TELNET Basics

BUPT/QMUL

2014-04-15

Refer to Chapter 24, Textbook

北京郵電大學 BEIJING UNIVERSITY OF POSTS AND TELECOMMUNICATIONS

Electronic Engineering

Agenda

- A brief introduction to TELNET
- Concept of remote/virtual terminal
- TELNET operations
- TELNET protocol
- TELNET options negotiation
- Other remote access technologies

A Brief Introduction to TELNET

Brief Introduction To TELNET

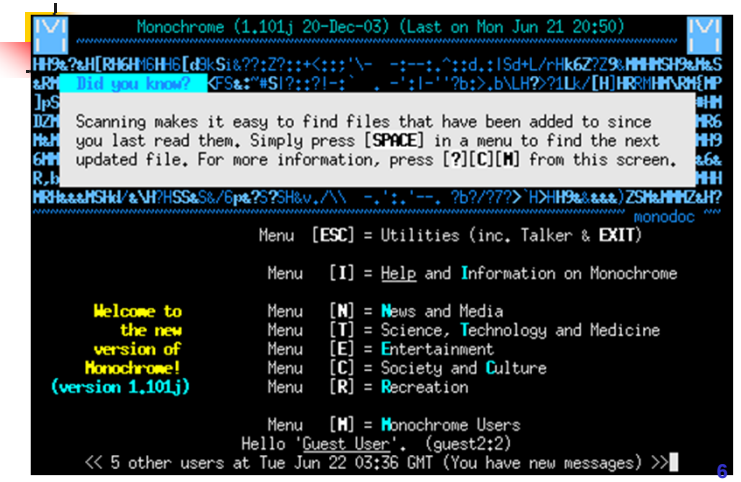
- Use of TELNET: BBS
- What is TELNET?
- History
- Major Ideas

BBS: Bulletin Board System

- A computer system running software that allows users to connect and log in to the system using a terminal program.
- Once logged in, a user can perform functions such as
  - uploading and downloading software and data;
  - reading news and bulletins;
  - and exchanging messages with other users, either through Email or in public message boards.

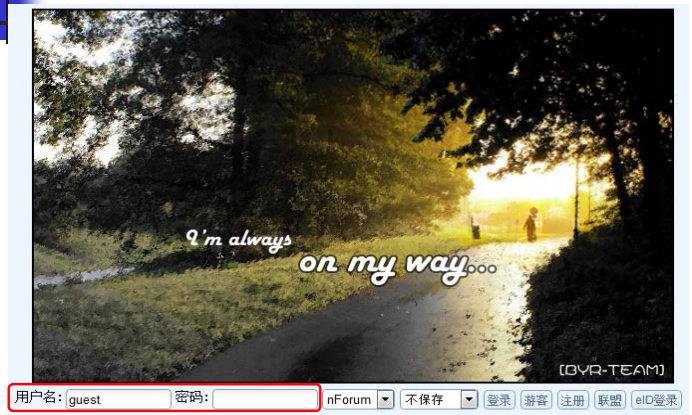
5

Traditional BBS example:



6

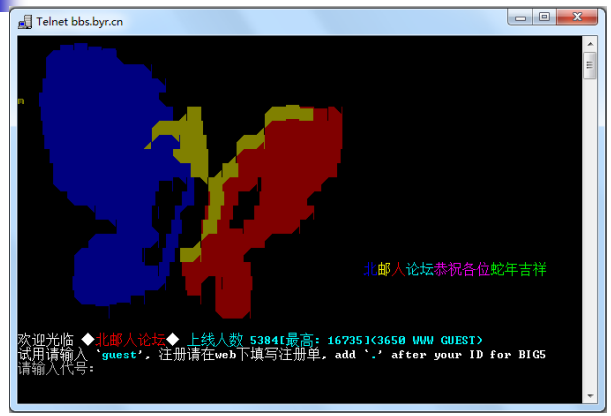
Example of BBS Today(1)



