0
svemir$ xauth list $DISPLAY
svemir/unix:0 MIT-MAGIC-COOKIE-1 076aaecfd370fd2af6bb9f5550b26926
svemir$ rlogin crna.ru.pa
Password:
crna% setenv DISPLAY svemir.svjet.la:0
crna% xauth add $DISPLAY . 076aaecfd370fd2af6bb9f5550b26926
crna% xfig &
[15332]
crna% logout
svemir$
```

Za više informacija također pogledajte man stranice za *rsh*(1) i *xauth*(1x).

Prijedlog: možda bi bilo moguće priljepiti kolačić TERM ili DISPLAY varijabli prilikom telneta na udaljeno računalo. To bi išlo na isti način kao i priljepljivanje DISPLAY varijable na TERM varijablu. Pogledajte poglavlje 5 (Objašnjavanje klijentu). Tu ćete se morati sami snaći, ali me zanima može li netko ovo potvrditi ili poreći.

6.2.3 Korištenje kolačića

X aplikacija na `crna.ru.pa`, u primjeru `xfig`, automatski će u `~/.Xauthority` potražiti kolačić i njime se autentificirati.

Postoji mala komplikacija kod korištenja `localhost:P`. X klijent aplikacije mogu kod traženja kolačića prevesti `localhost:P` u `računalo/unix:P`. Konkretno, to znači da kolačić za `localhost:P` u vašem `~/.Xauthority` nema **nikakvog** učinka.

6.3 ssh

Zapisi o autoritetu prenose se bez enkripcije. Ako ste zabrinuti da bi netko mogao prisluškivati vaše veze, koristite ssh, sigurnu ljusku. Proslijeđivat će X preko enkriptiranih veza. Osim toga, odlična je i za druge stvari. Dobra je strukturalna nadogradnja vašeg sustava. Samo posjetite `<http://www.cs.hut.fi/ssh/>`, ssh-ovu WWW stranicu.

Tko još zna nešto o autorizacijskim metodama ili enkriptiranju X veza? Možda Kerberos?

7 X aplikacije pod drugom korisničkom oznakom

Recimo da želite pokrenuti grafički konfiguracijski alat koji zahtijeva root privilegije. Međutim, vaša X sesija radi pod vašim običnim računom. Možda će vam se isprva činiti čudno, ali X server **neće** alatu dozvoliti da pristupa vašem prikazu. Kako je to moguće kada root obično može učiniti sve? I kako možete zaobići ovaj problem?

Poopćimo situaciju: želite pokrenuti X aplikaciju pod korisničkom oznakom `korisnikklijenta`, a X sesiju je pokrenuo `korisnikservera`. Ako ste pročitali poglavlje o kolačićima, jasno vam je zašto `korisnikklijenta` ne može pristupiti vašem prikazu: `~korisnikklijenta/.Xauthority` ne sadrži pravi čarobni kolačić za pristupanje prikazu. Pravi kolačić se nalazi u `~korisnikservera/.Xauthority`.

7.1 Različiti korisnici na istom računalu

Naravno, sve što radi za X na daljinu također radi i za različite korisnike (pogotovo `slogin localhost -l korisnikklijenta`). Jedina je razlika u tome što su računalo klijenta i računalo servera jedno te isto. I budući da se radi o istom računalu, postoje neke prečice u prenošenju čarobnog kolačića.

Pretpostavit ćemo da za prebacivanje između korisnika koristite `su`. U biti, ono što morate učiniti je napisati skriptu koja će zvati `su`, okružujući naredbu koju `su` izvršava s kodom koji obavlja stvari potrebne za X na daljinu. Te potrebne stvari su postavljanje varijable DISPLAY i prenošenje čarobnog kolačića.

Postavljanje varijable DISPLAY je relativno jednostavno; dovoljno je navesti `DISPLAY="$DISPLAY"` prije pokretanja naredbe. Dakle, mogli biste jednostavno učiniti ovo:

```
su - korisnikklijenta -c "env DISPLAY=$DISPLAY programklijent &"
```

To još uvijek ne radi jer nismo prenijeli i kolačić. Kolačić možemo saznati pomoću `xauth list "$DISPLAY"`. Ta naredba kolačić ispisuje u formatu koji `xauth` prihvaća kao argument; točno ono što nam treba! Tako ćemo dobiveni kolačić zadati naredbi `xauth` unutar `su` naredbe, tamo postaviti DISPLAY i pokrenuti naredbu koja nam je potrebna.

