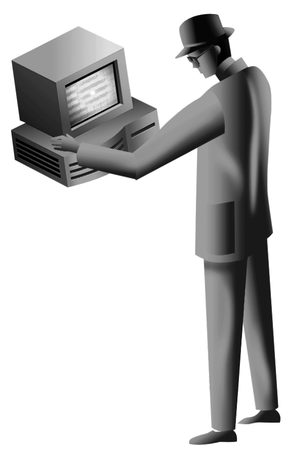
Citrix ICA Priority Packet Tagging

By Citrix Consulting Services

Citrix Systems, Inc.



Version History		
May 3, 2001	Melissa S. Hopkins with content provided by: • Kalyan Valicherla • Mike Discavage • Paul Wilson	Version 0.5
May 24, 2001	Melissa S. Hopkins - Updates from review with Brad Pedersen	Version 0.9
June 14, 2001	Melissa S. Hopkins with Jo Harder – Final Updates	Version 1.0

Table of Contents

Introduction

The Citrix ICA protocol includes a new feature in MetaFrame 1.8 Feature Release 1 and MetaFrame XP that identifies and tags ICA data based on the virtual channel from which the data originated. This feature, referred to as ICA Priority Packet Tagging, lays the foundation for a more granular Quality of Service (QoS) solution by providing the ability to prioritize ICA sessions based on the virtual channel data being transmitted.

This white paper describes virtual channel priorities and how ICA data is tagged with these priorities when sent over an Ethernet network using TCP/IP[1]. This white paper also discusses important considerations to be addressed by QoS solutions when implementing ICA Priority Packet Tagging.

This white paper assumes the reader is generally familiar with ICA virtual channels, the TCP/IP protocol, and QoS solutions.

[1] ICA Priority Packet Tagging is currently only available when using TCP/IP as the network protocol.

Virtual Channel Priorities

ICA Priority Packet Tagging provides the ability to prioritize ICA sessions based on the virtual channel data being transmitted. This is accomplished by associating each virtual channel with a two-bit priority. This two-bit priority is included as part of each ICA framing header (the ICA framing header is described in more detail in the section titled *ICA Data Transmission*). The two priority bits combine to form four priority values:

- 00 (0) – High Priority
- 01 (1) – Medium Priority
- 10 (2) – Low Priority
- 11 (3) – Background Priority

Each virtual channel is assigned one of these priority values. The default virtual channel priorities are as follows:

Virtual Channel	Default Priority	Description
CTXTW	0	Remote windows screen update data (Thinwire)
CTXTWI	0	Seamless windows screen update data (Thinwire)
CTXCLIP	1	Clipboard
CTXCAM	1	Client audio mapping
CTXLIC	1	License management
CTXVFM	1	Video server video (i.e. not Thinwire video)
CTXPN	1	Program Neighborhood
CTXCCM	2	Client COM port mapping
CTXCDM	2	Client drive mapping
CTXCM	3	Client management (Auto Client Update)
CTXLPT1	3	Printer mapping for non-spooling clients (i.e. WinTerms)
CTXLPT2	3	Printer mapping for non-spooling clients (i.e. WinTerms)
CTXCOM1	3	Printer mapping for non-spooling clients (i.e. WinTerms)
CTXCOM2	3	Printer mapping for non-spooling clients (i.e. WinTerms)
CTXCPM	3	Printer mapping for spooling clients
OEMOEM	3	Used by OEM's
OEMOEM2	3	Used by OEM's

The priority settings for all virtual channels are stored in the following Registry key:

[HKLM\System\CurrentControlSet\Control\Terminal Server\Wds\icawd\Priority] (REG_MULTI_SZ)

This key contains one line for each virtual channel in the format:

