COSC 301
Network Management

Lecture 16: Remote Terminal Services

Zhiyi Huang

Computer Science, University of Otago

Today's Focus

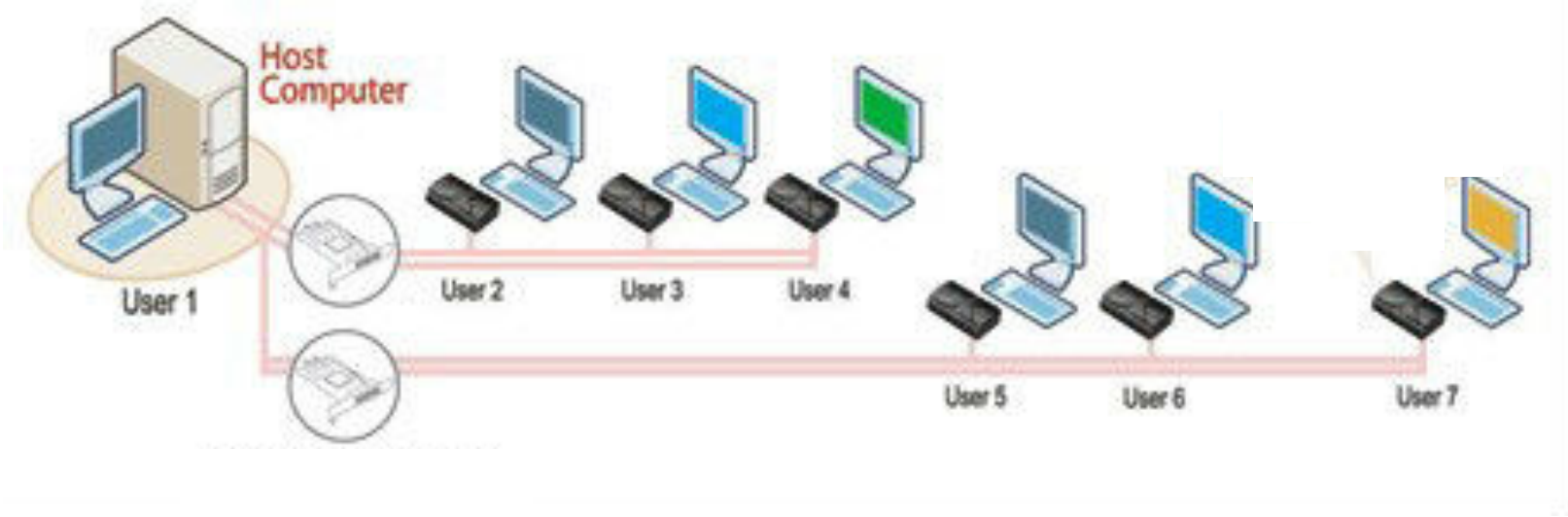
Remote Terminal Services

-- What is a remote terminal?

-- SSH

What is a terminal?

- An electronic device used for entering data into, and displaying data from a computer
  - Dumb terminal (thin client): no local processing ability
  - Smart terminal (fat client): has local processing ability

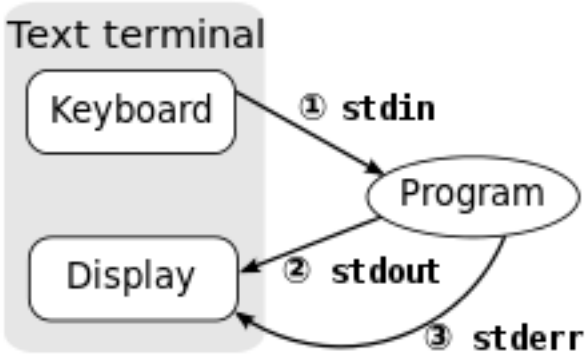


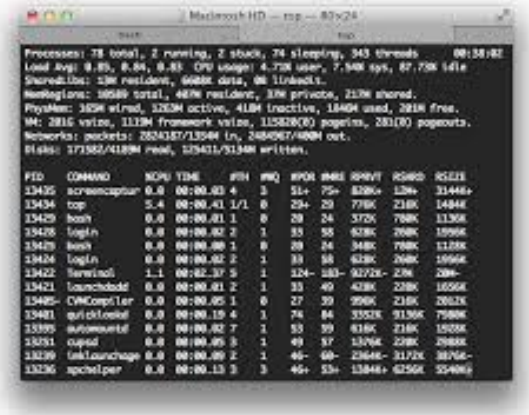
What is a terminal?

