User Experience: Running Multiple TCPIP Stacks on an IBM Mainframe LPAR

Thomas Y. Fitzpatrick II  
TGS Systems Inc.

March 10, 2014  
Session: 14992
Suggested Reasons for Running Multiple TCPIP Stacks in a LPAR

- Network Isolation for one or more of your applications.
  - Improved Diagnostics
  - Improved Tuning
- Isolate applications with multiple Open System Adapter (OSA) features.
- You may have a case for Separating a Piece of the Business.
- Create migration path to a new network.
Suggested Steps for Adding an Additional TCPIP stack

1. Decide on a stack naming convention and an associated stack configuration dataset naming convention.
2. Decide on network connections for the new stack. Remember that each stack is a separate TCP/IP host with its own network interface(s) and IP address(es).
3. Decide on a naming convention for your TCPPARMS library to use for explicitly allocated configuration data sets for these stacks, or create a new aaa.TCPPARMS library.
4. Update your SYS1.PARMLIB(BPXPRMxx).
5. Create a PROFILE2 member in aaa.TCPPARMS.
6. Create a TCPDAT2 member in aaa.TCPPARMS.
Suggested Steps for Adding an Additional TCPIP stack (continued)

• 7. Create a FTPDAT2 member in aaa.TCPPARMS
• 8. Create a new system started task JCL procedure.
• 9. Create application TCPIP connections required server address space JCL procedures with a SYSTCPD DD statement pointing to aaa.TCPPARMS(TCPDAT2).
• 10. Create server-specific configuration data sets, such as aaa.TCPPARMS(PROFTEL2).
• 11. Create required RACF definitions to assign started task user IDs to new address spaces.
• 12. If you are using a domain name server, ensure that it is updated with your new host name and address.
Suggested Steps for Adding an Additional TCPIP stack (continued)

• 13. Optionally create a REXX/CLlist program to switch TSO user’s stack affinity to the desired stack.

• 14. Depending on your system’s management strategy, you may optionally create a different USS table and different VTAM definitions to distinguish among the different stacks.
Stack Naming Conventions

- Decide on a stack naming convention with numbers and an associated stack configuration dataset naming convention with the same number.
- For Example: TCPIP2, OMPROUTE2
- aaa.TCPPARMS(TCPDAT2)
- aaa.TCPPARMS(FTPDAT2)
Decide on network connections for the new stack

- Decided on network connections for the new stack. Remember that each stack is a separate TCP/IP host with its own network interface(s) and IP address(es).
- Used OSA-Express3 Lx Feature
- Using Ports on two different OSAs for redundancy and will be using VIPA
OSA-Express3

- Need two CHPIPs per card
- Need two ports, from different cards for redundancy
- Need Features to match your Network
- Note how ports are numbered
- Picture from OSA-Express Customer Guide and reference
Example of OSA Express3 IOCP Deck Definitions

CHPID PATH=(CSS(0),99),SHARED,
PARTITION=((MVSA,MVSB,MVSC,MVSD,MVSE,VMPRD,VMST,VM1), (=)),PCHID=1D0,TYPE=OSD

CHPID PATH=(CSS(0),9D),SHARED,
PARTITION=((MVSA,MVSB,MVSC,MVSD,MVSE,VMPRD,VMST,VM1),(=)),PCHID=350,TYPE=OSD
Example of OSA Express3 IOCP Deck Definitions (continued)

CNTLUNIT
CUNUMBR=0010,PATH=((CSS(0),99)), CUADD=0, UNIT=OSA

IODEVICE
ADDRESS=(200,032),CUNUMBR=(0010),UNIT=OSA

IODEVICE
ADDRESS=(2FE,001),CUNUMBR=(0010),UNIT=OSAD

CNTLUNIT
CUNUMBR=0011,PATH=((CSS(0),99)), CUADD=1, UNIT=OSA
Example of OSA Express3 IOCP Deck Definitions (continued)

CNTLUNIT
CUNUMBR=000D,PATH=((CSS(0),9D)),CUADD=0,
UNIT=OSA
IODEVICE ADDRESS=(BE0,030),CUNUMBR=(000D),
UNIT=OSA
IODEVICE ADDRESS=(BFE,001),CUNUMBR=(000D),
UNIT=OSAD
CNTLUNIT
CUNUMBR=001D,PATH=((CSS(0),9D)),CUADD=1,
UNIT=OSA
IODEVICE ADDRESS=(1D0,016),CUNUMBR=(001D),UNIT=OSA
Naming convention for your TCPPARMS library

- Decided on a naming convention for your TCPPARMS library to use for explicitly allocated configuration data sets for these stacks, or create a new aaa.TCPPARMS library.