VirtualChannelName,*Priority*

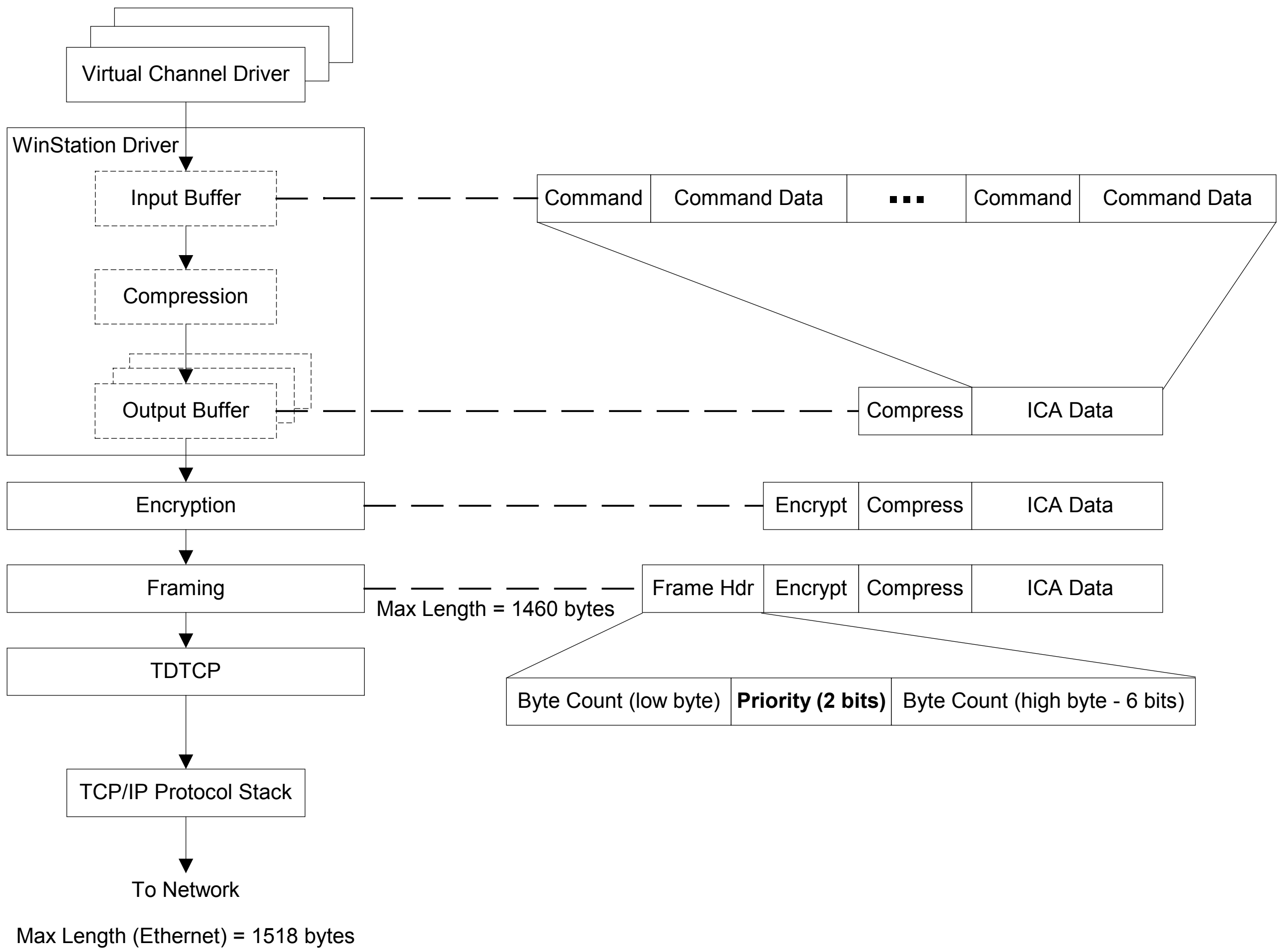
VirtualChannelName is the standard virtual channel abbreviation as specified in the above table. *VirtualChannelName* must be 7 characters, so trailing spaces must be added before the comma when necessary. *Priority* is one of the following numeric priority values: 0, 1, 2, 3.

The ThinWire virtual channels (CTXTW and CTXTWI) are the only high priority virtual channels by default, thus ensuring that time-sensitive user interface data is sent ahead of all other data.

ICA Data Transmission

The implementation details of ICA Priority Packet Tagging are better understood by examining the different layers of the ICA protocol and how the ICA protocol interacts with TCP/IP to send ICA data over an Ethernet network. The priority bits used for ICA Priority Packet Tagging are determined and set within this data transmission process.

The following diagram depicts the flow of ICA data through each protocol layer as it is generated by the client application (or server) and packaged for delivery to a server (or client application) over a TCP/IP network:



ICA data travels through the same protocol layers but in the reverse direction when received at the destination (client or server). All ICA protocol layers reside at the Presentation layer of the OSI networking model (Refer to *Appendix B: OSI Networking Model* for additional information about the layers of the OSI networking model). The ICA protocol layers depicted in the above diagram are described further in the following sections.

Virtual Channel Driver

Each virtual channel has its own virtual channel driver that sends virtual channel data to the WinStation driver (described in the following section). The format of the virtual channel data is not standardized as it depends completely on the virtual channel implementation.