- Hard-copy terminals
  - TeleTYpewriter (TTY)

- System console

- Terminal emulator
  - a program that does what a dumb terminal used to do
  - Terminal window





TTY Remote History

- Berkeley ‘r’-commands
  - rsh — remote shell commands
  - rlogin — remote terminal
  - rcp — remote copy

Weak host-based authentication

Privileged ports, .rhosts, no password

- Telnet
  - Remote terminal, similar to rlogin
  - User-based authentication

Past Problems

- Everything sent in clear-text, no encryption
- Weak Host-based authentication
  - Exploitable trust relationships
  - Privileged ports offer little protection
- Server not authenticated
  - Man-in-the-middle (MITM) attack potential

Resolutions

- Encrypt all traffic
- Authenticate both user and server
- Avoid trust relationships

Secure SHell (SSH)

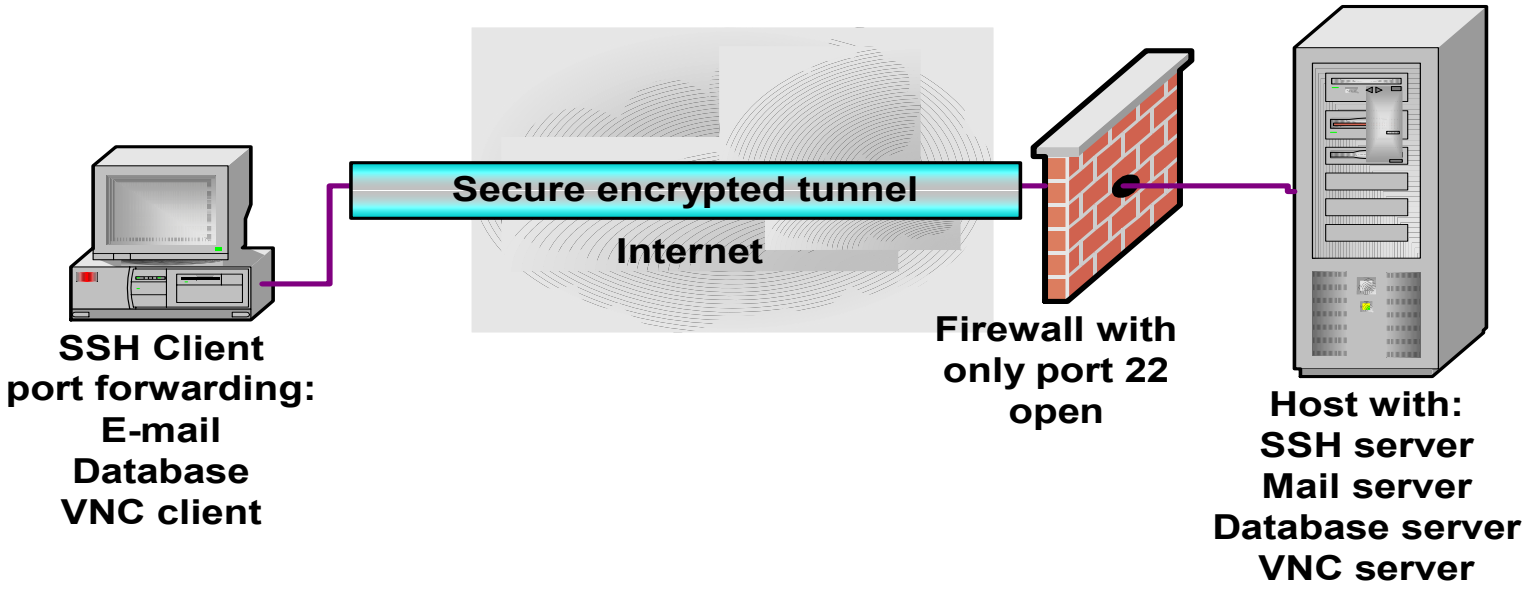
- SSH provides secure versions of the ‘r’-commands and telnet
- Encrypts all traffic
  - Public/Private Key for authentication
  - Fast block cipher for data transfer
- Authenticate both server and user
  - A wide range of means to do so

Functionality of SSH (1)

- Secure Command Shell: anything that can be done at a local machine can be done securely remotely
  - View contents in directories and edit files
  - Start batch jobs,
  - Start, view or stop services and process
  - Create user accounts, change permissions

Functionality of SSH (2)

