Secure Communications

- Cleartext vs. encryption and encapsulation
- Protocols not to use
  - SSH
    - scp/ftp
    - SSH tunnelling
- VPN
Cleartext vs. Security

• Once upon a time, the Internet was a friendly place

• Most communications (rsh, telnet, ftp) used cleartext authentication, in which usernames and passwords were sent over the network without any effort to hide them

• This is no longer appropriate!
Review: IP Datagram

<table>
<thead>
<tr>
<th>IP headers</th>
<th>Unencrypted data</th>
</tr>
</thead>
</table>

• The header contains info about the type of datagram this is and the source and destination IP addresses.

• The data appears without any encryption whatsoever; a packet sniffer can read your username, password, private data, etc.
Encryption

- The datagram headers are left unchanged, but the data is encrypted.

| IP headers | Encrypted data |

- Example: encrypted telnet (uses Kerberos session key as encryption key)

  kinit username@IASTATE.EDU
telnet -faxl username isua1.iastate.edu
Encapsulation

- The data is hidden, and the port number can be altered to override firewall settings.

  IP headers  Unencrypted data

The packet is encrypted...

  Encrypted datagram

...and encapsulated in a new datagram.

  New IP header  Encrypted datagram
Secure Shell

• Replaces rsh, telnet, ftp
• Provides encrypted connection between host and client
• Provides user authentication based on public/private keys or passwords
• Multiple datastreams can be multiplexed through the SSH connection
SSH connection (3 layers)

- Transport Protocol: host and client connect on port 22 and negotiate encryption and optional compression protocols
- User Authentication Protocol: User is authenticated by
  - Public/private key exchange
  - Password (cleartext but encrypted)
  - Other methods (GSSAPI/Kerberos V)
- Connection Protocol: user channel is multiplexed into several logical data channels
Text-mode SSH

• Replaces rsh, rlogin and telnet

• Use

ssh username@hostname

• If you haven't connected to that host before, you may be asked to accept the host's public key (this can be turned off for greater security)
SCP and SFTP

- To copy files to or from a secure host

- To:
  \[ scp \text{ filename} \text{ username}@\text{host}:filename \]

- From:
  \[ scp \text{ username}@\text{host}:filename \text{ filename} \]

- Secure FTP (sftp) looks like the normal ftp client, but uses ssh as the tunnel:
  \[ sftp \text{ username}@\text{hostname} \]
X over SSH

- X is a cleartext protocol and inherently insecure
- Running X through an SSH tunnel makes it an encrypted connection
- Log in with
  `ssh -X username@hostname`
  then start X applications on the host.
- `ssh -x` disables X tunneling
Tunneling Other Protocols

- Forward port nnn on the client to port mmm on the host:
  \texttt{ssh -L nnn:hostname:mmm user@hostname}

- Forward port mmm on the host to port nnn on the client
  \texttt{ssh -R nnn:hostname:mmm user@hostname}

- For info on using VNC over SSH, see
  \url{http://www.it.iastate.edu/pub/lnt315/lnt315.pdf}
Virtual Private Network (VPN)

- Used to connect a remote machine or LAN to corporate LAN over the Internet
  - Eliminates expensive dedicated lines
  - Maintains security via encryption
  - Allows remote machines to appear as if they have a "campus" IP address
Connecting a Remote Machine to a LAN

The VPN provides an encrypted tunnel through the insecure Internet.
VPN Hardware

• Server side:
  - Almost always dedicated hardware (could use a Linux machine, but quickly becomes performance limited)

• Client side:
  - For single remote machine, use software on client machine
  - For small to large LAN, use another dedicated router
VPN Implementations

- PPP over SSH
  - Connects PPP client to PPP server through an SSH tunnel instead of a dialup connection
  - Simple to implement, but subject to serious timing problems
VPN Implementations

• Point-to-Point Tunneling Protocol (PPTP)
  - Microsoft implementation (Windows NT)
  - Uses PPP through a data channel (which *may be* encrypted and/or compressed) and a separate control channel
  - Not considered very secure
  - See http://puptpclient.sourceforge.net/ and http://www.poptop.org/
VPN Implementations

• IPsec
  - one channel for authentication and exchanging encryption keys, plus one or more data channels; two protocols at the IP level plus one at a higher level
  - Used by the ISU VPN server and newer versions of Windows
  - See http://tldp.org/HOWTO/VPN-Masquerade-HOWTO.html
VPN Implementations

• CIPE
  - designed specifically for building encrypting routers
  - tunnels encrypted IP packets in UDP packets, so can forward all types of TCP/IP packets
VPN@ISU

- VPN servers at ISU are administered by ITS
- Software is available from http://www.sitelicensed.iastate.edu
- To install in Linux, you must have the kernel source installed (package kernel-source)