7

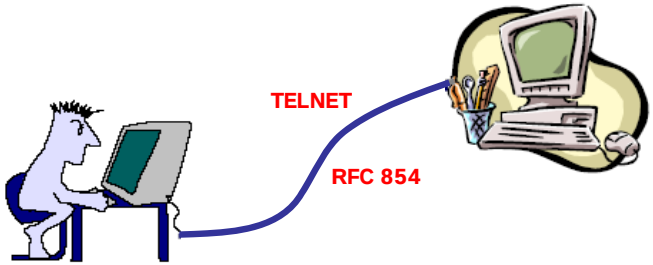
Example of BBS Today(2)

telnet BBS of BYR



What Is TELNET? (1)

- A protocol used to establish a dumb terminal session to another computer on the Internet
- An important Internet application for remote access



What Is TELNET? (2)

- Definition in RFC854
  - The purpose of the TELNET Protocol is to provide a general, bi-directional, byte oriented communications facility.
  - Its primary goal is to allow a standard method of interfacing terminal devices and terminal-oriented processes to each other.
  - It is envisioned that the protocol may also be used for terminal-terminal communication ("linking") and process-process communication (distributed computation).

TELNET vs. telnet

- TELNET is a *protocol* that provides “a general, bi-directional, eight-bit byte oriented communications facility”
- telnet is a *program* that supports the TELNET protocol over TCP
- Many application protocols are built upon the TELNET protocol

The History Of Telnet

- Telnet is simple
  - Total pages of RFC 854 is 15
  - HTTP (we will see later) is 176 pages
- The idea of option negotiation was a very good design feature
  - Enables telnet to evolve to meet new demands without endless new versions of basic protocol
- Currently over 100 RFCs on telnet and its options

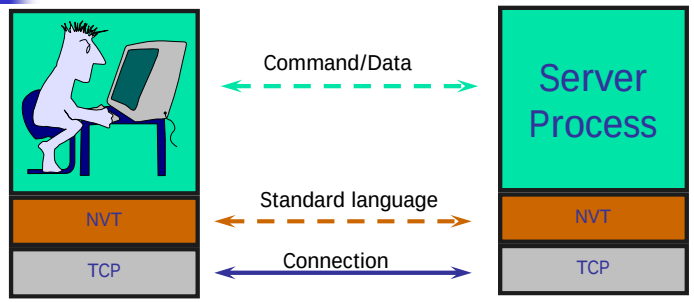
13

Major Ideas Of TELNET

- The concept of a NVT (Network Virtual Terminal)
  - Providing a standard interface to remote systems
- The principle of negotiated options
  - Enabling Telnet to evolve to meet new demands without endless new versions of basic protocol
- A symmetric view of terminals and processes
  - Allowing an arbitrary program to become a client

14

Network Virtual Terminal



15

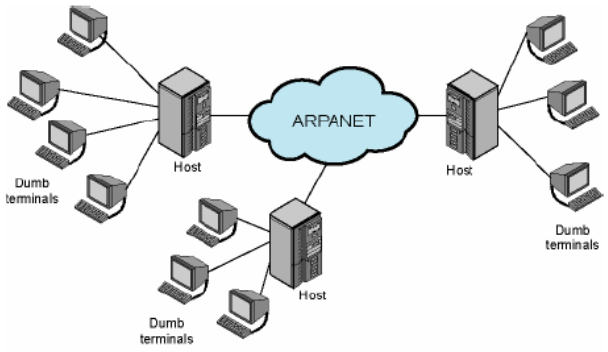
Concept Of Remote / Virtual Terminal

16

Remote Terminal Access

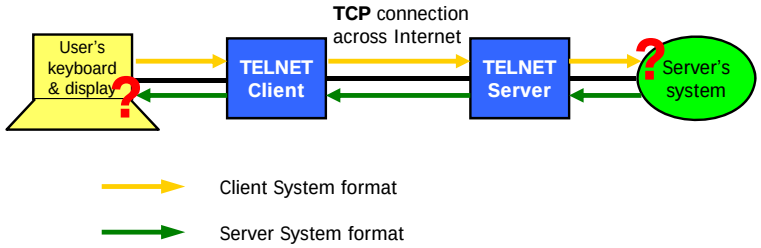
- Early motivation for networks was remote access to interactive systems
- Dumb terminals (see figure on the next slide)
  - Keyboard and screen with primitive communication hardware
  - Local host computer establish connection to remote host
- The challenge is that terminals and host systems were not standardized
  - local terminal was not speaking the same language as the remote host