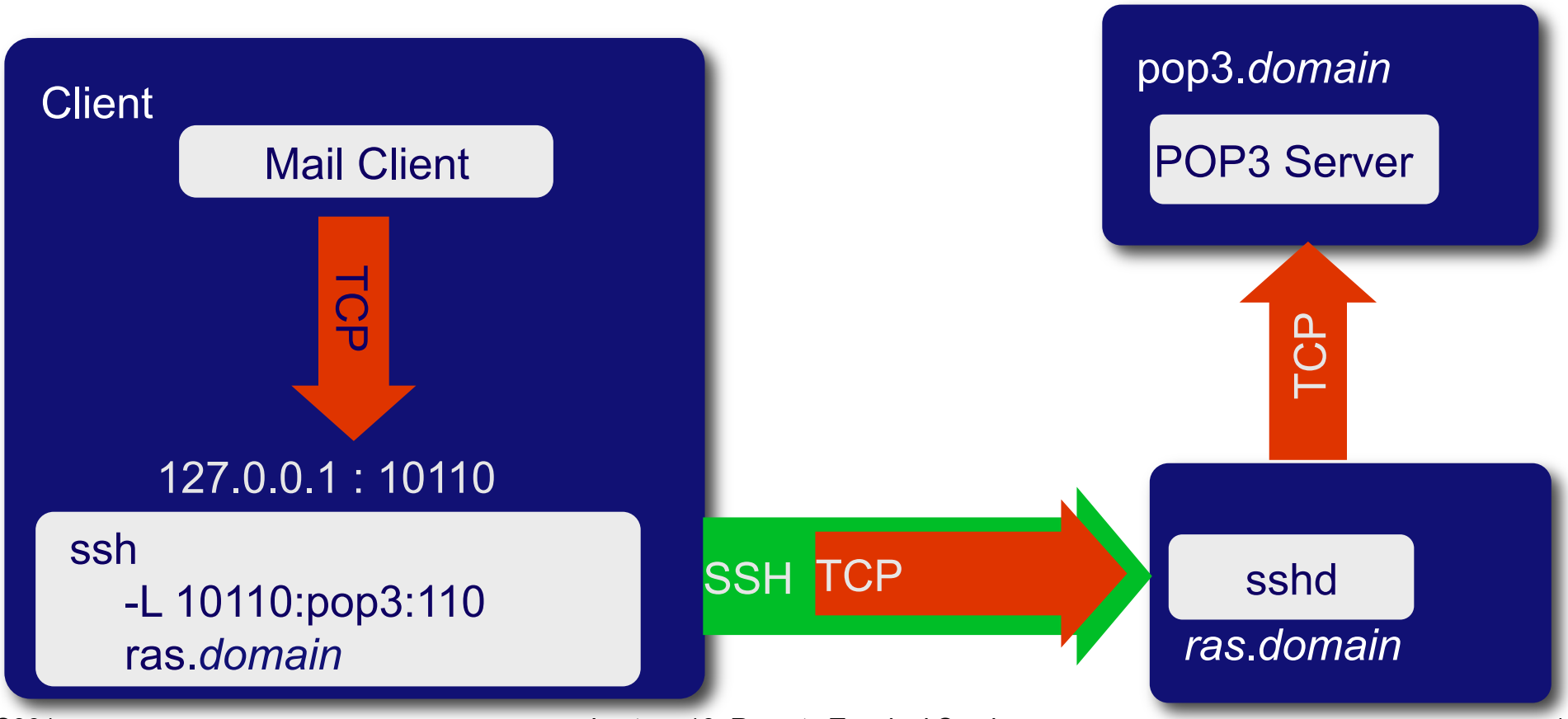
- Port forwarding
  - allows data from normally unsecured TCP/IP applications to be securely sent across the encrypted tunnel,
  - multiple applications can transmit data over a single multiplexed channel.



