Chapter 5

Application Layer

- STANDARD CLIENT SERVER APPLICATIONS
  - HTTP
  - FTP
  - SMTP
  - **Telnet**
  - **SSH**
  - DNS

TErminaL NETwork (***TELNET***)

- ❑ Local versus Remote Logging
- ❑ Network Virtual Terminal (NVT)
- ❑ Options & User Interface

TELNET

A server program can provide a specific service to its corresponding client program. However, it is impossible to have a client/server pair for each type of service we need. Another solution is to have a specific client/server program for a set of common scenarios, but to have some generic client/server programs that allow a user on the client site to log into the computer at the server site and use the services available there. We refer to these generic client/server pairs as remote logging applications. One of the original remote logging protocols is TELNET.

- One of the original remote logging protocols is TELNET, which is an abbreviation for **TE**rmina**L NET**work
- Connection Oriented Protocol (Well defined Port Number: 23)
- TELNET requires a logging name and password
- It sends all data including the password in plaintext (not encrypted)
- Simple plaintext architecture of TELNET allows us to explain the issues and challenges related to the concept of remote logging
- Network administrators often use TELNET for diagnostic and debugging purposes

Local logging

When a user logs into a local system, it is called **local login**

- User types at a terminal or at a workstation running a terminal emulator, the keystrokes are accepted by the terminal driver.
- The terminal driver passes the characters to the operating system.
- The operating system, in turn, interprets the combination of characters and invokes the desired application program or utility.

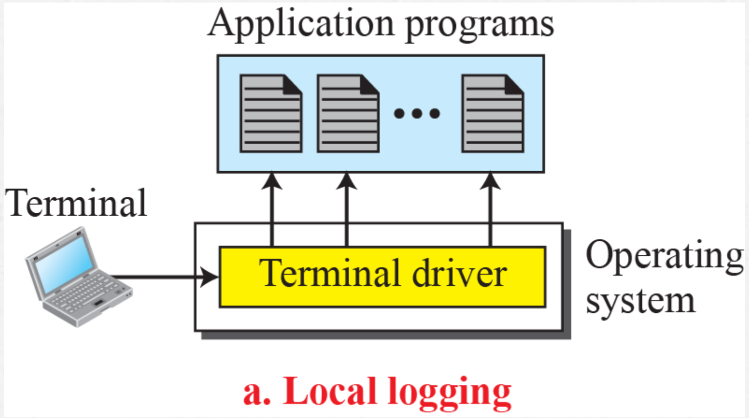
Remote logging

When a user wants to access an application program or utility located on a remote machine, she performs **remote logging**

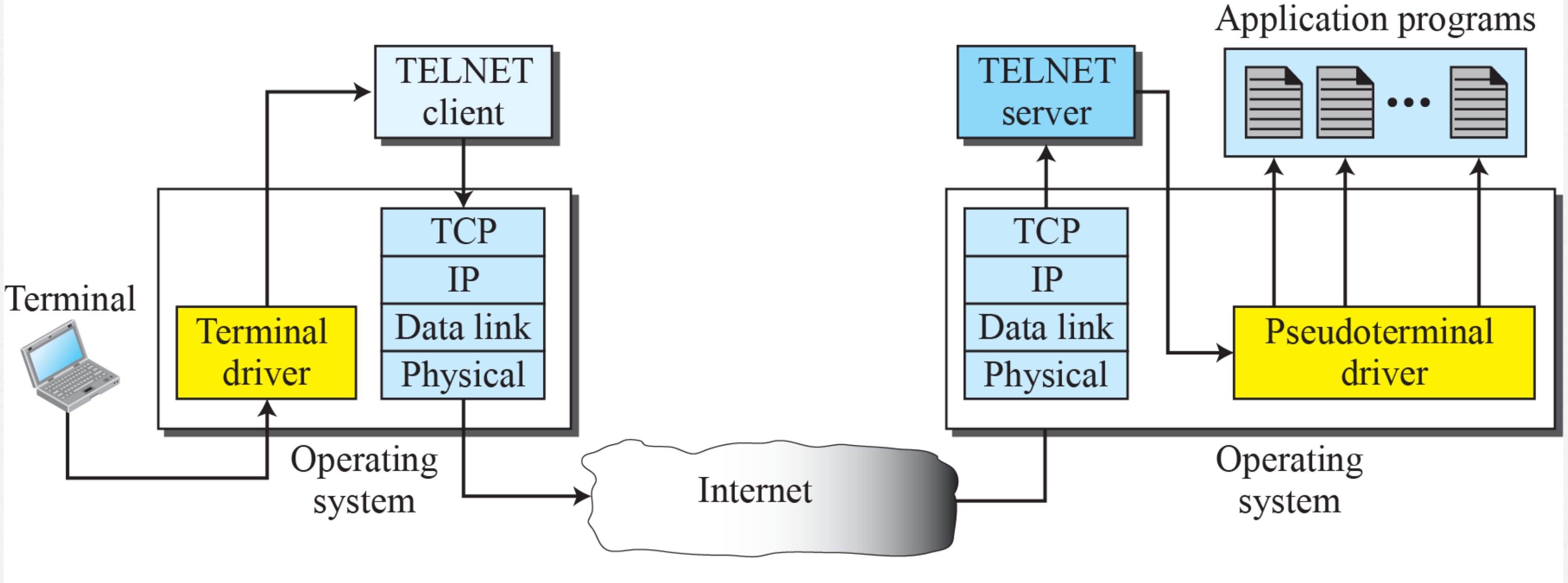
Here the TELNET client and server programs come into use.

