IBM Software Group

z/OS® V1R9 Communications Server

Overview: FTP and TN3270

business on demand.

Updated December 3, 2007

This presentation discusses the FTP and TN3270 enhancements for z/OS V1R9 Communications Server.

FTP SSL/TLS RFC compliance

- FTP was originally enabled for SSL/TLS back in z/OS V1R2
- That draft RFC has since that time undergone several revisions and has now made it into official RFC status
- FTP now supports RFC 4217 “Security FTP with TLS”

```
                    +--DRAFT----+
                    |           |
>--TLSRFCVERSION--+-----------+--><
                    |           |
                    +--RFC4217--+
```


FTP was originally enabled for SSL/TLS back in z/OS V1R2 Communications Server based on a draft RFC that described how the FTP protocol was to work with SSL/TLS. That draft RFC has since that time undergone several revisions and has now made it into official RFC status RFC 4217 “Securing FTP with TLS”. FTP was supporting Internet Draft 05 of this RFC. The RFC is less restrictive than the draft about flowing the AUTH and CCC commands to the server during a secure session. The upshot of this is that the full RFC 4217 functionality of the AUTH and CCC commands were not available to z/OS FTP users before z/OS V1R9 Communications Server.

A few changes have been made since the draft RFC version that was used in z/OS V1R2 Communications Server was written:

•Change USER command reply code from 232 to 230 if a password is not required.

•A REIN (Re-initialize) command on a control connection that is secured through an AUTH command should reset the TLS state. Currently, FTP resets everything on the control connection, except the TLS state when a REIN is processed. RFC 4217 says that the TLS state must also be cleared when this command is processed.

•The RFC explicitly states that the REIN server command reply must flow on the protected connection – the server cannot clear the session before sending the reply. The Internet Draft did not specify this level of detail. The z/OS server implementation clears the session before replying to REIN. Therefore, the z/OS FTP server did not interoperate with an RFC 4217 compliant FTP client when REIN is used during a TLS session. This is not as bad as you might think; REIN is not really recommended during an FTP session regardless of whether you are using TLS or not.

•Allow a CCC command on a control connection that is already secured through an AUTH command. Currently, FTP rejects such a command in this situation.

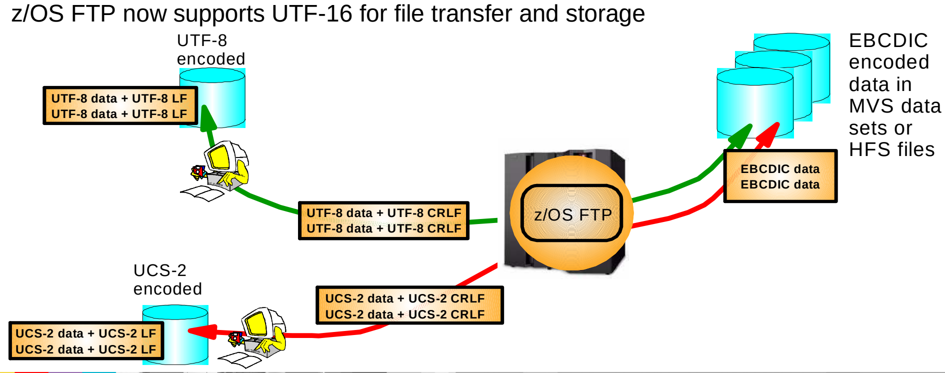
•Allow an AUTH command on a control connection that is already secured through an AUTH command. Currently, FTP rejects such a command in this situation. RFC 4217 says that an AUTH command re-initializes (if you are still running TLS) or reinstates security (if you did a CCC to clear the control connection).

•Stop using out-of-band data when connections are secured. RFC 4217 requires all commands (including ABOR and STAT) be sent over the TLS connection and not out-of-band.

•According to draft 05 of Securing FTP with TLS, when FTP clients connect to server port 990, the connection is secured with TLS without flowing an AUTH command – the connection is implicitly secured, as opposed to explicitly securing the connection by sending an AUTH command to the server. The RFC has dropped implicit security and secure port entirely. Thus, a connection between an RFC 4217 compliant FTP and an Internet Draft compliant FTP on the secure port cannot interoperate, because the Internet Draft side believes the connection is secure, and the RFC 4217 compliant side believes the connection is not secure. Again, this is not as bad as you might think. The existing TLSPORT statement for the client and server’s FTP.DATA allows you to reassign the TLSPORT, or disable it altogether. Therefore, the existing z/OS Communications Server provides a bypass for this problem.