Telnet Operation Environment On Early Internet



Problem

- Lack of common language between the terminal and the remote host

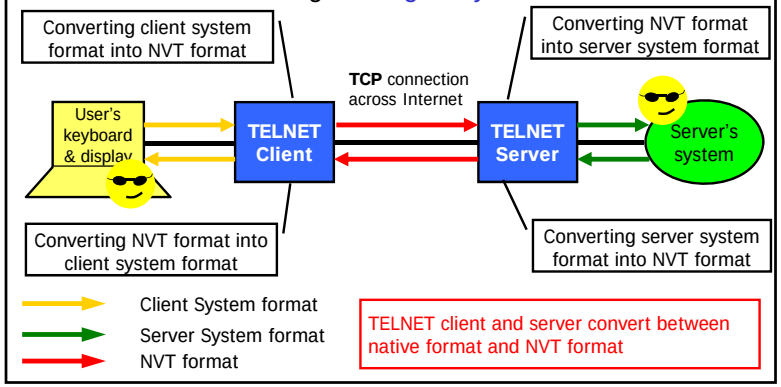


Network Virtual Terminal

- The approach to solve the problem of lack of a common language was to define a common language
  - Virtual terminal protocol (VTP)
- Transform local characteristics into standardized form
  - Network virtual terminal (NVT)
- Imaginary device
  - Well defined set of characteristics
- Both sides generate data and control signals in native language but translates them to NVT form
  - The sending side translates native data and control signals into NVT form before sending out
  - the receiving side gets the NVT data and signals and translates into its native form

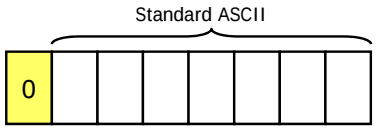
NVT Operation

- Accommodating heterogeneity



NVT Format

- NVT use two sets of characters: data and control
- Format of data characters



- Format of control characters

22

TELNET Operations

23

TELNET Operations

- Connection management
  - Connection request, establish and terminate
  - Telnet uses TCP (port 23) by default
- Negotiation
  - To determine mutually agreeable set of characteristics and options
- Exchange of control information (e.g. end of line), commands and transfer of data between two correspondents
- A typical telnet session is exchange of data between terminal and host
  - Multiple rounds
  - Not only for accessing remote accounts; was also used for interactive system
    - Try “telnet bbs.byr.cn”

24

TELNET Protocol

Related RFCs

- Basic protocol
  - RFC854: Telnet Protocol Specification
- Options
  - RFC855: Telnet Option Specifications
  - RFC856: Telnet Binary Transmission
  - RFC857: Telnet Echo Option
  - RFC858: Telnet Suppress Go Ahead Option
  - RFC859: Telnet Status Option
  - …

Some Features

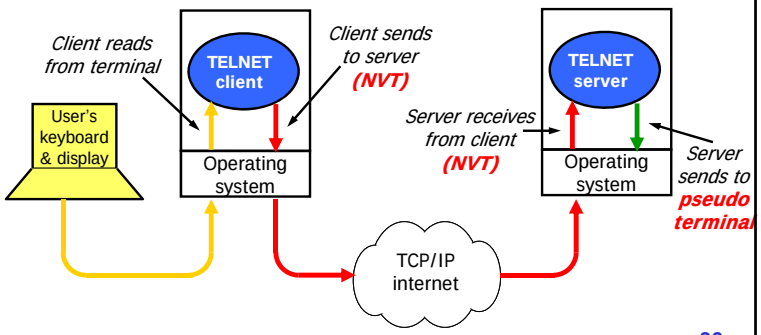
- TCP connection: directed toward port 23 of the server being asked to perform a service
- Data and control multiplexed over the same connection
- NVT - representation of a generic terminal
- Negotiated options - provides a standard language for communication of terminal control functions