WinStation Driver

The WinStation driver receives ICA virtual channel data from multiple virtual channel drivers and packages the data for receipt by lower network layers. The WinStation driver works at the Application, Presentation, and Session layers of the OSI networking model. The WinStation driver performs the following functions:

- Establishes the ICA session between the client and the server, and maintains session information such as whether compression and encryption are turned on, and whether ICA Priority Packet Tagging will be used.
- Encodes ICA command information and transforms input virtual channel data into ICA packets, which are placed in the WinStation driver's input buffer. An ICA packet consists of a single command byte followed by optional command data as shown below:

Command	Command Data

  An ICA packet is not required to contain command data, and therefore may only contain a single command byte. An ICA packet contains data from only one virtual channel. The maximum length of a single ICA packet cannot exceed 2048 bytes (2 KB).
- Compresses the ICA packets (when compression is turned on).
- Combines or separates compressed ICA packets (or uncompressed ICA packets if compression is not being used) into an available output buffer. The WinStation driver determines the amount of data to include in each output buffer so that the length of the ICA data when leaving the framing protocol driver does not exceed 1460 bytes (to keep ICA data from being broken up when transmitted by TCP/IP).
- Appends a compression header to the beginning of the output buffer (when compression is turned on).
- Determines the priority of each output buffer based on the virtual channel from where the data originated, and passes this information to the framing protocol driver. When multiple ICA packets are combined into one output buffer, the WinStation driver determines the priority of the output buffer based on the highest priority ICA packet included. For example, if the output buffer contains Thinwire (priority 0) and printing (priority 3) ICA packets, the output buffer is given a priority of 0 based on the included Thinwire data.
- Forwards the output buffer to the encryption protocol driver (when encryption is turned on).

Encryption Protocol Driver

When encryption is turned on, the encryption protocol driver adds an encryption header to the output buffer data passed from the WinStation driver. All data after the encryption header is encrypted, including the compression header (if included).

Framing Protocol Driver

The framing protocol driver calculates the byte count of the output buffer and adds a framing header. In addition to the byte count, the framing header includes a two-bit priority value as determined by the WinStation driver. For example, if the total byte count of the output buffer is 1320 bytes and the packet is high priority, the binary value of the framing header is as follows:

00	000101	00101000
Priority Bits	High Order Byte Count	Low Order Byte Count

The low order and high order bytes are reversed for network transmission, and the framing header is created as follows:

00101000	00	000101
Low Order Byte Count	Priority Bits	High Order Byte Count

The framing header described above was first introduced with ICA Priority Packet Tagging functionality in MetaFrame 1.8 Feature Release 1 and is also included with MetaFrame XP. Earlier versions of MetaFrame use a framing header that does not contain the two priority bits. All 16 bits of the framing header were used for the byte count. Since the byte count will never exceed 2048 bytes (the 2 KB limit for ICA packets), the top two bits will always be zero when an earlier version of the framing header is used. When an ICA session is established between servers running MetaFrame 1.8 Feature Release 1 or later and clients running previous versions of Program Neighborhood (or vice versa), ICA Priority Packet Tagging is not used and all ICA traffic within the established ICA session will not contain the two priority bits.

TDTCP

The ICA protocol transfers control to the TCP/IP protocol stack through TDTCP, the TCP transport driver. TDTCP is ICA's (and RDP's) interface to the TCP/IP protocol stack. TDTCP does not append any additional header or trailer information to the ICA data.

TCP/IP

Once TDTCP transfers control to the TCP/IP protocol stack, the TCP/IP protocol drivers prepare the ICA data for network transmission. Detailed information on the TCP/IP standards and how TCP/IP encapsulates data for network transmission can be found in the Request for Comments (RFC) and Standards (STD) documents available on the Internet (http://www.faqs.org/).

Quality of Service Solutions

Quality of Service (QoS) solutions are designed to prioritize ICA traffic against all other traffic on the network. These solutions are able to identify network traffic as ICA traffic either based on the TCP port (1494 by default) or by identifying the ICA initialization handshake that occurs when a new session is established (this is safer than using the TCP port, since the TCP port number is configurable). Some QoS solutions can also identify ICA traffic based on other information, such as published application or source IP address. This identification allows ICA sessions to be prioritized against each other across the entire network. For example, all ICA sessions where users are running a business critical application such as Peoplesoft can be given a higher priority than sessions performing functions that are not as business critical.