FTP Unicode support

- Unicode requirements are becoming more and more common when exchanging data with
  - partner companies
  - various public services
  - government agencies
- FTP is an easy-to use technology that is common to all platforms making it very easy to exchange files
- z/OS FTP now supports UTF-16 for file transfer and storage

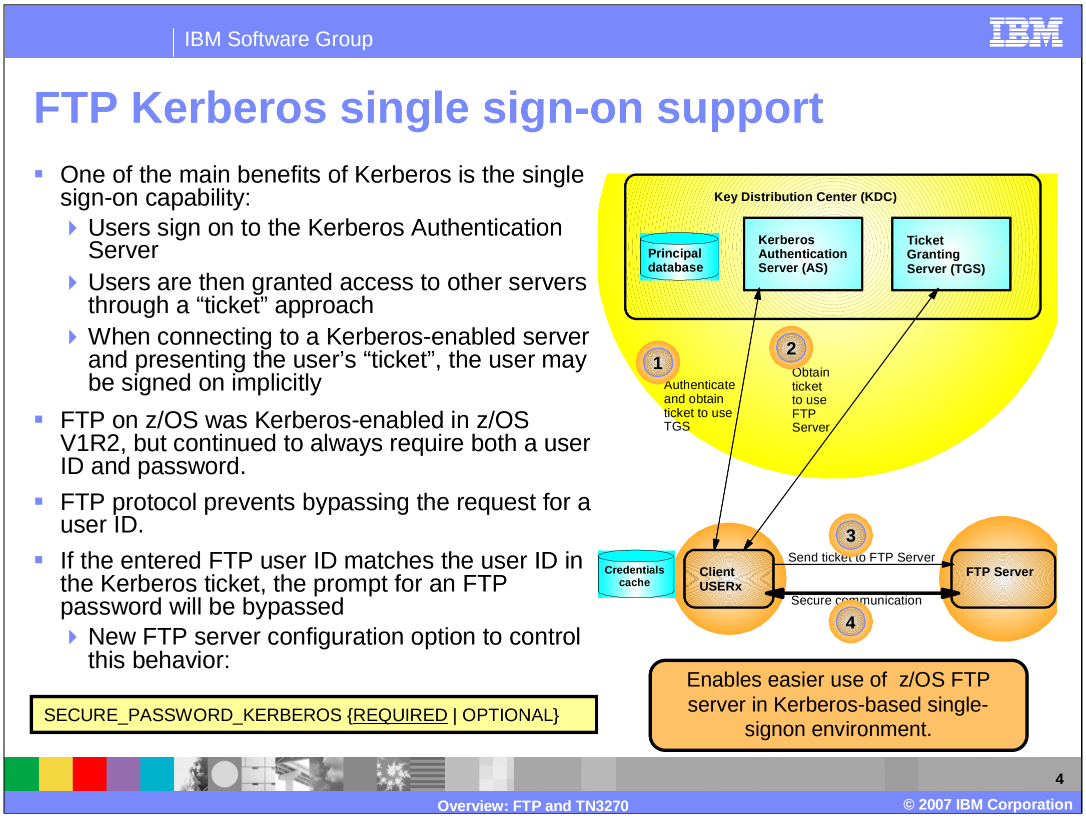


In z/OS V1R8 Communications Server, to support IBM Printing System's new support for UNICODE documents, Unicode File Transfer and storage support for UTF-8 was added. Users could now move UNICODE documents to a z/OS Communications Server host to store and to print.

The z/OS platform has started making use of the UTF-16 class of encodings. IBM Printing Systems supports UTF-16 encodings. The problem is users can not move these Unicode files with z/OS FTP.

z/OS V1R9 Communications Server builds upon the UNICODE support added in z/OS V1R8 Communications Server by adding support for UTF-16. For practical purposes, UTF-16 uses two bytes per character. A two byte character must use either little endian byte order or big endian byte order; therefore, UTF-16 is always either UTF-16BE or UTF-16LE. By definition, UTF-16 is UTF-16BE by default unless a BOM is present.

UTF-16 can be specified for the File system code page and the Network transfer code page can be specified as UTF-16, UTF-16LE, or UTF-16BE.