TELNET Protocol

- Transmission of data
- Standard representation of control functions

Transmission Of Data (1)

- Data path from the user's keyboard to the remote system



Transmission Of Data (2)

- Data sent half-duplex
  - Terminal-to-process, newline signifies end of user input
  - Process-to-terminal, control signal Go Ahead(GA) is used
- Underlying TCP full duplex
  - Control signals sent any time regardless of current data direction
- Data sent as stream of 8-bit bytes
  - No other formatting
- Control signals and other non-data information sent as Telnet commands
  - Byte strings embedded in data stream
  - User control signals, commands between Telnet processes as part of protocol and option negotiation and subnegotiation

Control Functions (1)

- TELNET includes support for a series of control functions commonly supported by servers
- This provides a uniform mechanism for communication of (the supported) control functions
- You can imagine them as some extra virtual keys in the NVT keyboard

Control Functions (2)

- Interrupt Process (IP)
  - Suspend/interrupt/abort/terminate process
- Abort Output (AO)
  - allow a process, which is generating output, to run to completion but without sending the output to the user's terminal
- Are You There (AYT)
  - check to see if system is still running
- Erase Character (EC)
  - delete last character sent
  - typically used to edit keyboard input
- Erase Line (EL)
  - delete all input in current line
  - typically used to edit keyboard input

Control Functions (3) – delivery

Command	Decimal Codes	Description
IAC	255	Interpret next octet as command
DONT	254	Denial of request to perform specific option
DO	253	Approval to allow specific option
WONT	252	Refusal to perform specific option
WILL	251	Agreement to perform specific option
SB	250	Start of option subnegotiation
GA	249	Go ahead
EL	248	Erase line
EC	247	Erase character
AYT	246	Are you there
AO	245	Abort output
IP	244	Interrupt process
BRK	243	Break
DMARK	242	Data mark
NOP	241	No operation
SE	240	End of subnegotiation
EOR	239	End of record

Control Functions (4) – IAC

- TELNET command structure
  - at least a two byte sequence: the IAC (Interpret as Command) escape character followed by the code for the command
- The IAC code is 255
  - If a 255 is sent as data - it must be followed by another 255
- Looking for a command
  - Each receiver must look at each byte that arrives and look for an IAC
  - If IAC is found and the next byte is "IAC" - a single data byte (value 255) is presented to the application/ terminal
  - If IAC is followed by any other code - the TELNET layer interprets this as a command

34

Control Functions (5) – DO, DONT, WILL, WONT

- Used for options negotiation
- Examples

Sender	Receiver	Meaning
WILL →	← DO	Sender wants to active a option, and receiver agrees
WILL →	← DON'T	Sender wants to active a option, and receiver refuses
DO →	← WILL	Sender wants receiver to active a option, and receiver agrees
DO →	← WONT	Sender wants receiver to active a option, and receiver refuses

35

TELNET Options Negotiation

36

Motivations

- All NVTs support a minimal set of capabilities
- Some terminals have more capabilites than the minimal set
- The two endpoints negotiate a set of mutually acceptable options (character set, echo mode, etc)
- The set of options is not part of the TELNET protocol, so that new terminal features can be incorporated without changing the TELNET protocol

37

Option Examples

- echo modes
  - Keyboard input be echoed on the terminal side or not
- Line mode vs. character mode
  - One line or one character per transmission
- character set (EBCDIC vs. ASCII)
  - EBCDIC - Extended Binary-Coded Decimal Interchange Code
  - ASCII - American Standard Code for Information Interchange

