

COMPUTER SECURITY

6 November 2001

Stephen Tether

Guiding principles

Protect your accounts and Kerberos principal

Don't break the rules

Report break ins

How to protect your account passwords (general)

Choose ones that are hard to guess

- Avoid dictionary words in any language.
- Use at least eight characters.
- Don't use birthdays, names of friends, Social Security number, or other personal information.

Don't use the same password for everything

Change passwords once in a while

- Every six months to a year.

Don't leave them around for people to find

- On paper.
- On disk (**scripts, screen lockers**).

Avoid transmitting them over the network

- Use a secure connection when you do.

Kerberos passwords

It is against Fermilab policy to regularly send your password over the network, even over secure connections

- Use kinit only on the system attached to your keyboard (the local system).
- "kinit -f" to get a forwardable ticket.
- "klist -f" to check (flag "F")
- "-F" (*capital F*) option for Kerberos versions of telnet, rlogin, and rsh.
- Have the following in /etc/ssh_config or your personal .ssh/config
 - KerberosAuthentication yes
 - KerberosTgtPassing yes

Don't use the same password for anything else

- E.g., screen locking: the locker has to have a record of the password.

A password must have at least ten characters and use at least two classes of character

- Classes: lower case, upper case, digits, English punctuation, other

Getting a secure connection

You must encrypt each hop

- Assume any step in "clear" will be sniffed.

Best: use ssh all the way through

- Strong encryption.
- Use X forwarding feature of ssh to encrypt X events.

Second best: "-x" option of kerberized telnet, rlogin, rsh, and rcp (lower-case "x")

- Weaker encryption than ssh.
- No X forwarding.

Avoid X terminals

- Most don't have ssh or Kerberos.

Use computers you can trust

- Not your buddy's laptop or home PC.
- Not flybynight.com.
- Official machines with competent (presumably honest) administrators and some degree of physical security.

Keep your back door locked

Take care with `.rhosts/.shosts`

- *Never* use "+".
- Spell out full host names with domains.
- Don't make it world or group readable.

No "xhosts +"

- Allows anyone to see what you type by monitoring X events.

No world-writable files or directories

Don't open executable e-mail attachments

Obey the rules

Don't try to

- Read other people's e-mail.
- Guess their passwords.
- Use attack software or even have it on any FNAL computer.

Use FNAL computers and networks for official business only

- No personal files.
- Do on-line shopping, etc., from home using a *commercial* ISP (**not** TACACS).
- No visits to sites promoting:
 - Computer cracking.
 - Terrorism (explosives, sabotage).
 - Financial fraud or other crimes.
 - Gambling or pornography.

The FNAL network is monitored, so breaking the above rules may get you an interview with the FBI. It's already happened.

A FNAL computer is any permanently attached to a FNAL network or paid for by the lab.

Authentication vs. encryption

Authentication checks who you are and what rights you have.

- Logging in
- Connecting to an X server
- Accessing files or device
- Reading your e-mail
- Signing e-mail from you
- Receiving network packets

Encryption hides data from those who have no right to see it.

- Passwords
- Encryption keys
- X11 magic cookies
- Credit card numbers
- Your opinion of your boss

Well-known services

Providers of authentication

- login, rlogin, rsh, telnet, etc.
- ssh
- X windows (xhost, xauth)
- Kerberos
- PGP

