Converting to INTERFACE Statement for OSA Port Interfaces in z/OS
Loss of Expected SOURCEVIPA

This document describes a common problem and its solution when converting from the traditional DEVICE/LINK coding for an IPv4 QDIO interface to the more current INTERFACE statement. It also provides information on determining the priority sequence for Source IP selection and how-to guidance for the conversion.

PROBLEM: I have converted my IPv4 DEVICE/LINK statements to the recommended INTERFACE statements and now my old SOURCEVIPA coding is no longer working. I am obtaining an entirely different source IP address from what I was expecting prior to the migration. How do I establish the correct SOURCEVIPA for my new INTERFACE statements?

EXPLANATION: When converting QDIO definitions from the z/OS DEVICE/LINK format to the INTERFACE format, you may see that you are no longer setting the expected SOURCEVIPA value that you had established with the HOME list coding for the old DEVICE and LINK. The new INTERFACE statement -- if used for connections originating at z/OS – no longer adopts the source IP value that you were accustomed to. This occurs, because the HOME list coding that establishes the SOURCEVIPA for DEVICE/LINK is no longer necessary for an INTERFACE statement.

SOLUTION: For an INTERFACE statement, the way to identify the VIPA to be used for TCP, UDP, ICMP, and other types of connection requests originating in z/OS is to add the SOURCEVIPAINTERFACE parameter to the statement while continuing to preserve the SOURCEVIPA coding in the IPCONFIG statement of the TCP/IP profile. An example follows:

1) IPCONFIG SOURCEVIPA
   • Note: This enables SOURCEVIPA globally.
     i. For DEVICE/LINK it allows the position of a static VIPA in the HOME list to determine the value of the source IP address for outbound-originating connections.

2) INTERFACE OSAD001
   DEFINE IPAQENET
   PORTNAME OSAE001
   SOURCEVIPAINTERFACE VIPAL1 <<<<<<<<<<<<<<<<<<<
   MTU 8992
   IPADDR 192.168.1.100/24
   VMAC ROUTEALL
   INBPERF DYNAMIC WORKLOADQ**1

1 Investigate to what extent you wish to deploy the INBPERF parameter and whether the OSA MCL level supports the various subparameters. See the IP Configuration Reference (SC31-8776) for details.

Author: Gwendolyn J. Dente, IBM ATS (gdente@us.ibm.com)
Migration_OSA_InterfaceSOURCEVIPAINT_FLASH_01.doc
© Copyright IBM Corporation, 2011
IMPORTANT NOTES:
Source IP Address Selection:
There are other ways to establish a Source IP address for connections originating at z/OS. Other methods may override your SOURCEVIPAINTEFTERFACE coding. Therefore, it is important to understand the precedence for Source IP address selection as described in the IP Configuration Guide (SC31-8775), particularly in the manual for z/OS V1R12, which includes significant documentation enhancements. The precedence described in the V1R12 manual is also valid for the V1R10 and the V1R11 versions of z/OS Communications Server. Consult the section named “Source IP address selection” in the IP Configuration Guide of V1R12 or a later release.

Procedure for Converting from DEVICE/LINK to INTERFACE:
The same IP Configuration Guide – beginning with the z/OS V1R12 version of the manual -- contains a step-by-step procedure for converting from the older DEVICE/LINK format to the newer, and recommended, INTERFACE format. Using the IP Configuration Guide at the V1R12 level or higher (SC31-8775), look for the section in the manual named “Consideration for networking hardware attachment.” Then look for the subordinate section called “Steps for converting from IPv4 IPAQENET DEVICE, LINK, and HOME definitions to the IPv4 IPAQENET INTERFACE statement” and follow the instructions there.

Implementing a non-disruptive change to the INTERFACE statement:
If you need to make a change to the INTERFACE statement in a non-disruptive – or nearly non-disruptive - fashion, you may do so with the OBEYFILE method described below:

1. Stop the interface (VARY TCPIP,,STOP,interfacename command).

2. Execute an OBEYFILE referencing a data set with the following statement:
   - INTERFACE interfacename DELETE
   - Note that you will get errors for any routes defined for this, you can ignore these.

3. Execute an OBEYFILE referencing a data set with just the updated
   - INTERFACE statement.

4. Start the interface (VARY TCPIP,,START,interfacename command).

If you execute these steps one at a time and you have sufficient routing definitions (or are using dynamic routing), it should not be disruptive to any connections using the VIPA. Those using the real address of the interface for the source IP will (likely) fail, although they might survive the timeout process if you can implement these steps listed above quickly enough.