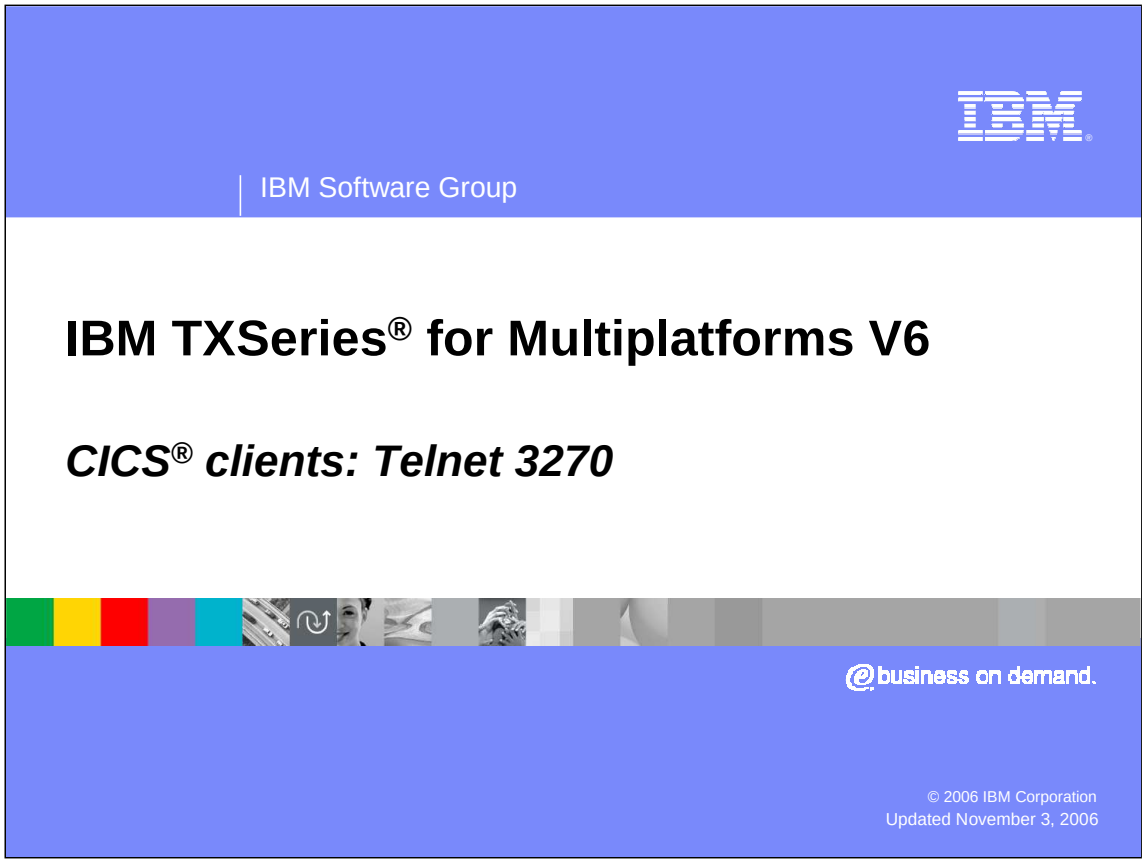


This presentation will cover the Telnet 3270 CICS client, which is one of the client deployment models used with TXSeries.

Goals

- Provide an overview of Telnet 3270 client
- Difference between a normal telnet session and a Telnet 3270 session
- How Telnet 3270 client communicates with the TXSeries region
- How to Configure, connect, unconfigure Telnet 3270 server

The goal is to provide an overview of the Telnet 3270 client, discuss differences between a normal telnet session and a Telnet 3270 session, and describe how the Telnet 3270 client communicates with the TXSeries region. How to configure, connect, and unconfigure Telnet 3270 server is also covered.

Agenda

- Telnet 3270 client Overview
  - What is Telnet 3270?
  - Difference between a normal Telnet session and a Telnet 3270 session?
  - Telnet 3270 client communication with TXSeries
- Configuring Telnet 3270 servers
- Unconfiguring Telnet 3270 servers
- Telnet 3270 specific considerations

The agenda for this presentation is to define Telnet 3270 and cover the related topics in the order shown here.