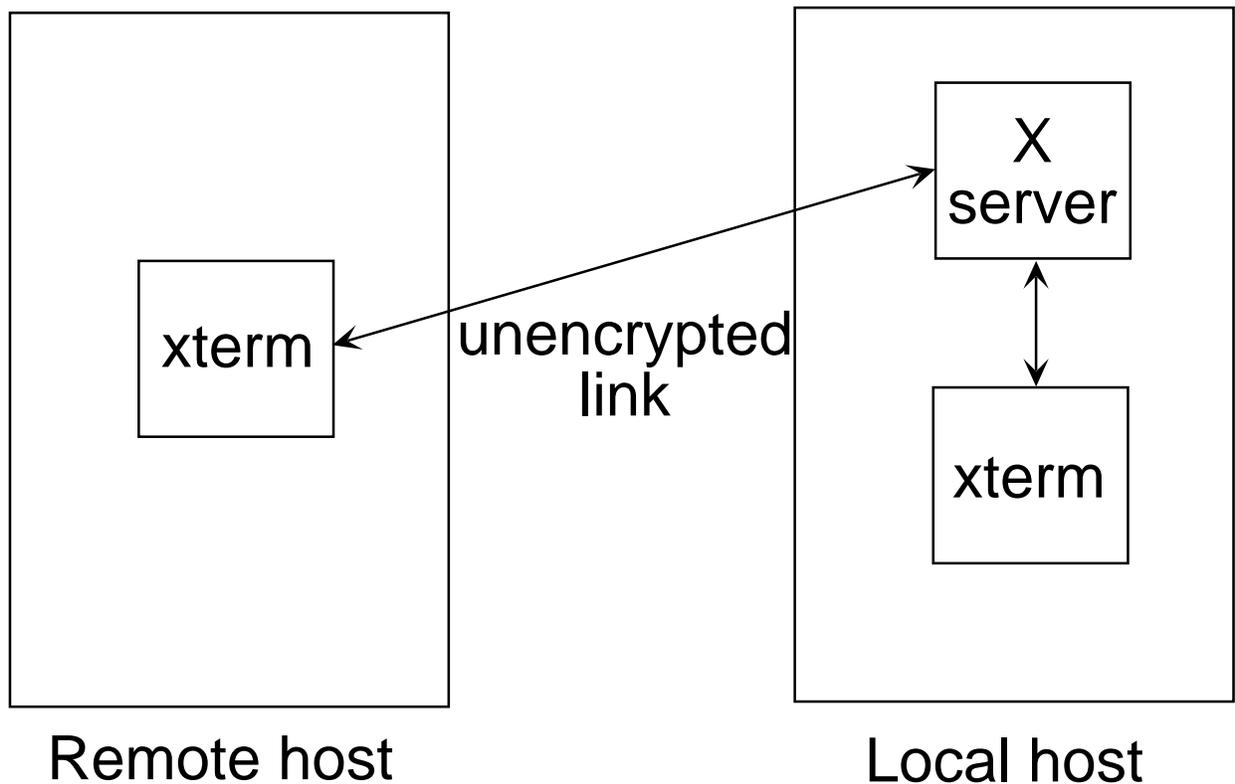
Providers of encryption to hide passwords, verify senders, etc., over the network

- ssh
- Kerberos
- PGP
- **Kerberized** rlogin, rsh, telnet
- **NOT** X windows

Providers of encryption to hide other data.

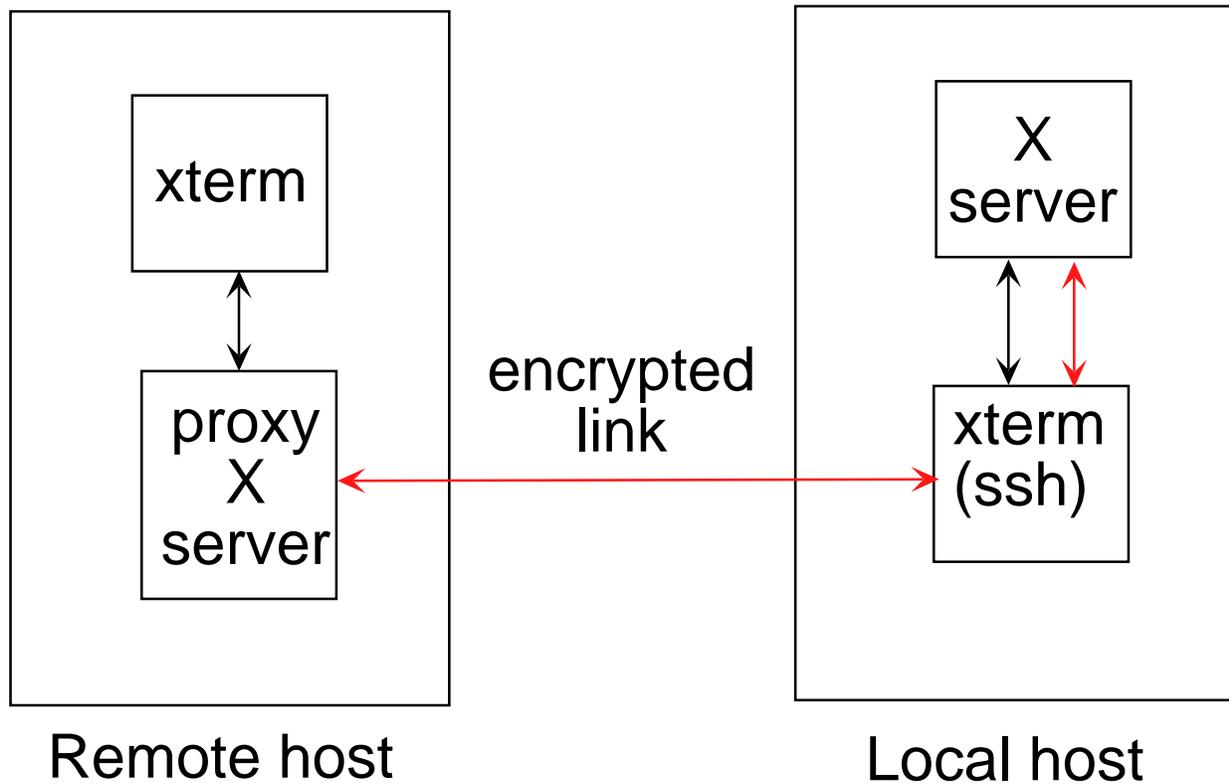
- ssh (entire telnet session)
- PGP (body of e-mail)
- **NOT** Kerberos by itself.
- **FNAL kerberized** rlogin, rsh, telnet ("-x")
- **NOT** X windows

