Secure Shell

Omkar Sanjay Nibandhe

Binghamton University

Oniband1@binghamton.edu

INTRODUCTION

Secure Shell (SSH) provides robust, user-friendly encryption and authentication. SSH is reliable, inexpensive security product (OpenSSH, Tectia, PuTTY) for computer networks and people who use them. SSH automatically encrypts the data sent by computer to the network and receiver decrypts the data. The users are unaware of the encryption/decryption and the result is transparent. The basic architecture needed for SSH includes a SSH server, SSH client and keys. The SSH server is usually called SSH daemon (shhd), ssh client and keys will be explained later. SSH protocol (SSH-1, SSH-2) provides authentication, authorization, encryption, forwarding/tunneling to encrypt other TCP/IP based sessions and the integrity of data transmitted over a network. You might have multiple accounts or multiple personal computers, so to exchange files or data or run commands securely over a network SSH is used. Various other methods like ftp, telnet, rsh exists but are unsecure. So secured encrypted tunnel is provided by SSH.

ARCHITECTURE OF SSH

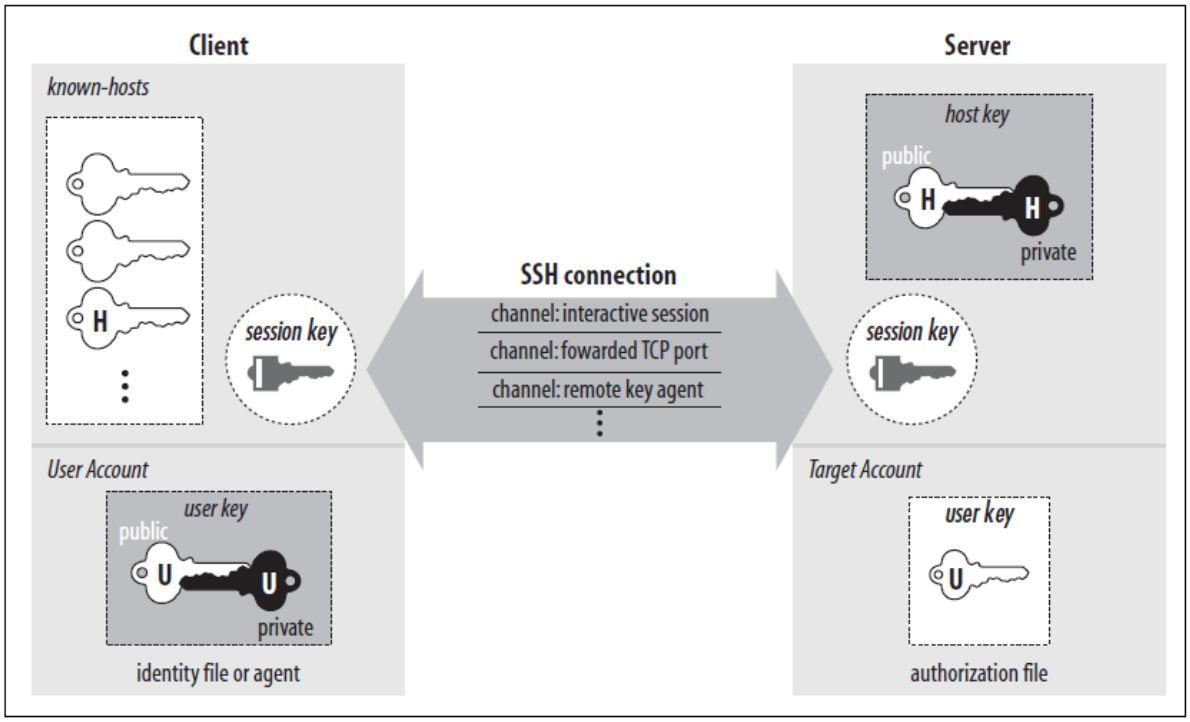

Figure 1 Architecture of SSH

The following are the major components used in the SSH model.

Server(sshd)

Each SSH server has a secret, unique ID, called a *host key* to identify itself to clients. Sshd allows incoming SSH connections to a machine, handling authentication, authorization etc.

Client

ssh, scp and sftp are examples of clients which connect to server for making requests.

Session

Ongoing connection after authorization. Session can be interactive or batch.

Key

A key is a digital identity. A key pair is set of public and private key. The private key needs to be saved with a paraphrase, whereas the public key be saved on the SSH server machine. Ssh-keygen program helps the user to generate a key pair by either DSA or RSA algorithm. The public key is saved by default under the /home/.ssh directory. The identity file is readable only by your account. The passphrase can be changes as and when required. Regeneration of ssh key pair is also possible but the public key has to be manually update to all hosts, so it's an extra overhead after regeneration.