Section

Telnet 3270 client overview

This section will provide an overview of Telnet 3270 client.

Telnet Clients (TN3270)

- TN3270 is a protocol to emulate IBM 3270 terminal display over an IP network.
  - Originally defined in RFC 1576 and is based on the Telnet protocol.
- A Normal Telnet Session:
  - uses the American National Standard Code for information Interchange (ASCII) character set and sends a line of data at a time.
- A TN3270 Telnet Session:
  - uses the Extended Binary Coded Decimal Interchange Code (EBCDIC) character set and sends a block of data at a time.

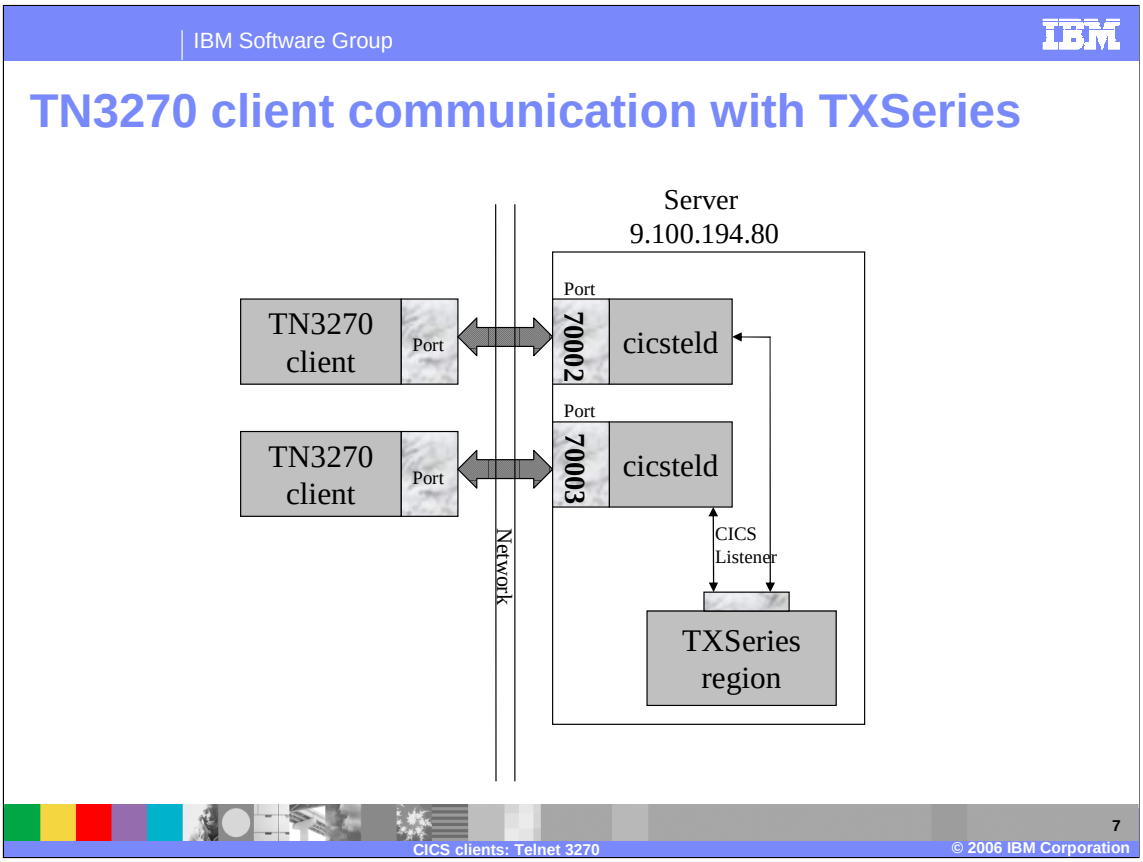
TN3270 is a protocol to emulate IBM 3270 terminal display over an IP network. This is originally defined in RFC 1576 and is based on the Telnet protocol. The basic difference between a normal Telnet session and a TN3270 session is that Telnet uses the American National Standard Code for Information Interchange (ASCII) character set and sends a line of data at a time. TN3270r uses the Extended Binary Coded Decimal Interchange Code (EBCDIC) character set and sends a block of data at a time. The TN3270 is primarily enabled to connect to a 3270-based CICS system over an IP network.