- The user sends the keystrokes to the terminal driver where the local operating system accepts the characters but does not interpret them.
- The characters are sent to the TELNET client, which transforms the characters into a universal character set called Network Virtual Terminal (NVT) characters (discussed below) and delivers them to the local TCP/IP stack.
- The commands or text, in NVT form, travel through the Internet and arrive at the TCP/IP stack at the remote machine.
- Here the characters are delivered to the operating system and passed to the TELNET server.
- However, the characters cannot be passed directly to the operating system because the remote operating system is not designed to receive characters from a TELNET server; it is designed to receive characters from a terminal driver
- The solution is to add a piece of software called a pseudoterminal driver, which pretends that the characters are coming from a terminal.
- The operating system then passes the characters to the appropriate application program

Local versus remote logging



a. Local logging



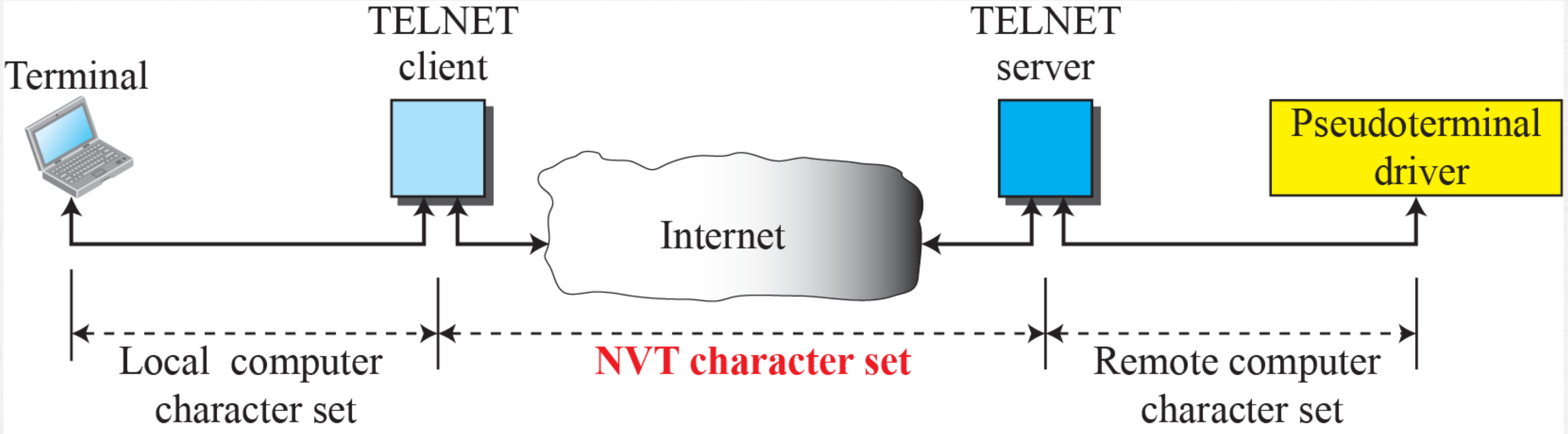
b. Remote logging

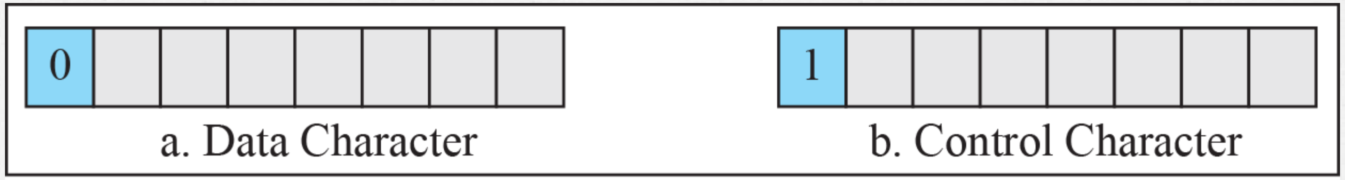
Network Virtual Terminal (NVT)