Name	Lifetime	Created by	Type	Purpose	Type
User key	Persistent	User	Public	Identify user to server	Asymmetric
Host key	Persistent	Administrator	Public	Identify server/machine	Asymmetric
Session key	One Session	Client (and Server)	Secret	Protect Communications	Symmetric

Table 1 Types of keys

Key generator

Program that generates the persistent keys for SSH. Example ssh-keygen

Known-hosts database

A collection of host keys. Clients and servers refer to this database to authenticate one another.

Seed

A pool of random data used by SSH components to initialize software pseudorandom

Random number generators.

This protocol binds each session key to the session by including random, session specific data in the hash used to produce session keys. Special care should be taken to ensure that all of the random numbers are of good quality.

Configuration file

A collection of settings to tailor the behavior of an SSH client or server.

The following explains the terms used in SSH.

Known hosts

1st time add the particular remote host to known hosts list. The server authenticates the client by passphrase and the client authenticates the server by its public key. Each time you reconnect to that remote host, the SSH client checks the remote host's identity using this public key. Of course, it's better to have recorded the server's public host key before connecting to it the first time, since otherwise you are technically open to a man-in-the-middle attack that first time.

The Escape Character

By default, the escape character is the tilde (~), but you can change it. That character must be the first character on the command line, i.e., following a newline (Control-J) or return (Control-M) character.

The SSH Agent

A program that keeps private keys in memory and provides authentication services to SSH clients. The effects of the agent last until you terminate the agent, usually just before logging out. To avoid SSH agent or connect even without passphrase we can use one of the following options:

- Public-key authentication with an agent
- Host based authentication
- Kerberos authentication

The SSH protocol consists of three major components: Transport Layer Protocol provides server authentication, confidentiality, and integrity with perfect forward secrecy. The User Authentication Protocol authenticates the client to the server. The Connection Protocol multiplexes the encrypted tunnel into several logical channels.

Host Keys

Server should have a host key. Hosts may have multiple host keys using different algorithms. Multiple hosts may share same host key. The server host key is used during

key exchange to verify that client is talking to correct server. For client to know the server it must know the server's public host key.
Two approaches to save the public host key of server:

1. The client maintains a local database that associates each host name with corresponding public host key. This method has overhead of creating and maintaining the name-to-key association.
2. The host name-to-key association is certified by a trusted certificate authority. The client knows the CA root key and can verify the validity of all host keys certified by accepted CAs. No local database storage needed but a lot of trust is put on the central infrastructure.

Both implementation provide an option to not to accept the host keys that cannot be verified.

Extensibility

DNS names are used to create local namespaces where experimental or classified extensions can be defined without fear of conflicts with other implementations. However, some mandatory algorithms are specified for the relevant protocol document which must be provided to ensure interoperability.

Policy Issues

Each algorithm to share keys will be different based on encryption, integrity and compression algorithms. Thus each newly added algorithm should specify the policy for the preference of encryption, integrity, etc.
Public key algorithms and key exchange method to be used for host authentication. The existence of trusted host keys for different public key algorithms also affects this choice.
The required algorithms may depend on the location from where the user is trying to gain access or require multiple authentications for some users.
Due to security reasons, the user may not be allowed to run all commands upon connection like starting a new session on client side.

SSH FEATURES

Secure Remote Logins, Secure File Transfer, Secure Remote Command Execution, Keys and Agents, Access Control, Port Forwarding, etc.

OS SUPPORT

Most Linux distributions, Macintosh OS X, Sun Solaris, OpenBSD, and virtually all other Unix-inspired operating systems. Microsoft Windows has plenty of SSH clients and servers. SSH can be found on PalmOS, Commodore Amiga, and most other platforms.

SSH PACKETS

SSH packets are numbered from 1 to 255. These are explained layer wise as follows:

Transport layer protocol:

1 to 19	Transport layer generic (e.g., disconnect, ignore, debug, etc.)
20 to 29	Algorithm negotiation
30 to 49	Key exchange method specific (numbers can be reused for different authentication methods)

User authentication protocol:

50 to 59	User authentication generic
60 to 79	User authentication method specific (numbers can be reused for different authentication methods)

Connection protocol:

80 to 89	Connection protocol generic
90 to 127	Channel related messages

Reserved for client protocols:

128 to 191	Reserved

Local extensions:

192 to 255	Local extensions

References

[1] SSH, the Secure Shell The Definitive Guide *Daniel J. Barrett, Richard E. Silverman, and Robert G. Byrnes.*

[2] T. Ylonen, SSH Communications Security Corp and C. Lonvick,Ed., Cisco Systems, Inc. https://www.ietf.org/rfc/rfc4251.txt The Internet Society (2006)

[3] Girish Venkatachalam, The OpenSSH Protocol under the Hood, Apr 01, 2007 http://www.linuxjournal.com/article/9566