Telnet Clients (TN3270)

- TN3270 communication has two elements:
  - TN3270 client
    - A front end that emulates the 3270 display terminal
  - TN3270 server
    - A listener program to a TN3270 client.
    - converts the TN3270 data stream to Systems Network Architecture (SNA) 3270 data stream and passes the data to the TXSeries region.
    - resides on the machine where the TXSeries region is running.
    - TXSeries provides a TN3270 server named '**cicsteld'**

The TN3270 communication has two elements. The first is the TN3270 client, which is a front end that emulates the 3270 display terminal and accesses a TN3270 server over an IP network.

The second is the TN3270 server, a listener to a TN3270 client. The server converts the TN3270 data stream to Systems Network Architecture, or SNA, 3270 data stream and passes the data to the TXSeries region. The TN3270 server resides on the machine where the TXSeries region is running.



This figure depicts the TN3270 client communication with TXSeries.

TXSeries provides a TN3270 server called *cicsteld*, which establishes a Telnet session with a TN3270 client, and uses TERMINAL-TYPE negotiation for the Telnet option.

It is important to note that each TN3270 client session requires a dedicated TN3270 cicsteld server process running on the server machine.

Section

Telnet 3270 server configuration

This section will cover configuration of the TN3270 server.

Configuring TN3270 server on the region

- **cicscp create telnet_server** command:

  cicscp -v create telnet_server <servername>

  - Adds an entry in the operating system's inetd.conf
  - Creates a script file under the cics_clients directory
  - Registers a Telnet port with the operating system
- To connect from a TN3270 client
  - Hostname and the port

Use the **cicscp create telnet_server** command to configure TN3270 server on a region. Alternatively, you can start the cicsteld manually by passing command line parameters. The cicsteld session is terminated when the TN3270 session is disconnected. When a TN3270 client connects, the inetd operating system process automatically executes the shell script located in the /var/cics_clients directory, thereby creating a TN3270 cicsteld server process. Then the cicsteld server process establishes the Telnet session with the TN3270 client, after negotiating the Telnet options.

To connect from a TN3270 client, you must enter the host name and the port number on which a TN3270 server is listening. When the client establishes a connection with the server, it lists all the available regions running on the server. Select the region you want to connect to, and run the transaction.

Unconfiguring TN3270 server for a region

- cicscp destroy telnet_server command:

cicscp -v destroy telnet_server <*servername*>

<*servername*> is the name of the TN3270 server.

You can run the cicscp –v destroy telnet_server command to unconfigure the TN3270 server.

TN3270 considerations

- Code Page conversions
  - Code page conversions are necessary as the TN3270 client and the CICS region operate in two different character sets.
- Terminal mode emulation
  - The cicsteld server automatically determines the terminal mode to be used when communicating with TN3270 client.
  - Override the default by specifying –e emulation option in the cicsteld process

Code page conversions are necessary because the TN3270 client and the CICS region operate in two different character sets. The TN3270 client uses the EBCDIC character set as supported by the Telnet protocol, and the CICS server uses the ASCII character set. Therefore, it is mandatory for the cicsteld server process to convert from EBCDIC to ASCII when a request is received from the client, and to convert from ASCII to EBCDIC when a response is sent back to the client. The cicsteld server process uses client code page 037 (EBCDIC) by default and 8859-1 (ASCII) as the default server code page. If you are using a different locale, you can change the default code page by specifying the *-c clientCodePage* and *-s serverCodePage* respectively in the cicsteld server process.