- In a heterogeneous system, the mechanism to access a remote computer is complex.
- This is because every computer and its operating system accepts a special combination of characters as tokens.
- If we want to access any remote computer in the world, we must first know what type of computer we will be connected to, and we must also install the specific terminal emulator used by that computer.
- TELNET solves this problem by defining a universal interface called the **Network Virtual Terminal (NVT)** character set
- The client TELNET translates characters (data or commands) that come from the local terminal into NVT form and delivers them to the network.
- The server TELNET, on the other hand, translates data and commands from NVT form into the form acceptable by the remote computer

Concept of NVT

NVT uses two sets of characters, one for data and one for control
Both are 8-bit bytes.(NVT ASCII)



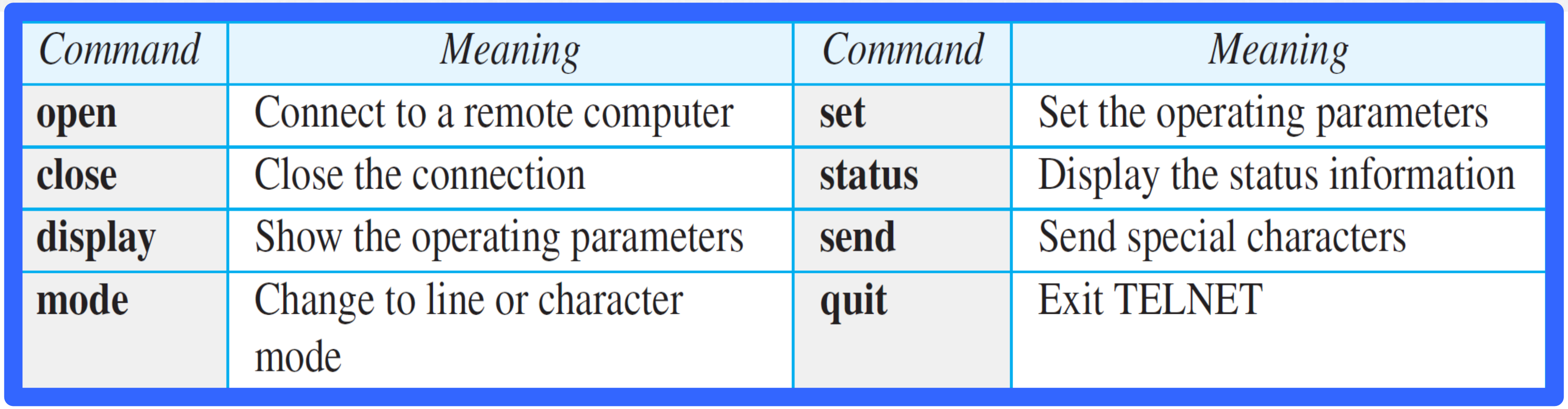


NVT character format

Options & User interface commands

TELNET lets the client and server negotiate options before or during the use of the service. Options are extra features available to a user with a more sophisticated terminal.

The operating system (UNIX, for example) defines an interface with user-friendly commands.