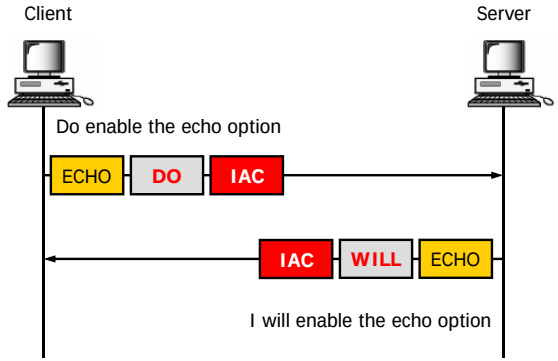
38

Options Negotiation

- Each option is assigned a byte value
- The DO, DONT, WILL, and WONT commands are used to negotiate options
- Options negotiation is symmetric
- Steps must be taken to avoid option processing loops
- Subnegotiations are used when more information is needed, such as when negotiating terminal type, window size, etc

39

Example: Negotiation of Echo Option



40

TELNET Options List (1)

Option	Name	References
0	Binary Transmission	RFC 856
1	Echo	RFC 857
2	Reconnection	NIC 50005
3	Suppress Go Ahead	RFC 858
4	Approx Message Size Negotiation	ETHERNET
5	Status	RFC 859
6	Timing Mark	RFC 860
7	Remote Controlled Trans and Echo	RFC 726
8	Output Line Width	NIC 50005
9	Output Page Size	NIC 50005
10	Output Carriage-Return Disposition	RFC 652
11	Output Horizontal Tab Stops	RFC 653
12	Output Horizontal Tab Disposition	RFC 654
13	Output Formfeed Disposition	RFC 655
14	Output Vertical Tabstops	RFC 656
15	Output Vertical Tab Disposition	RFC 657
16	Output Linefeed Disposition	RFC 658
17	Extended ASCII	RFC 698
18	Logout	RFC 727
19	Byte Macro	RFC 735

41

TELNET Options List (2)

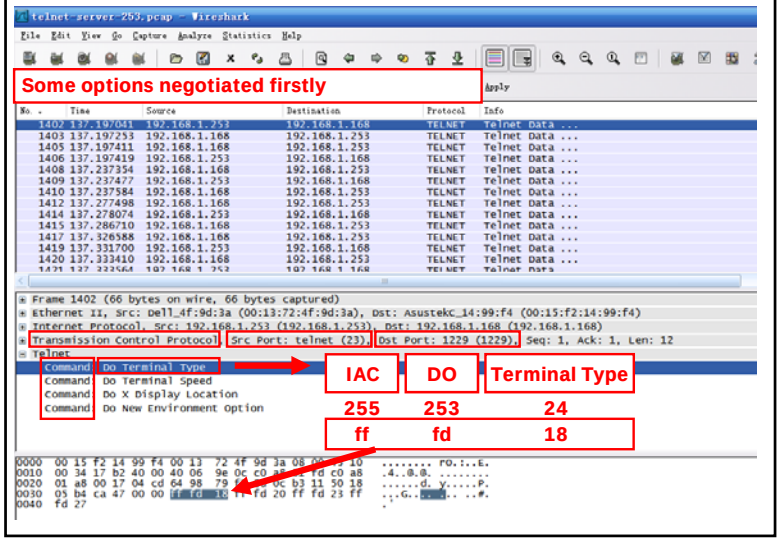
Option	Name	References
20	Data Entry Terminal	RFC 1043, RFC 732
21	SUPDUP	RFC 736, RFC 734
22	SUPDUP Output	RFC 749
23	Send Location	RFC 779
24	Terminal Type	RFC 1091
25	End of Record	RFC 885
26	TACACS User Identification	RFC 927
27	Output Marking	RFC 933
28	Terminal Location Number	RFC 946
29	Telnet 3270 Regime	RFC 1041
30	X.3 PAD	RFC 1053
31	Negotiate About Window Size	RFC 1073
32	Terminal Speed	RFC 1079
33	Remote Flow Control	RFC 1372
34	Linemode	RFC 1184
35	X Display Location	RFC 1096
36	Environment Option	RFC 1408
37	Authentication Option	RFC 2941
38	Encryption Option	RFC 2946