Remote X client via rsh, telnet



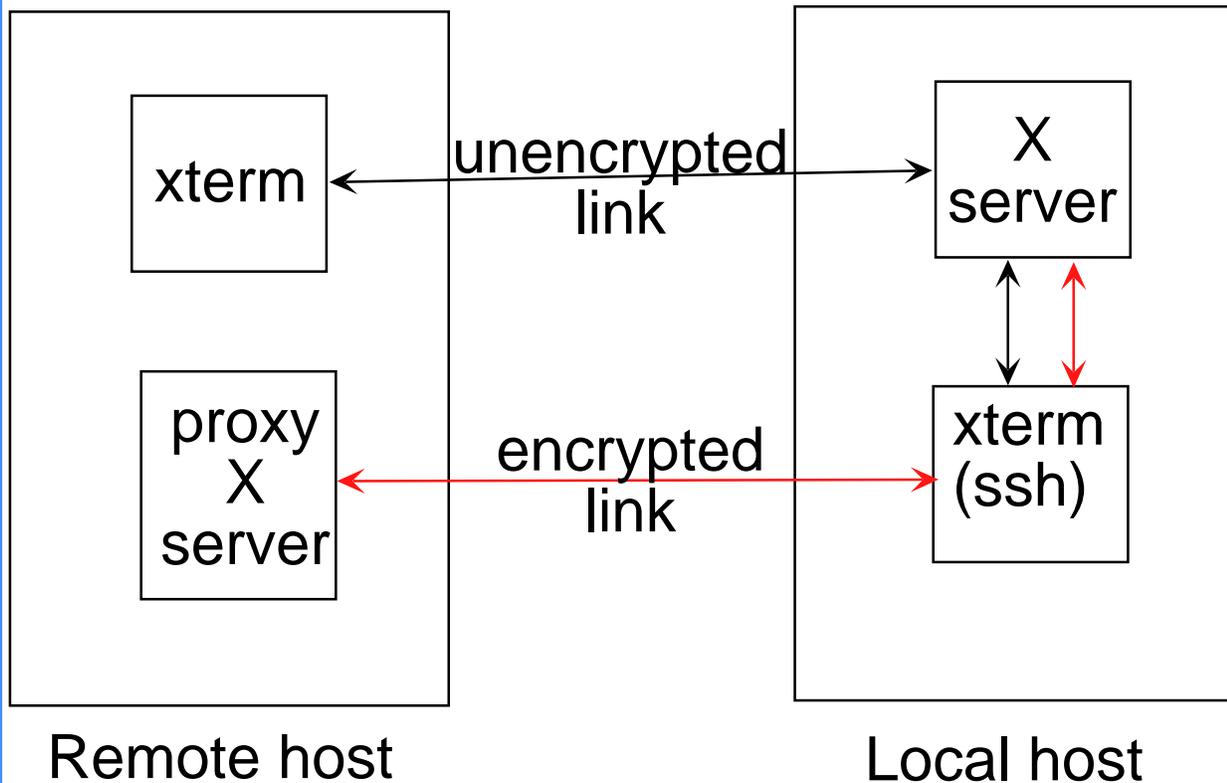
You set DISPLAY after logging in to the remote host and then launch the remote X client. This gives you a direct, insecure network connection back to the local X server.