Command	*Meaning*	*Command*	*Meaning*
open	Connect to a remote computer	**set**	Set the operating parameters
close	Close the connection	**status**	Display the status information
display	Show the operating parameters	**send**	Send special characters
mode	Change to line or character mode	**quit**	Exit TELNET

Secure Shell (SSH)

- ❑ Components
  - ❖ SSH Transport-Layer Protocol (SSH-TRANS)
  - ❖ SSH Authentication Protocol (SSH-AUTH)
  - ❖ SSH Connection Protocol (SSH-CONN)
- ❑ Applications
  - ❖ SSH for Remote Logging
  - ❖ SSH for File Transfer
- ❑ Port Forwarding
- ❑ Format of the SSH Packets

Secure Shell (SSH)

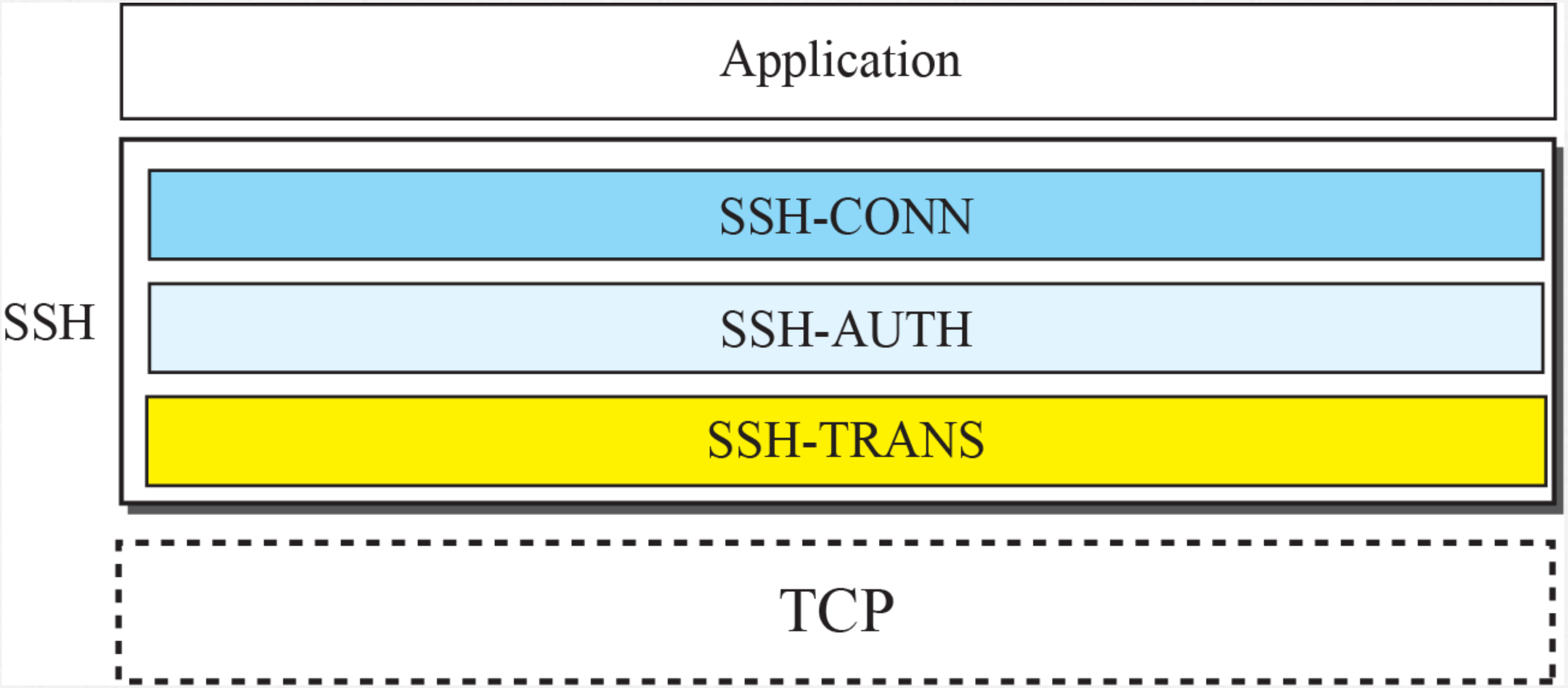
Although Secure Shell (SSH) is a secure application program that can be used today for several purposes such as remote logging and file transfer, it was originally designed to replace TELNET.

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.

Connection Oriented protocol (Well Known Port No. : 22)

Components of SSH



SSH Transport-Layer Protocol (SSH-TRANS)
SSH Authentication Protocol (SSH-AUTH)
SSH Connection Protocol (SSH-CONN)

SSH Transport-Layer Protocol (SSH-TRANS)

- 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
- Services Provided by SSH-Trans are:
  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)
- 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.