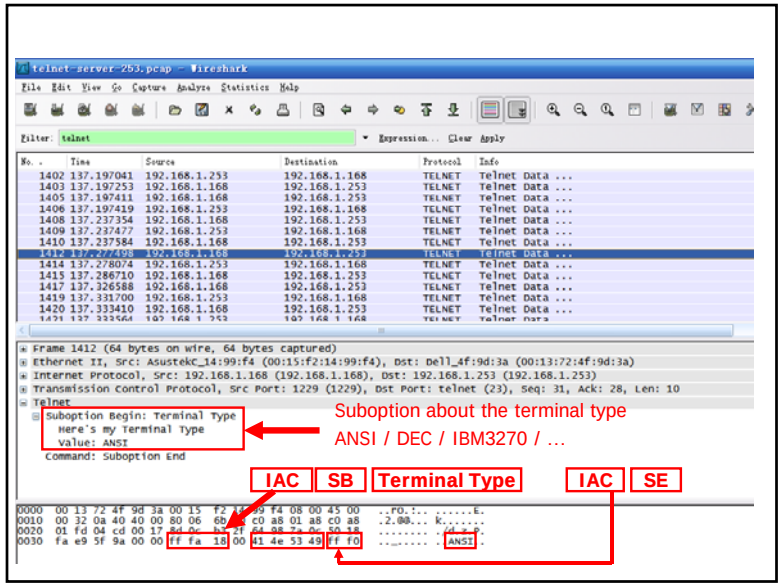
42

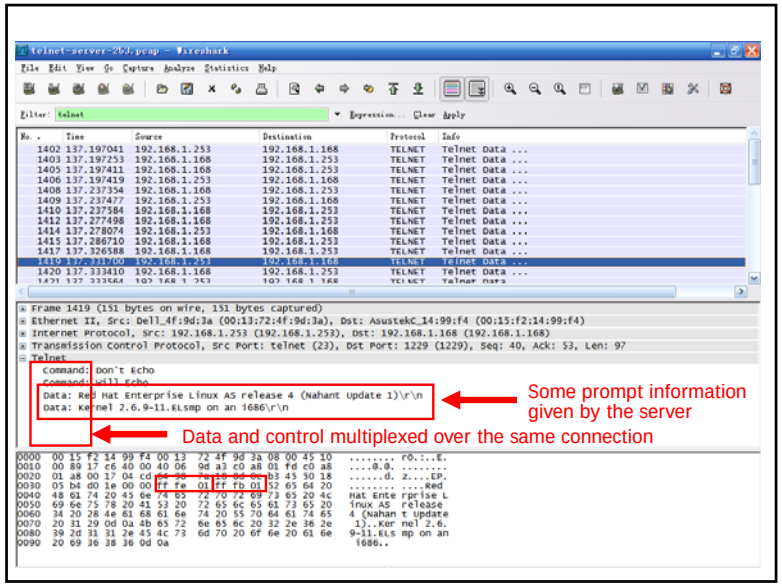
A Telnet Session Example (1)