Ref: An Overview of the Secure Shell (SSH), Vandyke Software

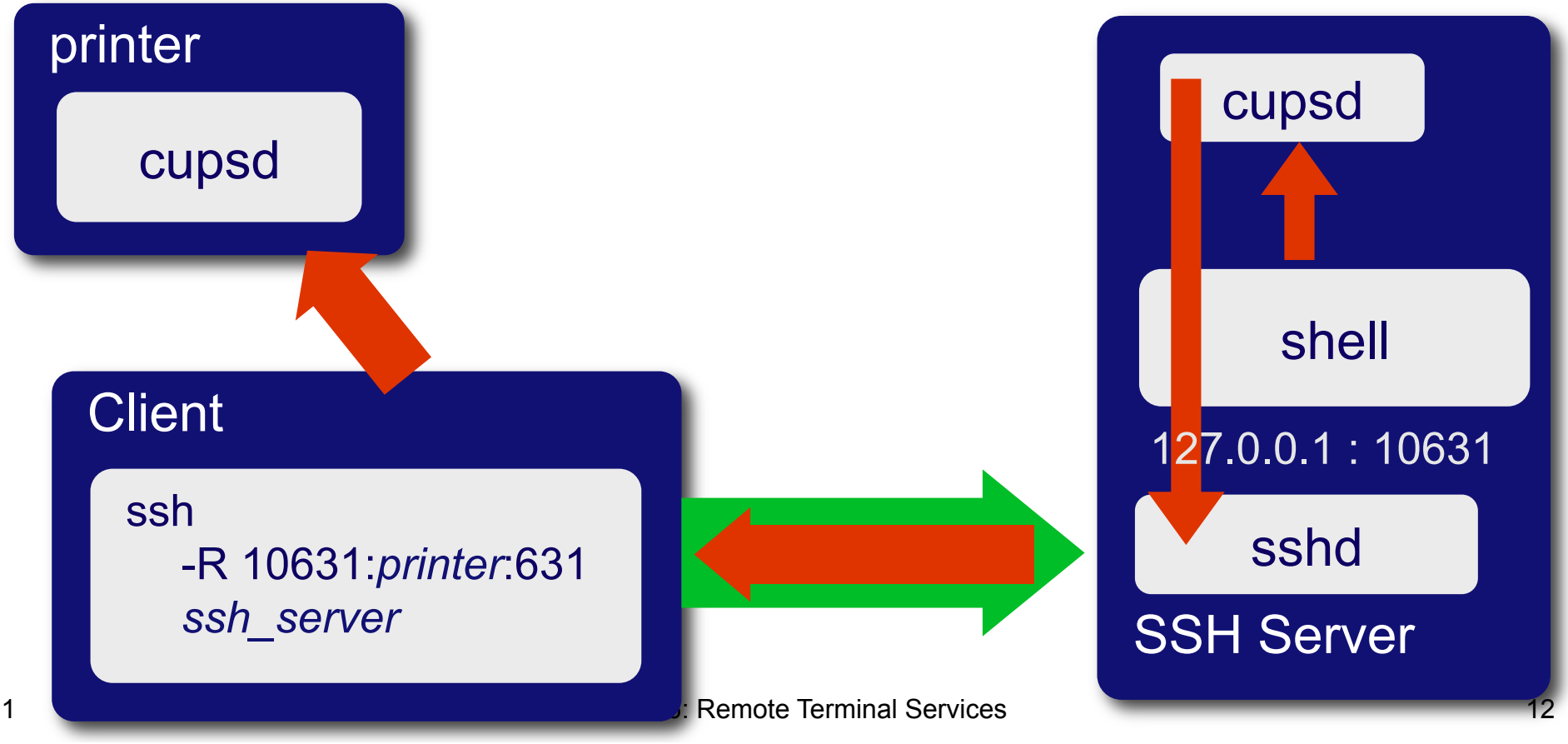
Functionality of SSH (3)

- Port forwarding
  - Local port forwarding: forward data securely from another client application running on the same computer as the Secure Shell Client



Functionality of SSH (4)

- Port forwarding
  - Remote port forwarding: enables applications on the server side of a Secure Shell connection to access services residing on the SSH's client side.