Remote X client via ssh: X forwarding



After you are logged in ssh creates a proxy X server and sets DISPLAY to point to it. The network traffic between hosts is encrypted. Notice that you can't exit ssh locally without first killing the remote X clients. Do not change DISPLAY or you will get •••

Remote X client via ssh: DISPLAY altered



If you change DISPLAY to point back to the local X server, any remote clients you launch thereafter will establish insecure links. You may want to do this for speed, in which case never type sensitive info such as passwords in remote client windows.

X windows host-based authentication

Problem: how to tell who can use the X display server on your local machine. Most of the time you just want to run remote X clients for yourself.

Solution 1: host-based authentication

- You use "xhost *hostname*" locally.
- Lets anyone on machine *hostname* use your X display.
- *hostname* = "+" lets in anyone, any where.
- *hostname* = "-" disables host-based authentication.

Never use "xhost +"

- Anyone can connect to your display, see your keystrokes, mouse clicks, etc., even without network sniffing.

Use "xhost -"

- xhost doesn't give fine control.
- You can still use xauth (see next page).

X windows cookie-based authentication

Each time you log in locally using X windows, a "magic cookie" (random password) is put in the file `.Xauthority` in your home directory.

`.Xauthority` contains one cookie for each X display you're allowed to use.

Use the `xauth` program to manipulate `.Xauthority`

To give the local cookie to yourself on a remote machine

- Get it from your local `.Xauthority`
- Add it to your remote `.Xauthority`

```
xauth extract – $DISPLAY |  
ssh remote xauth merge –
```

Remember – the cookie for your local display changes every time you log in using the X windows login screen.

SSH authentication

Several different methods are available

- .rhosts/.shosts file (**AVOID**)
- User password + host public RSA key
- User public/private RSA key
- Kerberos5 (SSH1 only)

Choose method in ~/.ssh/config

Except for Kerberos, these methods have no centralized key management.

You collect host public RSA keys in ~/.ssh/known_hosts (semi-automated)

To use user RSA-key method:

- Run ssh-keygen
- Copy ~/.ssh/identity.pub to each node you want to log into, merge into file ~/.ssh/authorized_keys.
- Copy ~/.ssh/identity (private key) to the same file on each node from which you wish run ssh.
- Nothing to stop you from using different secret keys for different nodes.

One-key and two-key encryption

One-key methods

- Same key both encrypts and decrypts.
- Anyone who can send you encrypted messages can read messages others send you.

Two-key methods (e.g., RSA)

- One key you keep private
- The other you let everyone know.
- Messages encrypted with the public key can only be decrypted by the private key (and vice versa)
- Anyone can send you encrypted messages that only you can read
- An encrypted message that one can read with your public key had to have come from you

Two-key methods allow challenges

- You say "I am Joe"
- I say "Prove it! Decode this message encrypted with Joe's public key"

Kerberos authentication

Centralized key management (per "realm")

- Known users (hashed passwords)
- Known servers
- One server (TGS) is special: dispenses "tickets" granting access to other servers.

Logging in

- You tell key manager your name
- It sends you a message you can decode with your secret key (password)
- Message contains key (TGT) used for messages to special server
- Special server can then tell you keys (tickets) you need to talk to other servers.

This description omits other security details such as time stamps on messages and tickets.

Report break ins

Suspicious activities

- Logins you didn't make (use "last")
- Missing or changed file contents or protections
- New files or directories you didn't make
- The system is so busy it can't do anything more
- False messages sent in your name
- Unexpected reboots

Report these to the system administrator

Don't try to correct the problem yourself

Export controls

"Strong" encryption techniques cannot be exported outside the U.S.A.

The law classifies them as weapons!

Consult with U.S.A. system administrator before trying to install encryption software outside the States.