ssh – The Secure Shell

David Morgan

A client-server pair of programs

- ssh - client
  - /usr/bin/ssh (configurable)
- sshd - server
  - /usr/sbin/sshd
  - assigned port number 22
- Originated by Tatu Ylonen as a secure drop-in replacement for rsh/rlogin/rcp
Secure Shell’s Functions

- Explicit functionalities
  - Remote login tool
  - Remote command executor
- Implicit activities
  - Authentication
  - Encryption

SSH stated mission

“ssh is a program for **logging in** to a remote machine and for **executing commands** in a remote machine. It is intended to replace rlogin and rsh, and provide secure encrypted communications between two untrusted hosts over an insecure channel.”

ssh man page – first sentence
ssh syntax

Logging in

ssh remote-user-name@remote-machine-id

  e.g.,  ssh student@193.6.37.12

Executing a command

ssh remote-user-name@remote-machine-id command

  e.g.,  ssh student@193.6.37.12 cat /etc/passwd


ssh dynamic encryption

- all session/command traffic passes through ssh/sshd (sshd runs on port 22)
- encrypted going out/decrypted coming in
- for duration of session/command
ssh – why secure?

- uses RSA (public-key) authentication
- then strong-key symmetrical encryption

ssh implementation

for passwordless/unattended authentication

- ssh-keygen utility (local) generates:
  - private/decrypt key in
    /home/<localuser>/ssh/id_dsa
  - public/encrypt/ascii key in
    /home/<localuser>/ssh/id_dsa.pub

LOCALUSER EFFECTS KEY TRANSFER:

/home/<localuser>/ssh  /home/<remoteuser>/ssh
id_dsa
id_dsa.pub

authorized_keys2

Copy ascii
public key
**ssh implementation**

When local machine issues:

```sh
ssh remote-user-name@remote-machine-id
```

Local machine (ssh) sends *local* user’s public key to remote machine (sshd)

Remote machine authenticates if
1) that key appears in `remote-user-name`’s authorized_keys file, if so
2) local machine can decrypt random text encrypted with it by remote machine as a challenge.

**Configuration files**

- `ssh` - `/etc/ssh/ssh_config`
  - cipher selection
  - compression level
  - port forwardings

- `sshd` - `/etc/ssh/sshd_config`
  - authentication options
    - key-based
    - password
    - both
  - logging options
ssh feature: port forwarding

ssh port forwarding:

- correspond some port on the client (e.g., 3000) to some port (e.g., 80) on a machine reachable thru the server…

Example: http://127.0.0.1:3000 in client’s browser gets served from 192.168.1.111

PuTTY

multi-platform GUI front end

http (web) ssh port forwarding:

- platform

Example: http://127.0.0.1:3000 in client’s browser gets served from 192.168.1.111
Don’t leave back door ajar!

“System security is not improved unless rshd, rlogind, rexecd, and rexd are disabled (thus completely disabling rlogin and rsh into that machine).”

sshd man page

Getting ssh

- open source
  http://www.ssh.com/

- commercial
  http://www.openssh.com/
Free windows clients

- PuTTY
  http://www.chiark.greenend.org.uk/~sgtatham/putty/

- cygwin under Windows / openSSH under cygwin
  http://www.cygwin.com/

ssh information

- OpenSSH FAQ
  http://www.openssh.com/faq.html

- “Getting Started with ssh”
  https://kimmo.suominen.com/docs/ssh//