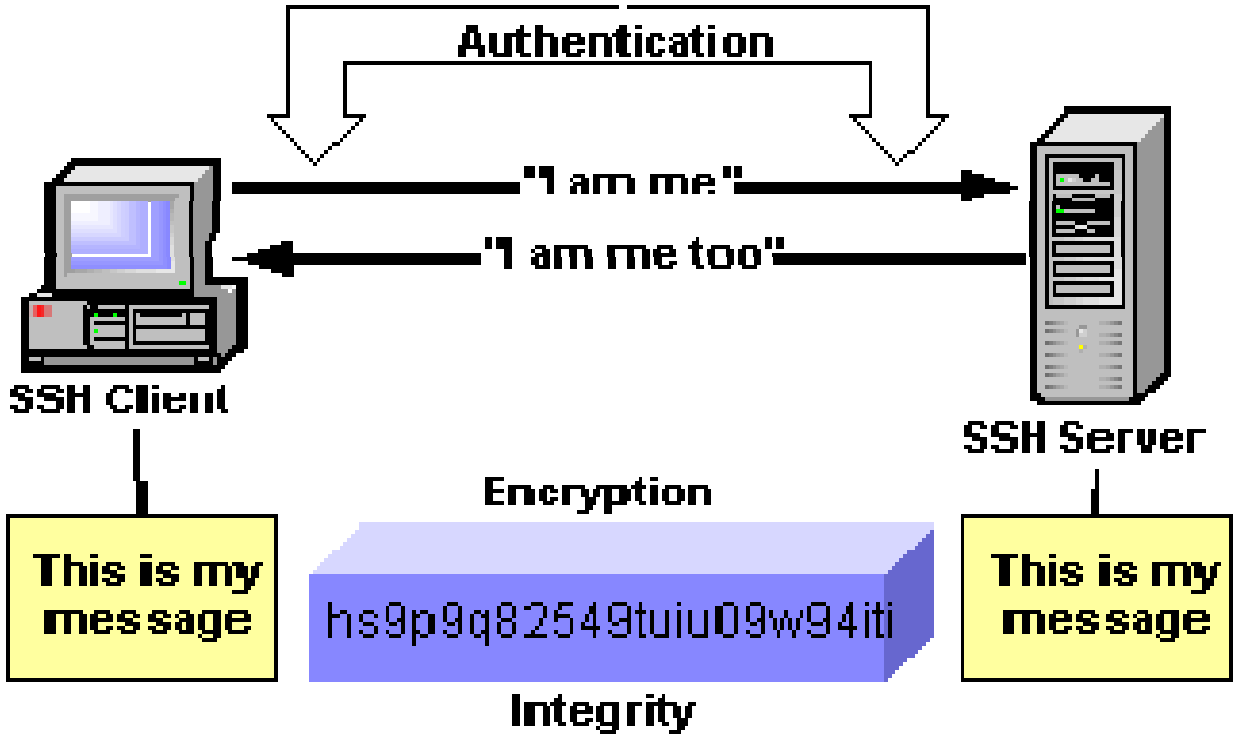


Functionality of SSH (5)

- Secure File Transfer
  - Secure copy (SCP): RCP protocol over SSH.
  - Rsync: intended to be more efficient than SCP
  - SSH File Transfer Protocol (SFTP): a secure alternative to FTP (not to be confused with FTP over SSH)
  - Files transferred over shell protocol (FISH): evolved from Unix shell command over SSH

Basics of SSH

- User Authentication
- Host Authentication
- Data Encryption
- Data Integrity



Keys, Keys, Keys

- User Key
  - A persistent, asymmetric key used by clients as proof of a user's identity.
  - A single user may have multiple keys
- Host Key
  - A persistent, asymmetric key used by a server as proof of its identity
  - Used by a client when proving its host's identity as part of trusted-host authentication
- Server Key
  - A temporary, asymmetric key used in the SSH-1 protocol.
  - It is regenerated by the server at regular intervals (by default every hour) and protects the session key
- Session Key
  - A randomly generated, symmetric key for encrypting the communication between an SSH client and server.

User Authentication

- Password authentication
  - The username and password are encrypted before transmission.
  - Inherently vulnerable in that they can be guessed
- Public key authentication
  - Public key and private key: generated using ssh-keygen
  - Private key should never be distributed, and should be protected by “passphrase”

My public key:

ssh-dss AAAAB3NzaC1kc3MAAACBAOYxKP/MGpQ4WzWZWwMhhkdUqzYlt/MQ9wGqvtjYemsAT6JFhsoQhBxetqRl//
M1CmX6I29M9Xosi9y6n0ch8WYQOYb2pJmoLOs+imn71E52c/BcQ+81QMIKbXDirWYfFBKVO6/laEHzhmI
+84mGMgtEX5yHShJJ1T1Kw03Oq57DAAAAFQDbQ0zs3AuDYJSjNwUY+48hGvnyQwAAAIA7vSwfKmfXp6frnx8UwBD/
6HwmGiMiwMOg8mXxm8iVm8Qg210TIFihNU8b6Y/chWfjsy0iYo1Rczs/
0yMfdLgupYRluYOEbj58+Rg5WNKa0Np7aOuCrftLRrdwOQCiT93EQSObFzBYxlChEG75rgQIBFc65M8cE9ear0oeC8JWqQA
AAIAFEHQHFCUeZsusoWRm7uP5nTB2rnIb2ZpvpdTb/8UqNtIIn0WctqxELWGfCLiKEVpOYsjvCtHHYr/
3tsQ3PloudD479Uke9fn8N5E2rIRQkbQM4yLi0elAV9lgIh6ctJUQqRdmmAZC+xidE6JxMUssvUIOHosRqSc70XSabUYswQ==
haibo@oucs1325.otago.ac.nz

