SSH

Aravind Kannan Rathinasabapathi

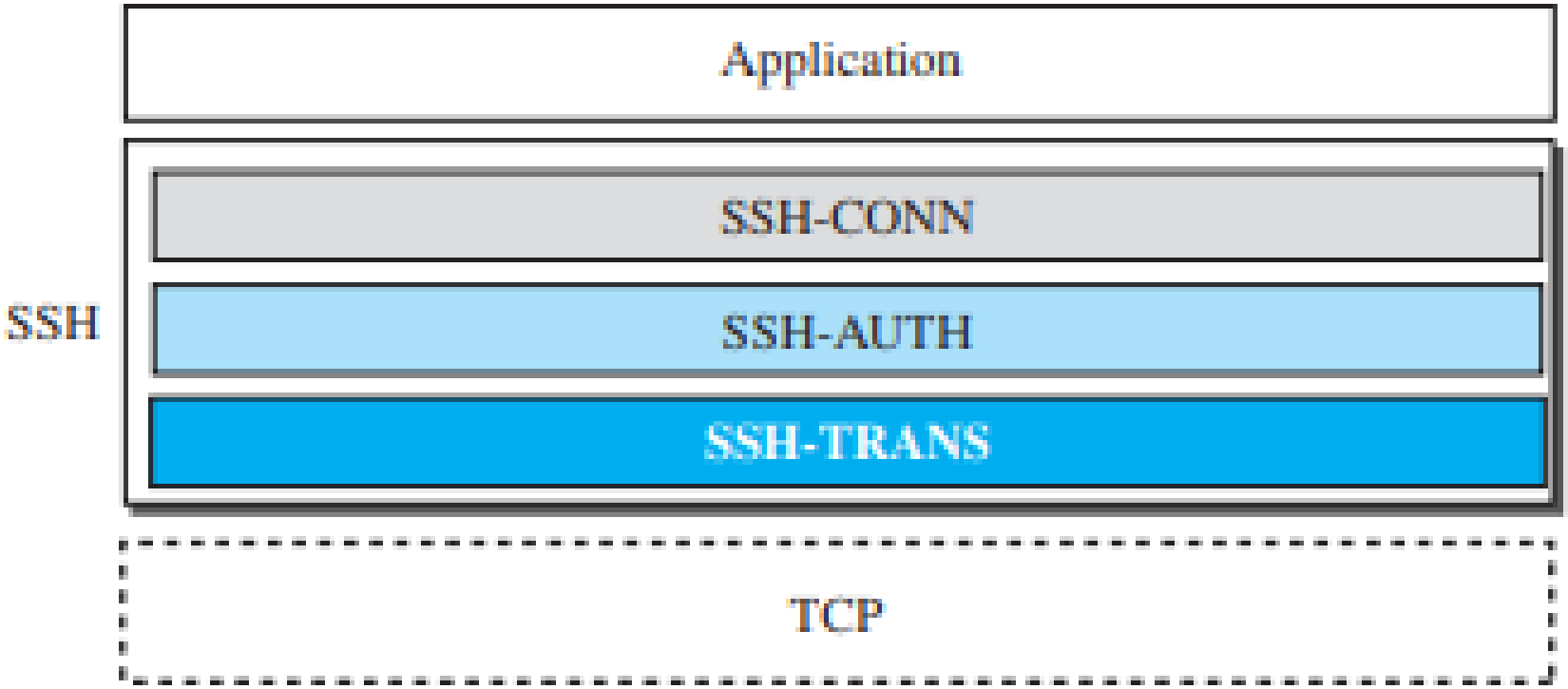
Student at CSE Dept of SSNCE

October 22, 2021

Introduction

Although Secure Shell (SSH) is a secure application program that can be used today for several purposes such as remote logging and file transfer, it was originally designed to replace **TELNET**. There are two versions of SSH: SSH-1 and SSH-2, which are totally incompatible. The first version, SSH-1, is now deprecated because of security flaws in it. In this section, we discuss only **SSH-2**.

Figure 26.25 *Components of SSH*



SSH Transport-Layer Protocol (SSH-TRANS)

- Since *TCP is not a secured transport-layer protocol*, SSH first uses a protocol that creates a secured channel on top of the TCP. This new layer is an independent protocol referred to as SSH-TRANS.
- When the procedure implementing this protocol is called, the client and server first use the TCP protocol to establish an insecure connection.
- Then they exchange several security parameters to establish a secure channel on top of the TCP.

SSH Transport-Layer Protocol (SSH-TRANS)

Services provided:

1. **Privacy** or **confidentiality** of the message exchanged
2. **Data integrity**, which means that it is guaranteed that the messages exchanged between the client and server are not changed by an intruder
3. **Server authentication**, which means that the client is now sure that the server is the one that it claims to be
4. **Compression** of the messages, which improves the efficiency of the system and makes attack more difficult

SSH Authentication Protocol (SSH-AUTH)

After a secure channel is established between the client and the server and the server is authenticated for the client, SSH can call another procedure that can authenticate the client for the server.

The **client authentication** process in SSH is very similar to what is done in Secure Socket Layer (SSL). This layer defines a number of authentication tools similar to the ones used in SSL.

Authentication starts with the client, which sends a request message to the server. The request includes the ***user name, server name, the method of authentication, and the required data***. The server responds with either a success message, which confirms that the client is authenticated, or a failed message, which means that the process needs to be repeated with a new request message.