ICA Priority Packet Tagging provides QoS solutions with the opportunity to identify virtual channel priorities within an ICA session so that ICA sessions transmitting higher priority data are delivered first. ICA Priority Packet Tagging requires that the following considerations be addressed when used in combination with a QoS solution:

- TCP and IP are stream-oriented protocols. When ICA data is received by TCP and then by IP, it may be combined or broken up differently than how it was packaged by the ICA protocol drivers. The ICA output buffers are specifically limited to 1460 bytes so that they remain intact when delivered to the TCP/IP protocol stack. However, it is not guaranteed that the output buffers will remain intact. Therefore, the priority bits in the ICA framing header may not always be in the same place in the TCP segment or IP packet. This prevents QoS solutions from relying on a data offset to identify the priority bits at the TCP or IP layers. To circumvent this potential issue, QoS solutions must verify that the byte count in the header information of the TCP and IP layers matches the byte count in the first two bytes of the ICA data (when aligned correctly, these first two bytes will include the priority bits and the byte count of the ICA framing header). When the byte counts do not match, the ICA output buffers are most likely not intact within the TCP segments; therefore, the first two bits of ICA data in the IP packet should not be interpreted as priority bits.
- ICA Priority Packet Tagging is implemented at the Presentation layer (the sixth layer of the OSI networking model). Most routers read data at lower layers (layers two through four). Therefore, routers don't have access to the ICA Priority Packet Tagging information. When IP packets are sent through a router, the packets may be fragmented. If this is the case, the first packet will contain the framing header, including the priority bits and a now incorrect byte count (since the packet has been fragmented). Subsequent packet fragments will not have a framing header and thus will not include the priority bits (or a byte count). Therefore, if QoS solutions receive the ICA traffic after fragmentation by a router, not all IP packets will have the priority bits. Verifying the byte counts between the IP layer and the ICA framing header as described above ensures that the priority bits are interpreted correctly.
- TCP requires an acknowledgement of receipt for each TCP segment in the TCP buffer before sending additional segments. This prevents QoS solutions from being able to implement functionality that holds back printing ICA data and forwards on Thinwire ICA data within a single ICA stream (which is also a single TCP stream). TCP would report a failure of receipt for the TCP segments being held since they were not received by the destination in a timely manner. QoS solutions must implement ICA Priority Packet Tagging in such a way that the transmission speed of each TCP stream is dynamically altered based on the priority bits of the ICA data being transmitted, instead of attempting to hold back individual pieces of data within the stream.
- Program Neighborhood clients and MetaFrame servers running a software version prior to MetaFrame 1.8 Feature Release 1 will establish ICA sessions without ICA Priority Packet Tagging. Unless QoS solutions detect the Citrix software version in use by the ICA session, all ICA traffic in these sessions will be treated as high priority (priority 0) because the two bits that are now used for ICA Priority Packet Tagging were not used (and thus set to 0) in previous versions of MetaFrame.

Packeteer, Inc., a leading provider of application performance infrastructure systems, has released a QoS solution that takes advantage of ICA Priority Packet Tagging. Packeteer's PacketWise 5.1 release is incorporated in the PacketShaper and AppVantage product lines. For additional information about Packeteer's QoS solutions, visit www.packeteer.com.

Summary

ICA Priority Packet Tagging provides a mechanism for prioritizing ICA sessions based on the virtual channel from which the data originated. The implementation of ICA Priority Packet Tagging is best understood after examining how ICA data is packaged for transmission across an Ethernet network using TCP/IP. QoS solutions that take advantage ICA Priority Packet Tagging will provide QoS benefits that are more granular than prioritizing ICA traffic based only on application name or username.

Appendix A: ICA Protocol Driver File Locations

The following table lists the associated client and server file locations for each ICA protocol driver.

Protocol Driver	**Client File** **(C:\Program Files\Citrix\ICA Client)**	**Server File** **(WINNT/System32/Drivers)**
WinStation	wfica32.exe	wdica.sys icareduc.sys (Compression)
Encryption	pdc0N.dll (basic) pdc40N.dll (40-bit) pdc56N.dll (56-bit) pdc128N.dll (128-bit)	pdcrypt1.sys (Basic) pdcrypt2.sys (40-,56- and 128-bit)
Framing	pdframen.dll (Win32 platform) pdframew.dll (Win16 platform)	pdrframe.sys

Appendix B: OSI Networking Model

The Open System Interconnection (OSI) model is a seven-layer architectural standard for networks. A basic overview is provided here as a reference only.

Layer	Protocol Data Unit	Functionality	Examples
Application		Program-to-program communications such as file, print, database and other application services	WWW, E-Mail Gateways
Presentation		Data conversion, compression, decompression, encryption, decryption	ICA
Session		Creating, managing, and tearing down communications sessions by using simplex, half-duplex, and full-duplex modes	RPC, X Windows
Transport	Segment	Segments and reassembles data into a data stream; end-to-end data transport services; port numbers	TCP, UDP, SPX
Network	Packet	Routes data packets	IP, Routers
Data Link	Frame	Contains Logical Layer Control (LLC – flow control and timing) and Media Access Control (MAC – physical address)	NICs, Bridges, Switches, VLANs
Physical	Bit	Sends and receives bits	Cabling, Hubs

6400 NW 6th Way Fort Lauderdale, FL 33309 954-267-3000 http://www.citrix.com