Host Authentication

- Host key
  - Used by a server to prove its identity to a client
  - Used by a client to verify its “known” host
  - Persistent (change infrequently) and asymmetric
  - Guards against the Man-in-the-Middle attack

- Two operations:
  - Looking up the client host key
  - Matching the client host

http://docstore.mik.ua/orelly/networking_2ndEd/ssh/ch03_03.htm

Data Encryption/Integrity

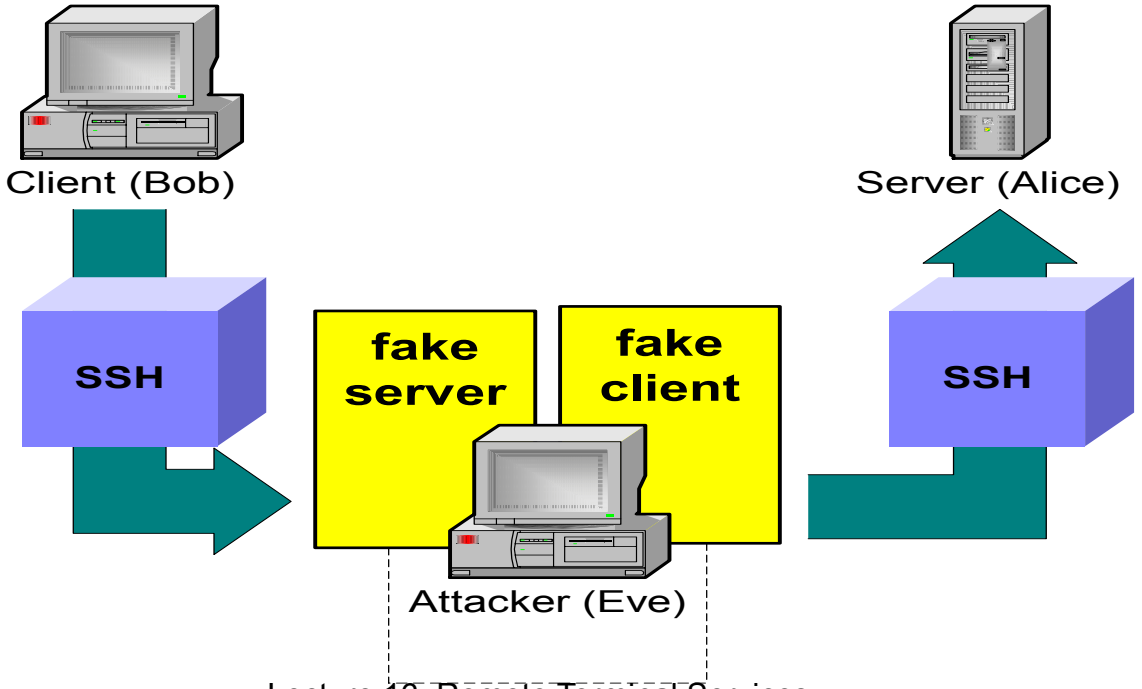
- Encryption
  - Use ciphers to encrypt and decrypt data being send over the wire
  - Block cipher such as DES, 3DES, use a shared key (session key)
  - Agree which cipher use during connection setup
  - Session keys are randomly generated by both the client and server, after host authentication and before user authentication
- Integrity
  - Message Authentication Code (MAC) in SSH2
  - Simple 32-bit CRC in SSH1

SSH versions

- Two major versions
  - SSH1
    - Has inherent design flaws which make it vulnerable
  - SSH2
    - Currently used
    - Support any number of sessions over a single SSH connection
    - Better security: Diffie-Hellman key exchange and strong integrity checking.

Threats Addressed by SSH

- Eavesdropping or Password Sniffing
  - All transmitted data is encrypted
- Man-in-the-middle attack (MITM)
  - Host authentication
  - Can not happen unless the host itself has been compromised



Threats Addressed by SSH

- Insertion and Replay attack
  - Attacker is not only monitoring the SSH session, but is also observing the keystrokes
  - By comparing what is typed with the traffic in the SSH stream, the attacker can deduce the packet containing a particular command, and reply the command at a particularly inappropriate time during the session.
  - Message authentication code prevents such attacks.

SSH Doesn't Prevent

- Password Cracking
- IP and TCP attacks
- Traffic Analysis