SSH Connection Protocol (SSH-CONN)

After the secured channel is established and both server and client are authenticated for each other, SSH can call a piece of software that implements the third protocol, SSH-CONN.

One of the services provided by the SSH-CONN protocol is **multiplexing**. SSH-CONN takes the secure channel established by the two previous protocols and lets the client create **multiple logical channels** over it. Each channel can be used for a different purpose, such as remote logging, file transfer, and so on.

Applications - SSH for Remote Logging

Several free and commercial applications use SSH for remote logging. Among them, we can mention **PuTTy**, by Simon Tatham, which is a client SSH program that can be used for remote logging. Another application program is **Tectia**, which can be used on several platforms.

Applications - SSH for File Transfer

One of the application programs that is built on top of SSH for file transfer is **the Secure File Transfer Program** (sftp). The sftp application program uses one of the channels provided by the SSH to transfer files. Another common application is called **Secure Copy** (scp). This application uses the same format as the UNIX copy command, cp, to copy files.

Applications – Port forwarding

- We can use the secured channels available in SSH to access an application program that does not provide security services. Applications such as TELNET and Simple Mail Transfer Protocol (SMTP) can use the services of the SSH port forwarding mechanism.
- The SSH port forwarding mechanism creates a tunnel through which the messages belonging to other protocols can travel. For this reason, this mechanism is sometimes referred to as **SSH tunneling**.
- The FTP client can use the SSH client on the local site to make a secure connection with the SSH server on the remote site. Any request from the FTP client to the FTP server is carried through the tunnel provided by the SSH client and server. Any response from the FTP server to the FTP client is also carried through the tunnel provided by the SSH client and server.

Figure 26.26 *Port forwarding*

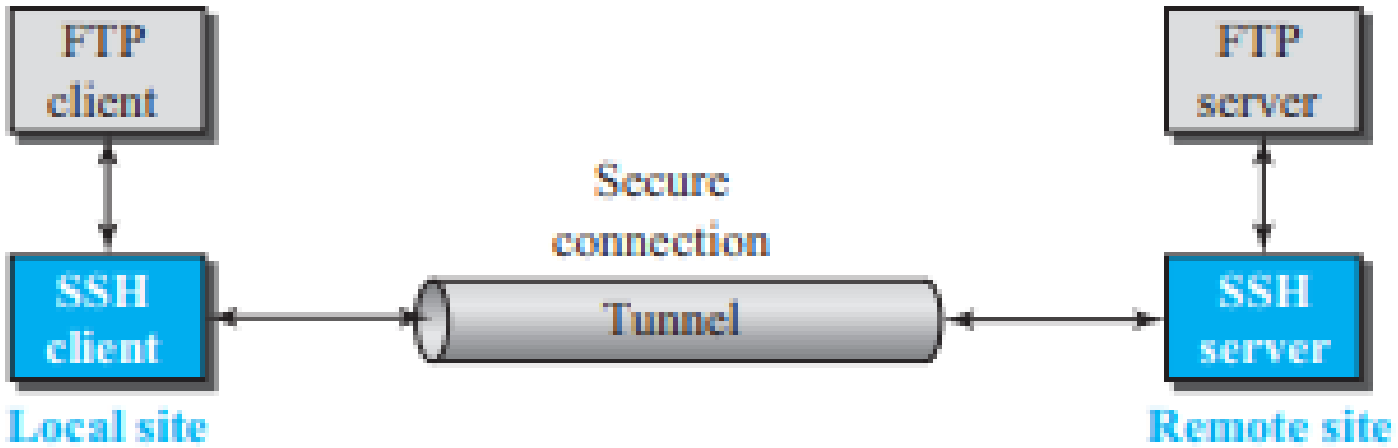
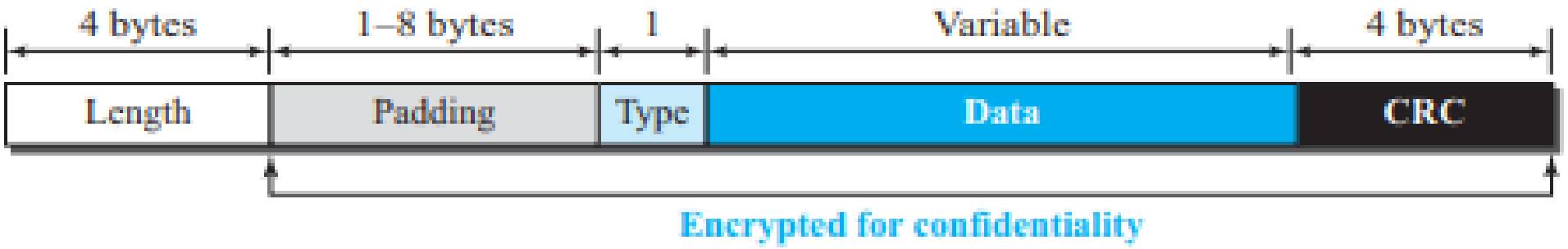


Figure 26.27 *SSH packet format*



The **length field** defines the length of the packet but does not include the padding.

One to eight bytes of **padding** is added to the packet to make the attack on the security provision more difficult.

The **cyclic redundancy check (CRC) field** is used for error detection.

The **type field** designates the type of the packet used in different SSH protocols.

The **data field** is the data transferred by the packet in different protocols.

Examples

TELNET Connection:

telnet towel.blinkenlights.nl

telnet freechess.org 5000