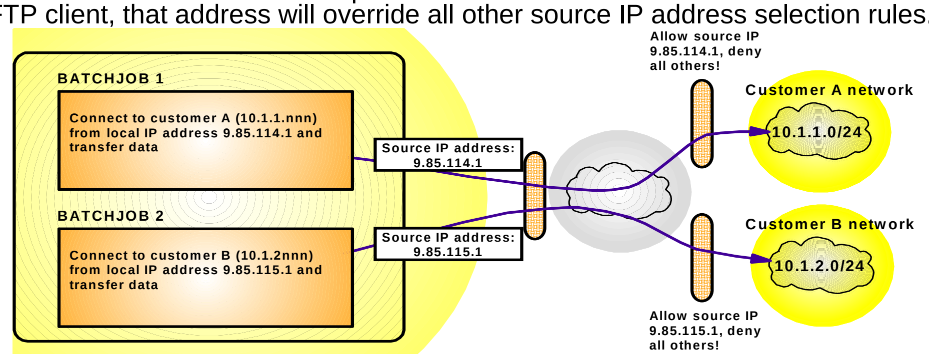
In a Kerberos environment, users must authenticate to the Kerberos Key Distribution Center (KDC) by supplying their user name and password. Users are also accustomed to using single sign-on support. The user authenticates once to the Kerberos KDC and then should be able to access and be authenticated by other services without having to enter their password again. However, if they then login to a Kerberos enabled z/OS FTP server, they must enter their user name and password again.

The solution to the problem is to allow users to login to the z/OS FTP server without having to re-enter the password. First, the user must authenticate to the Kerberos KDC. Then the user starts the FTP client and connects to the z/OS FTP server using GSSAPI authentication. GSSAPI, or Generic Security Service Application Programming Interface, is the authentication method used by the FTP protocol to allow connections between Kerberos enabled clients and servers.

The FTP protocol still requires that the client supply a user name to the FTP server. If the user name supplied to the z/OS FTP server is the same user name used to authenticate to the Kerberos KDC, the z/OS FTP server will not prompt for the password.

Allow FTP client to select source IP address

- Despite significant flexibility in how to direct TCP/IP to choose source IP address for outbound connections, there remains a need to be able to specify which source IP address a given FTP client invocation should use when connecting to an FTP server.
- The z/OS FTP client provides a new command line option where a user can specify which local IP address the connection to the FTP server should come from.
- When a source IP address is specified in the command line invocation of the FTP client, that address will override all other source IP address selection rules.



Despite significant flexibility in how to direct TCP/IP to choose source IP address for outbound connections, there remains a need to be able to specify which source IP address a given FTP client invocation should use when connecting to an FTP server. If batch FTP client job preparation is done by a group of people who do not have access to update (or maybe even view) the source IP address rules (SRCIP) in the TCP/IP profile, a need for them to specify a specific source IP address when preparing the batch jobs still exists.

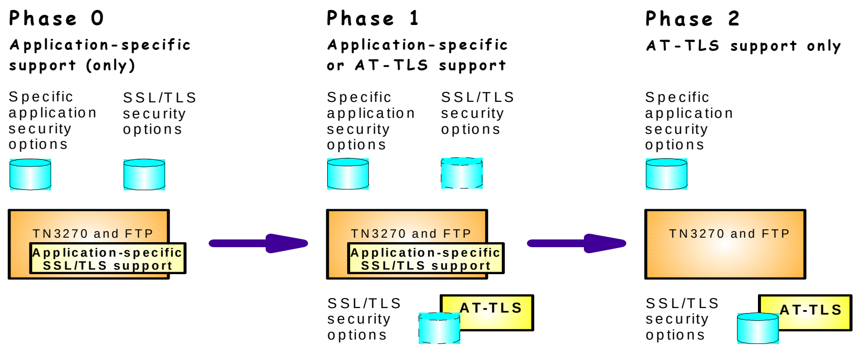
In the diagram, the customer has a network setup where the z/OS system running the FTP client has two interfaces into the network. The customer needs to be able to FTP into two other networks which are protected by firewalls. The firewalls are configured to only allow connections from specific IP addresses. So the only way to successfully FTP into “Customer A network”, is to use a source IP address of 9.85.114.1. Since there is no way for the FTP client to specify a source IP address, there is no guarantee that the TCP/IP stack would choose the correct interface. Since there are two interfaces into the network the TCP/IP stack may choose either interface.