The cicsteld server automatically determines the terminal mode emulation to be used when communicating with TN3270 client. However, if you choose to use a user-defined terminal mode, you can override the default by specifying the *-e* emulation option in the cicsteld server process. The preferred CICS model types that can be used by cicsteld are:

hft and mft for terminals with extended data stream and

lft for terminals that do not have extended data stream

TN3270 considerations

- Startup transaction
  - Automatically run a startup transaction, after the connection is established with the TN3270 client.
- Using LUName
  - Identify clients through the LUName option of the TN3270 client
  - Client must use TN3270E services to use this option
  - You can access the value specified in this option through the NETNAME

You can set up the cicsteld server process to run a startup transaction, after the connection is established with the TN3270 client. For example, you can run the CESN transaction as the first transaction to enable users to log on to your system. You can do this by specifying the transaction name using the –t initialTransaction option in the cicsteld server process.

The *LUName* option in the TN3270 client is used to represent a logical unit name connected over an SNA network. This option is available only for TN3270E supported server. However, TXSeries uses this field with the ability to represent an eight-character name, which is usually used to identify the clients connected to the region. To enable this, you must first turn on the appropriate options to indicate to the client that it should use TN3270E services when supported. On the TXSeries regions side, you can access the value specified in this field (LUName) through the NETNAME.

Section

Summary

This section will provide a summary of topics covered by this presentation.

Summary

- Provide an overview of Telnet 3270 client
- Difference between a normal telnet session and a Telnet 3270 session
- How Telnet 3270 client communicates with the TXSeries region
- How to Configure/Connect/Unconfigure Telnet 3270 server

In summary, this presentation has covered the Telnet 3270 client, including differences between TN3270 and normal telnet, how the TN3270 communicates with the TXSeries region, and how to configure the TN3270 server.

Trademarks, copyrights, and disclaimers

The following terms are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both:

IBM	CICS	IMS	MQSeries	Tivoli
IBM(logo)	Cloudscape	Informix	OS/390	WebSphere
e(logo)business	DB2	iSeries	OS/400	xSeries
AIX	DB2 Universal Database	Lotus	pSeries	zSeries

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are registered trademarks of Microsoft Corporation in the United States, other countries, or both.

Intel, ActionMedia, LANDesk, MMX, Pentium and ProShare are trademarks of Intel Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Linux is a registered trademark of Linus Torvalds.

Other company, product and service names may be trademarks or service marks of others.

Product data has been reviewed for accuracy as of the date of initial publication. Product data is subject to change without notice. This document could include technical inaccuracies or typographical errors. IBM may make improvements and/or changes in the product(s) and/or program(s) described herein at any time without notice. Any statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only. References in this document to IBM products, programs, or services does not imply that IBM intends to make such products, programs or services available in all countries in which IBM operates or does business. Any reference to an IBM Program Product in this document is not intended to state or imply that only that program product may be used. Any functionally equivalent program, that does not infringe IBM's intellectual property rights, may be used instead.

Information is provided "AS IS" without warranty of any kind. THE INFORMATION PROVIDED IN THIS DOCUMENT IS DISTRIBUTED "AS IS" WITHOUT ANY WARRANTY, EITHER EXPRESS OR IMPLIED. IBM EXPRESSLY DISCLAIMS ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NONINFRINGEMENT. IBM shall have no responsibility to update this information. IBM products are warranted, if at all, according to the terms and conditions of the agreements (e.g., IBM Customer Agreement, Statement of Limited Warranty, International Program License Agreement, etc.) under which they are provided. Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products in connection with this publication and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. IBM makes no representations or warranties, express or implied, regarding non-IBM products and services.

The provision of the information contained herein is not intended to, and does not, grant any right or license under any IBM patents or copyrights. Inquiries regarding patent or copyright licenses should be made, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. All customer examples described are presented as illustrations of how those customers have used IBM products and the results they may have achieved. The actual throughput or performance that any user will experience will vary depending upon considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve throughput or performance improvements equivalent to the ratios stated here.

Note to U.S. Government Users - Documentation related to restricted rights-Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract and IBM Corp.