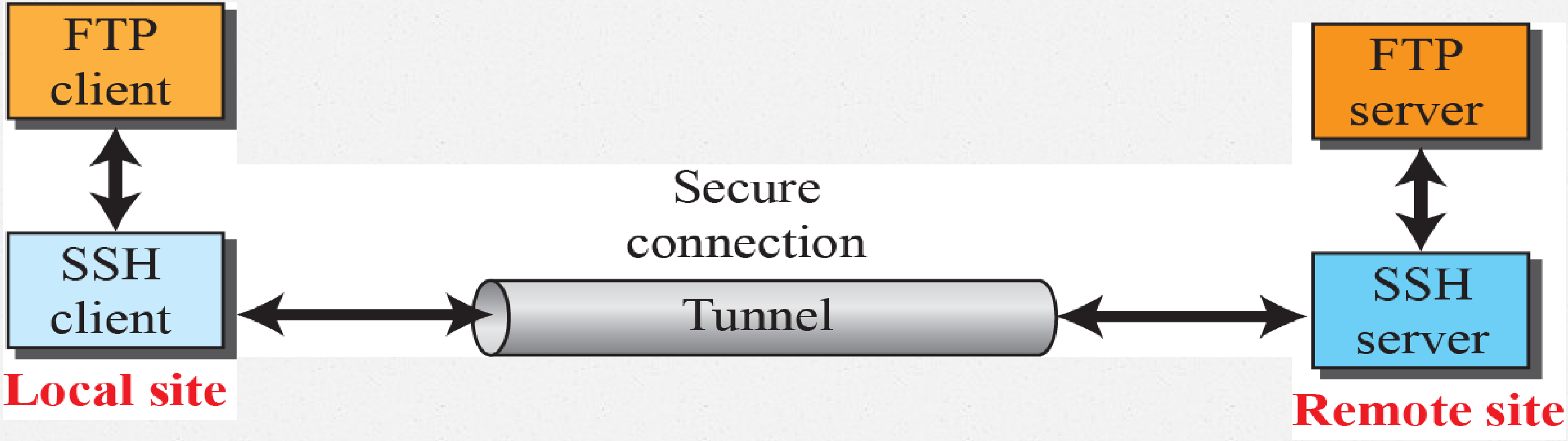
SSH Connection Protocol (SSH-CONN)

- 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, SSHCONN.
- 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.

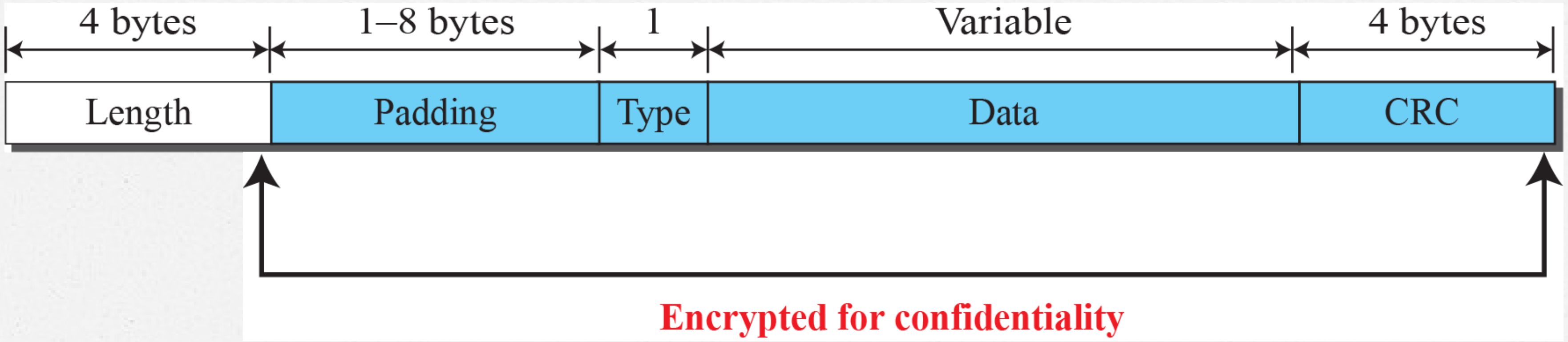
Port Forwarding

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), which are discussed, 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**.



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.

SSH 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.

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.