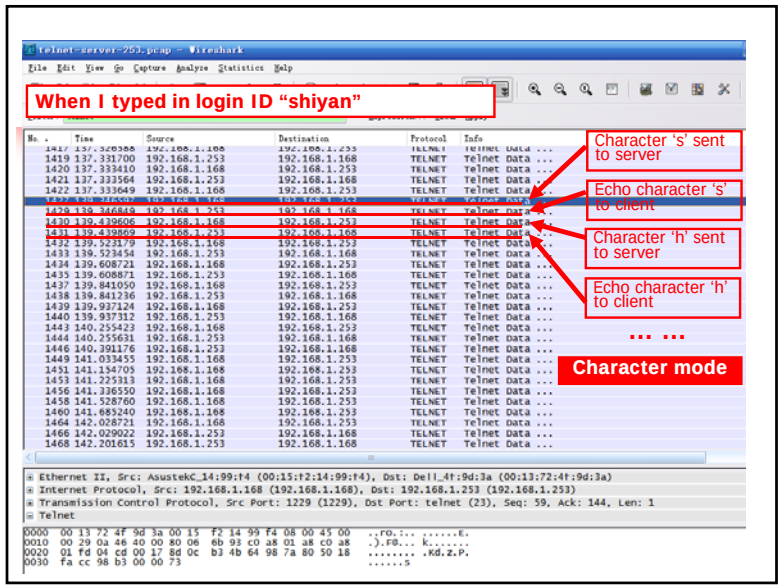
```
C:\Documents and Settings\Administrator> telnet 192.168.1.253
Red Hat Enterprise Linux AS release 4 <Nahant Update 1>
Kernel 2.6.9-11.Elsmp on an i686
Login: shiyan
Password:
Last login: Sun Nov 11 17:48:30 from 192.168.1.168
[shiyan@localhost ~]$
```

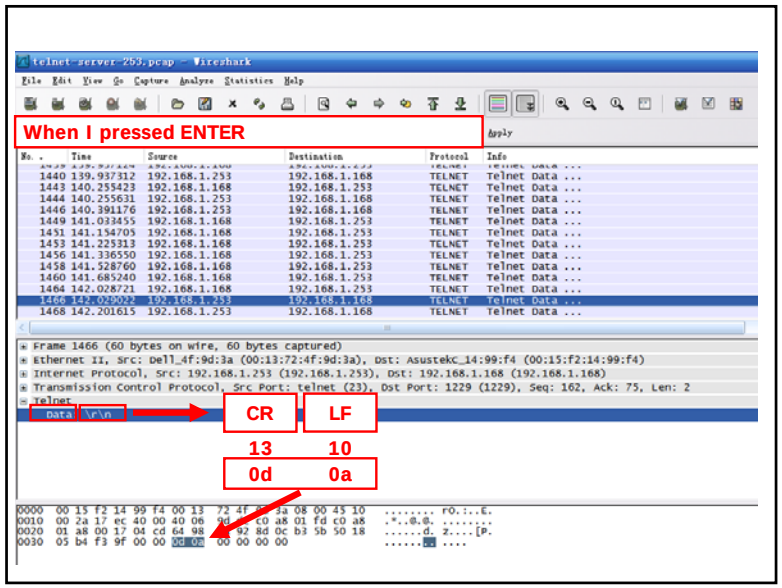
43




Suboption about the terminal type
ANSI / DEC / IBM3270 / …
IAC
SB
Terminal Type
IAC
SE


Some prompt information given by the server
Data and control multiplexed over the same connection


When I typed in login ID “shiyan”
Character ‘s’ sent to server
Echo character ‘s’ to client
Character ‘h’ sent to server
Echo character ‘h’ to client
… …
Character mode


When I pressed ENTER
CR
LF
13
10
0d
0a

Summary (1) – usages of telnet

- Use Internet accounts you may have on remote computers
  - you need an account (login ID) and password on the remote computer to permit access
- Use free services accessible with telnet, e.g.
  - library catalogues
  - databases
  - BBS (Bulletin Board System)
  - Router/switch configuration

49

Summary (2) – Disadvantages of telnet

- Poor user interface
  - Based on dumb terminal
  - Text-only display
  - Monochrome
    - One color for text, one for background
  - Have to type command-line commands
    - Often have complex syntax
  - Not very secure, SSH made enhancement
    - TELNET does not encrypt any data sent over the connection (**including passwords**)

50

Other Remote Access Technologies

51

Other Remote Access Technologies

- Remote login in text-based system
  - telnet
  - SSH
  - Rlogin
- Remote desktop in windowing system
  - VNC (Virtual Network Computing)
  - RDP (Remote Desktop Protocol)