In prior releases there was no way for the FTP client to specify which source IP address should be used when connecting to the FTP server. The TCP/IP stack determined the source IP address. This can be based on TCP/IP configuration options such as Job-Specific Source IP or it may be determined when the route to the FTP server is found. In some situations the FTP client may want to use a different source IP address when connecting to different FTP servers. In firewall configurations, it may be necessary to use a specific source IP address for the firewall to allow the connection. But, there was no way for the FTP client, itself, to specify the source IP address that should be used.

The z/OS V1R9 Communications Server FTP client supports a new command line option where a user can specify which local IP address the connection to the FTP server should come from. It is the user's responsibility to verify that the chosen address is a valid local IP address that is reachable from the FTP server node. When a source IP address is specified in the command line invocation of the FTP client, that address will override all other source IP address selection rules.

Enable AT-TLS for the TN3270E Telnet server and the FTP client and server

- Both the FTP server and client, and the TN3270E Telnet server on z/OS have in the past implemented SSL/TLS support
- Some system SSL functions are not available in FTP and Telnet
- AT-TLS implementation supports all System SSL functions
  - With the advantages of AT-TLS, it is desirable to enable FTP and Telnet for AT-TLS



When FTP implemented System SSL in z/OS V1R2 Communications Server, all the functions of System SSL were not exploited. System SSL allows for LDAP servers to be used for certificate revocation lists (CRLs). System SSL also supports specifying a certificate label to allow certificates other than the default certificate to be used. System SSL allows session keys to be refreshed during the lifetime of a session.

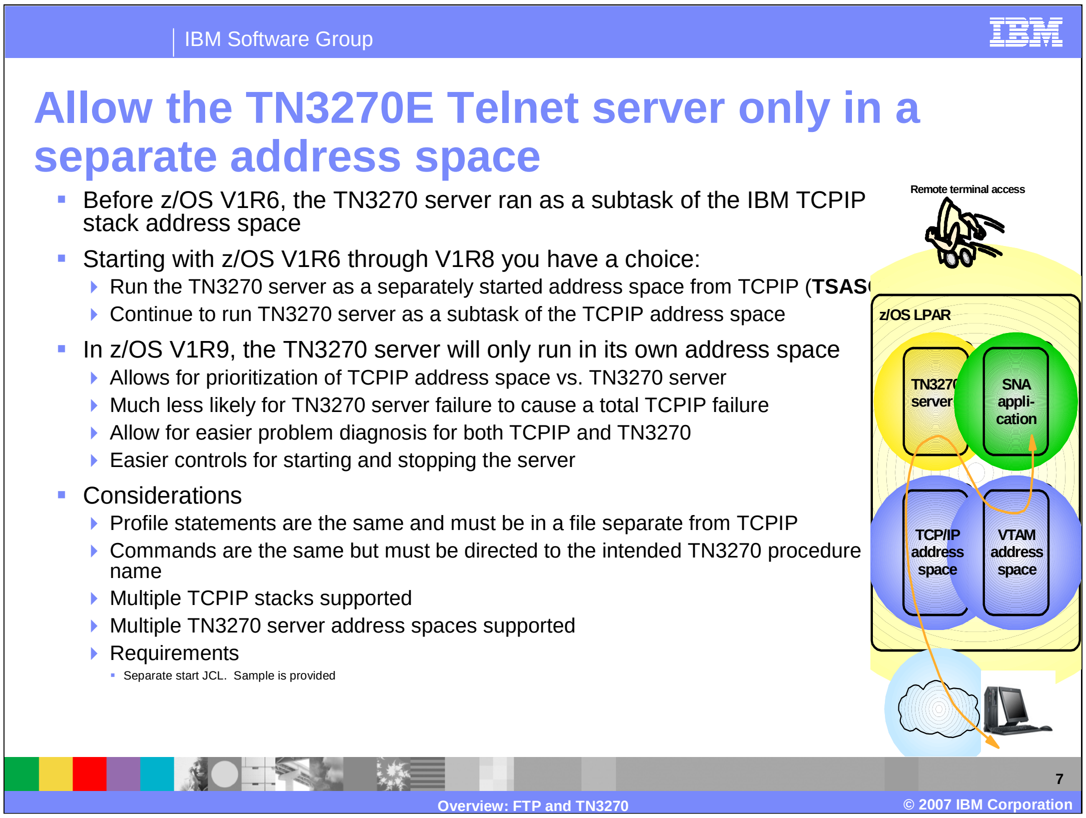
The FTP client and server can now be configured to use AT-TLS to support SSL/TLS connections. There are 3 types of AT-TLS applications. Those that are completely unaware they are using AT-TLS (they use AT-TLS with no code changes at all), those that have AT-TLS awareness but do not control AT-TLS (they can query the stack but not affect the choices it makes), and those that are AT-TLS controlling, meaning the application starts and stops security on the connection. FTP is a controlling AT-TLS application which requires the ApplicationControlled On statement in the AT-TLS policy.