```
su - korisnikklijenta -c "xauth add ‘xauth list $DISPLAY‘; \
                          exec env DISPLAY=$DISPLAY programklijent"
```

Oko ovoga možete napisati skriptu koja će `korisnikklijenta` i `programklijent` prihvaćati kao parametre. Kad smo već kod toga, skriptu ćemo malo poboljšati, čineći je manje čitljivom, ali robusnijom. Ona izgleda ovako:

```
#!/bin/sh
if [ $# -lt 2 ]
then echo "Korištenje: ‘basename $0‘ korisnikklijenta naredba" >&2
     exit 2
fi
KORISNIKKLIJENTA="$1"; shift
exec su - "$KORISNIKKLIJENTA" \
        -c "xauth add ‘xauth list \"$DISPLAY\"‘; \
            exec env DISPLAY='$DISPLAY' "'"$SHELL"'" -c '$*'"
```

Mislim da je to portabilno i da radi dobro u većini slučajeva. Jedini nedostatak kojeg se sada mogu sjetiti je da će zbog korištenja `'$*'` jednostruki navodnici u argumentu `naredba` zeznuti navođenje u argumentima naredbe. Ako je nešto ozbiljno krivo, javite mi.

Tu skriptu nazovite `/usr/local/bin/xsu`, pa možete raditi ovo:

```
xsu korisnikklijenta 'naredba &'
```

Lako, zar ne?

7.2 Korisnik klijenta je root

Očito, sve što radi za ne-root korisnike radit će i za roota. Međutim, za roota su stvari još jednostavnije jer root može čitati bilo čiji `~/.Xauthority`. Nema potrebe za prenošenjem kolačića. Sve što trebate je postaviti DISPLAY, te postaviti XAUTHORITY na `~korisnikservera/.Xauthority`. Dakle:

```
su - -c "exec env DISPLAY='$DISPLAY' \
                  XAUTHORITY='${XAUTHORITY-$HOME/.Xauthority}' \
                  naredba"
```

Skripta koja to radi izgledala bi otprilike ovako:

```
#!/bin/sh
if [ $# -lt 1 ]
then echo "Korištenje: `basename $0` naredba" >&2
     exit 2
fi
su - -c "exec env DISPLAY='$DISPLAY' \
                  XAUTHORITY='${XAUTHORITY-$HOME/.Xauthority}' \
                  "'"$SHELL"'" -c '$*'"
```

Tu skriptu nazovite `/usr/local/bin/xroot`, pa možete raditi ovo:

```
xroot 'control-panel &'
```

Još lakše, zar ne?

8 Pokretanje window managera na daljinu

Window manager (kao što je `twm`, `wmaker`, ili `fvwm95`) je aplikacija baš kao i svaka druga. Trebao bi raditi uobičajeni postupak.

Pa, skoro. U bilo koje vrijeme na jednom prikazu može raditi najviše jedan window manager. Ako već radite u lokalnom window manageru, ne možete pokrenuti udaljeni (požalit će se i završiti). Najprije morate ubiti (ili jednostavno ugasiti) lokalni window manager.

Nažalost, mnoge skripte za X sesiju završavaju s

```
exec izabrani-window-manager
```

što znači da vaša sesija završava čim (lokalni) window manager završi, nakon čega ju X sustav smatra gotovom i odlogira vas.

Morate prijeći nekoliko dodatnih zapreka, ali to nije nemoguće i nije preteško. Jednostavno se igrajte svojom skriptom sesije (obično `~/.xsession` ili `~/.xinitrc`) kako biste ju natjerali da radi ono što želite.

Budite svjesni da window manager često nudi načine pokretanja novih programa i da će oni biti pokrenuti na lokalnom stroju. To jest, na onom stroju na kojem radi window manager. Ako radite na udaljenom window manageru, pokretat će udaljene aplikacije, a to možda nije ono što želite. Naravno, one će se i dalje prikazivati na vašem lokalnom prikazu.

9 Problemi

Kada prvi put pokrenete udaljenu X aplikaciju, obično ne radi. Evo nekoliko čestih grešaka, njihovih vjerojatnih uzroka, te rješenja koja bi vam mogla pomoći.

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

U okružju nema DISPLAY varijable, a niste ga naveli aplikaciji ni preko argumenta -display. Aplikacija pretpostavlja prazan niz, ali to je sintaktički nepravilno. Da bi ovo rješili, provjerite da je varijabla DISPLAY točno postavljena (pomoću `setenv` ili `export`, ovisno o vašoj ljusci).

```
_X11TransSocketINETConnect: Can't connect: errno = 101
xterm Xt error: Can't open display: crna.ru.pa:0
```

Broj greške 101 znači "Mreža je nedostupna". Aplikacija nije mogla uspostaviti mrežnu vezu sa serverom. Provjerite da li je DISPLAY točan i da se do računala servera može doći s klijenta (trebalo bi se moći, ipak ste vjerojatno logirani na server i telnetom pristupate klijentu).

```
_X11TransSocketINETConnect: Can't connect: errno = 111
xterm Xt error: Can't open display: crna.ru.pa:0
```

Broj greške 111 znači "Veza odbijena". Spajanje na računalo server je uspjelo, ali tamo navedeni server ne postoji. Provjerite da li koristite pravo ime i broj prikaza.

```
Xlib: connection to ":0.0" refused by server
Xlib: Client is not authorized to connect to Server
xterm Xt error: Can't open display: crna.ru.pa:0.0
```

Klijent se povezao sa serverom, ali server mu ne dopušta korištenje (nije autoriziran). Provjerite da li ste klijentu prenijeli pravi čarobni kolačić i da on nije istekao (server uzima novi kolačić kod svakog pokretanja).

10 Hrvatski prijevod

Najnovija verzija ovog prijevoda može se naći na `<http://dokumentacija.linux.hr/X-na-daljinu.html>`. Održava ga Matej Vela, *mvela@public.srce.hr*. Svi su prilozi, primjedbe i prijedlozi dobrodošli.