52

SSH (1) – brief information

- Secure Shell
- Command line terminal connection tool
- All traffic encrypted
- Both ends authenticate themselves to the other end
- Ability to carry and encrypt non-terminal traffic
- Private key kept on client, public key stored on server
- Now, it is an IETF standard
  - RFC4251, The Secure Shell (SSH) Protocol Architecture

53

SSH (2) – two enhancements of telnet

- Providing secure communications
- Providing users with the ability to perform additional, independent data transfer over the same connection that is used for remote login

54

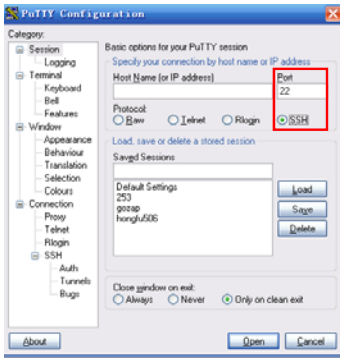
SSH (3) – three major mechanisms

- A transport layer protocol that provides sever authentication, data confidentiality, and data integrity with perfect forward secrecy
- A user authentication protocol that authenticates the user to the server
- A connection protocol that multiplexes multiple logical communications channels over a single underlying SSH connection
  - Port forwarding, could be used as a secure tunnel

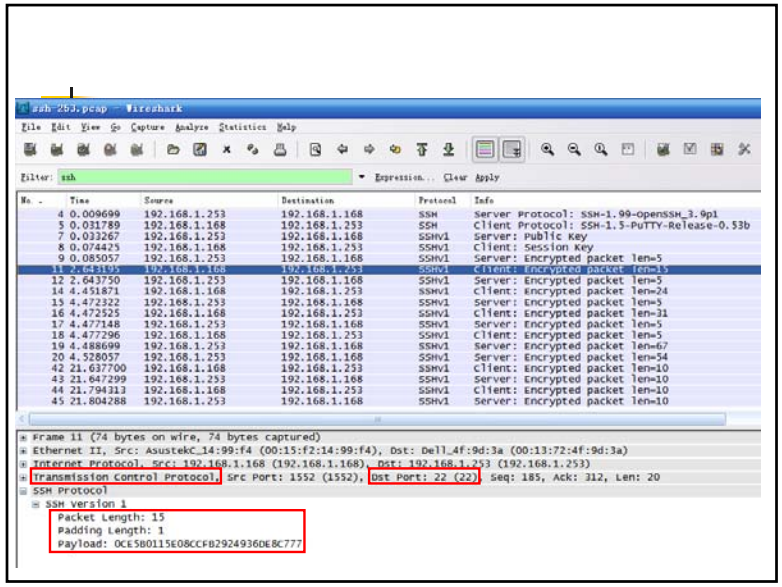
55

SSH (4) – tools

- PuTTy



56



Other Ways Of Remote Access

- Except telnet, there are other ways
  - rlogin family utility
  - VNC (Virtual network computing)
  - RDP (Remote Desktop Protocol)
- Comparison with Telnet

Helpful URLs

- RFCs
  - http://www.ietf.org/
- Useful utilities
  - http://winfiles.search.com/search?cat=316&tag=ex.sa.fd.srch.wf&q=TELNET
- About telnet
  - http://en.wikipedia.org/wiki/Telnet
  - http://baike.baidu.com/view/44255.htm
- About SSH
  - http://www.ssh.com
  - http://www.openssh.org
- About realVNC
  - http://www.realvnc.com/

Abbreviations of Week 6-9

BBS	Bulletin Board System
BOOTP	BOOTstrap Protocol
DHCP	Dynamic Host Configuration Protocol
DNS	Domain Name System
FQDN	Fully Qualified Domain Name
IAC	Interpret As Command
MSC	Message Sequence Chart
NVT	Network Virtual Terminal
SSH	Secure SHell
STD	State Transition Diagram
TLD	Top Level Domain
TLV	Type-Length-Value
TTL	Time To Live