- Examples are:
  - aaa.TCPPARMS(TCPDAT2)
  - aaa.TCPPARMS(FTPDAT2)
  - aaa.TCPPARMS(OSPFCON2)
  - aaa.TCPPARMS(OSPFENV2)
  - aaa.TCPPARMS(PROFTEL2)
Update your SYS1.PARMLIB(BPXPRMxx)

SUBFILESYSTYPE NAME(TCPIP) /* Name of file system */
   TYPE(CINET) /* Type matching Cinet's TYPE */
   ENTRYPOINT(EZBPFINI) /* Entry point of load module */
   DEFAULT /* <- The Default Socket PFS */

- Depending on the number of stacks you want to run on the sockets interfaces, you can use Integrated Sockets AF-INET or Common INET. Integrated Sockets AF-INET supports one TCP/IP stack at a time. It is used when applications communicate through a single stack.
- CINET is used when applications communicate through multiple stacks, and its daemon uses a timeout when opening sockets.
Update your SYS1.PARMLIB(BPXPRMxx) (continued)

- Recommend defining more than two stacks, since IPL is needed for this update.

```plaintext
SUBFILESYSTYPE NAME(TCPIP2) /* Name of file system */ TYPE(CINET) /* Type matching Cinet's TYPE */ ENTRYPONT(EZBPFINI) /* Entry point of load module */

SUBFILESYSTYPE NAME(TCPIP3) /* Name of file system */ TYPE(CINET) /* Type matching Cinet's TYPE */ ENTRYPONT(EZBPFINI) /* Entry point of load module */
```
Create a PROFILE2 member in aaa.TCPPARMS

- Define VIPA and Interface Addresses.
- Keep Port Definitions the same as the Primary stack.
- Keep Port Ranges the same as the Primary stack.
- Update your AUTOLOG section with the servers you want using the new name from your naming convention.
- Account for your Hypersocket Definitions if you have them.
- Update the Start Statements at the end.
- Make sure not duplicate IP addresses
Profile statements sample

; VIPA Device and Link Statements (New IP Address)
;

DEVICE DEVVIP2A VIRTUAL 0
LINK LINKVIP2A VIRTUAL 0 DEVVIP2A
Profile statements sample (continued)

;

; MCPIPA for Gigabyte OSA Card - Note(AMQ6TRLE must be active in VTAM)
;

INTERFACE GLINK99A
   DEFINE IPAQENET
   INBPERF DYNAMIC
   IPADDR 153.30.167.21
   PORTNAME GPORT99D
   SOURCEVIP Interface LINKVIP2A
Profile statements sample (continued)

INTERFACE GLINK9DA
DEFINE IPAQENET
INBPERF DYNAMIC
IPADDR 153.30.167.22
PORTNAME GPORT9DD
SOURCEVIP INTERFACE LINKVIP2A
Profile statements sample (continued)

;  
; HiperSockets on CHPID E0  
;  
DEVICE IUTIQDE0 MPCIPA  
LINK  HIPERLE0 IPAQIDIO IUTIQDE0
Profile statements sample (continued)

; HOME Internet addresses of each link in the host.

; HOME

153.30.168.1   LINKVIP2A
169.254.0.20    HIPERLE0
Profile statements sample (continued)

; Start all the defined devices.

START GLINK99A
START GLINK9DA
START IUTIQDE0
Create a TCPDAT2 member in aaa.TCPPARMS

- TCPIPJOBNAME specifies the name of the started procedure. Default TCPIP.
- HOSTNAME specifies the TCP host name of this system. Default HOSTNAME will be the node name specified in the IEFSSNxx PARMLIB member.
- DOMAINORIGIN specifies the domain origin that will be appended to host names passed to the resolver.
- NSINTERADDR specifies the IP address of the name server.
Create a FTPDAT2 member in aaa.TCPPARMS

- Good News! This can look pretty much like the member for the Primary stack, but you do want to code a separate member.
- This FTP.DATA file is used to specify default file and disk parameters used by the FTP server.
Create a new system started task JCL procedure.

//PROFILE  DD
DISP=SHR, DSN=SYS1.TCPPARMS(PROFOSP2)

//PROFILE  DD
DISP=SHR, DSN=SYS1.TCPPARMS(PROFOSP)

//SYSTCPD  DD
DSN=SYS1.TCPPARMS(TCPDAT2), DISP=SHR

//SYSTCPD  DD
DSN=SYS1.TCPPARMS(TCPDATA), DISP=SHR
Address space JCL procedures with a SYSTCPD DD statement

• Create application TCPIIP connections required server address space JCL procedures with a SYSTCPD DD statement pointing to aaa.TCPPARMS(TCPDAT2).
Create server-specific configuration data sets

- Create server-specific configuration data sets, such as aaa.TCPPARMS(PROFTEL2).
TELNET ServerParms Sample