Using AT-TLS allows all of the System SSL parameters supported by AT-TLS to be configured for FTP. For example, multiple LDAP servers can be configured or a certificate label can be configured instead of the default certificate. AT-TLS can also be configured to refresh the session key on a connection.

When Telnet first implemented secure connections on OS/390 V2R6, System SSL was not as robust as it is today. System SSL allowed only one active environment to support telnet connections. Telnet security setup was developed around that assumption and others based on System SSL capability at the time. For example, because only one System SSL environment could be activated, Telnet allows only one key ring name for all ports.

Customer have asked for Telnet to support different key rings on different ports and even different key rings on the same port. Customers have a need to be able to refresh security parameters without having to stop/restart the secure ports. This is particularly useful when the default certificate expires and must be replaced. Some customers have backup Certificate Revocation List Lightweight Directory Access Protocol, CRL LDAP, servers and would like to specify these backups. Customers would like to quickly use new ciphers that are periodically added. Customers have client emulators that support session ID caching and renegotiation of a cipher key during an active secure session. Customers want to specify a certificate label to be used instead of the default key ring certificate. System SSL has continued to improve and now supports these functions. Telnet configuration has not been enhanced to take advantage of the new System SSL function.

Application Transparent Transport Layer Security (AT-TLS) was introduced in z/OS V1R7 Communications Server and supports all of the new functions in System SSL. AT-TLS is the z/OS Communications Server strategic application security option and will continue to be updated as new System SSL functions become available. To satisfy existing Telnet security requirements, we could either make additional updates to Telnet configuration to make use of the new System SSL function or enable Telnet to fully utilize AT-TLS. Because AT-TLS is strategic and provides System SSL functions beyond the current requirements, we chose to enable Telnet for AT-TLS. With AT-TLS the customer will be able to specify multiple key rings for different ports or the same port, change key rings without stopping ports, specify up to five CRL LDAP servers, specify new ciphers immediately, cache session IDs, manage session IDs and cipher renegotiation, and use a certificate other than the default certificate during the SSL negotiations. Telnet provides the customer a great deal of flexibility through its current configuration options. That flexibility had to be retained while moving to AT-TLS. Being able to specify Conntype and client authentication at very granular levels is a popular Telnet feature that must be retained.



Telnet has been able to run in its own address space since z/OS V1R6 Communications Server which was generally available in September, 2004. Since that time, customers have had the option to continue configuring Telnet and TCP/IP to run in a single shared address space or configure Telnet to run in its own address space.

There are several advantages to running them separately. Telnet priority can be set to a different priority than that of TCP/IP. Telnet can be stopped and restarted without stopping TCP/IP. When the TCP/IP stack is stopped, Telnet remains active. Separating Telnet and TCP/IP makes problem diagnosis easier. You can start up to eight instances of Telnet. In a common INET environment, Telnet can be associated with multiple stacks, or have affinity to a single stack by using the TCP/IPJOBNAME statement in TELNETGLOBALS.

Note that even though you can have a maximum of 8 TN3270 server address spaces per LPAR, only one can activate the SNMP subagent (for response time data reporting using SNMP) in a stack. The TN3270 address space must have affinity to that stack. The first one started with stack affinity activates the SNMP subagent

The TN3270E Telnet server must run with stack affinity for the TN3270 SNMP subagent and WLM functions.

Dual support was implemented to allow careful, deliberate migration of Telnet from the TCP/IP address space into its own address space with the strategic direction that all customers will move Telnet to realize the TSASO advantages.

Feedback

Your feedback is valuable

You can help improve the quality of IBM Education Assistant content to better meet your needs by providing feedback.

- Did you find this module useful?
- Did it help you solve a problem or answer a question?
- Do you have suggestions for improvements?

Click to send e-mail feedback:

mailto:iea@us.ibm.com?subject=Feedback_about_whatsnewFTPtn.PPT

This module is also available in PDF format at: ../whatsnewFTPtn.pdf

You can help improve the quality of IBM Education Assistant content by providing feedback.

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 z/OS

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 or changes in the products or programs 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 (for example, 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.