TELNETGLOBALS
   TCPIPJOBNAME TCPIP2
ENDTELNETGLOBALS
BEGINVTAM
   PORT 23
   LUGROUP SESSMGR
      APQ0Z501..APQ0Z999
ENDLUGROUP
TELNET Server Parms Sample (continued)

IPGROUP ALLUSERS
   0.0.0.0:0.0.0.0
ENDIPGROUP
DEFAULTAPPL APQ0SMGR
LUMAP SESSMGR ALLUSERS
ALLOWAPPL *
ENDVTAM
Create required RACF definitions

- Create required RACF definitions to assign started task user IDs to new address spaces
- TCPIP2
- FTPE
- OMPROUTE2
- TN32702
Domain Name Server

• If you are using a domain name server, ensure that it is updated with your new host name and address.
Program to switch TSO user’s stack affinity

- Optionally create a REXX/CList program to switch TSO user’s stack affinity to the desired stack.
Sample Rexx to switch to TCPIP2 stack

SAY 'SWITCHING TO TCPIP2 STACK'
MSGSTAT = MSG()
Z = MSG("OFF")
"FREE FI(SYSTCPD)"
"FREE FI(SYSFTPD)"
"ALLOC FI(SYSTCPD) DA('aaa.TCPPARMS(TCPDAT2)') SHR"
"ALLOC FI(SYSFTPD) DA('aaa.TCPPARMS(FTPDAT2)') SHR"
Z = MSG(MSGSTAT)
EXIT(0)
Different VTAM definitions to distinguish among the different stacks

• Depending on your system’s management strategy, you may optionally create a different USS table and different VTAM definitions to distinguish among the different stacks.
• An example would be LUs used for Telnet sessions.
Sample of CICS TCPIPSERVICES listen on multiple stacks

DEFINE TCPIPSERVICE(JGATEWAY)
GROUP(PCA2JGAT)
DESCRIPTION(ECI TCPIPSERVICE)
  PORTNUMBER(4084) STATUS(OPEN)
  PROTOCOL(ECI) TRANSACTION(CIEP)
  BACKLOG(5) IPADDRESS(ANY) SOCKETCLOSE(NO)
  SSL(NO)
  ATTACHSEC(VERIFY) GRPCRITICAL(NO)
MVS Sockets for CICS listen on single stack

REGION2 - MVS SOCKETS FOR CICS REGION
EZACICD TYPE=CICS, CICS RECORD DEFINE
APPLID=APQ0PCA2, APPLID OF CICS RGN
TCPADDR=TCPIP2, TCPIP JOBNAME
CACHMIN=15, MIN CACHE RFRSH TIME
CACHMAX=30, MAX CACHE RFRSH TIME
CACHRES=10, # of resolvers
TCBLIM=0, Open API TCB limit
OTE=NO, Open Trans Env
TRACE=NO, No CICS trace recs
SMSGSUP=NO start msgs not supresd
MVS Sockets for CICS listen on single stack (continued)

REGION2 - MVS SOCKETS FOR CICS CSKL LISTENER
EZACICD TYPE=LISTENER, LISTENER
FORMAT=STANDARD, STANDARD LISTENER
APPLID=APQ0PCA2, APPLID OF CICS RGN
TRANID=CSKL, LISTENER TRANID
PORT=4057, USE PORT NUMBER
AF=INET, Lstnr addr family
IMMED=YES, START IMMEDIATE
BACKLOG=40, BACKLOG VALUE
NUMSOCK=200, SUPPORT CONCURRENT
MVS Sockets for CICS listen on single stack (continued)

MINMSGL=4, MIN INPT MSG LEN
ACCTTIME=60, TIMEOUT VALUE
GIVTIME=30, GIVESOCKET TIME
REATIME=30, READ TIMEOUT
RTYTIME=15, Stay with TCPIP
LAPPLD=NO, READ TIMEOUT
TRANTRN=YES, READ TIMEOUT
TRANUSR=NO, READ TIMEOUT
SECEXIT=EZACICSE SECURITY EXIT
Sample Commands for TCPIP2 stack

- DISPLAY TCPIP,TCPIP2,NETSTAT,ALLCONN
- DISPLAY TCPIP,TCPIP2,NETSTAT,PORTLIST
- VARY TCPIP,TCPIP2,OBEYFILE,SYS1.TCPPARMS(ADDPOR3)
Conclusion

• Follow these guidelines to start multiple stacks in a LPAR
• Know the limitations of your applications.
• There is no place like 127.0.0.1. This may be important if you are depending on the local address.
• The first stack initialized is the primary stack, with the ones afterword being secondary.
Thank you for Attending

- You can contact me at tfitzpatricknj@yahoo.com