



z/VM Simplified Network Configuration

Miguel Delapaz
z/VM TCP/IP Development
Monday, 21 May 2007

In this session we will cover two tools designed to make configuring z/VM TCP/IP easier. The IPWIZARD function that allows you to quickly and easily perform the base configuration as you initially try and get TCP/IP running. The IFCONFIG command allows you to quickly and easily add new connections to your running TCP/IP stack. These functions mean that you can get up and running quickly without have to learn the format of the z/VM TCP/IP configuration files. The IFCONFIG command allows you to display information about and make temporary dynamic changes to the TCP/IP configuration without stopping and restarting the TCPIP virtual machine. The command syntax is very similar to that of Linux, making skills more transferable.

References to IBM products, programs, or services do not imply that IBM intends to make these available in all countries in which IBM operates. Any reference to an IBM product, program, or service is not intended to state or imply that only IBM's product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe on any of the intellectual property rights of IBM may be used instead. The evaluation and verification of operation in conjunction with other products, except those expressly designed by IBM, are the responsibility of the user.

The following terms are trademarks of IBM Corporation in the United States or other countries or both:

IBM z/VM System z HiperSockets

UNIX is a registered trademark in the United States and other countries licensed exclusively through X/Open Company Limited.

Linux is a trademark of Linus Torvalds in the United States, other countries, or both.

Other company, product, and service names, which may be denoted by double asterisks (**), may be trademarks or service marks of others.

© Copyright International Business Machines Corporation, 2004-2007

Agenda

- **Discuss tools available in z/VM to simplify network configuration tasks**
 - **Generate initial configuration files**
 - **Modify a running stack without bringing it down**

The Problem

- **Format of z/VM configuration files differs from that of Linux[®]/Unix systems**
- **Growth of Linux on System z means many Linux sys-admins must begin to learn about z/VM**
- **Need to address learning curve issues**

Solution #1: The **IPWIZARD** Command

What You Need

- **A fresh z/VM TCP/IP stack**
 - **User must be previously defined**
 - **IPWIZARD is NOT a migration tool**
 - **IPWIZARD creates a initial/minimal setup**
- **Access to MAINT's 193 disk**
- **Write access to the TCP/IP stack's 592 & 198 disks**
- **A full screen 3270 device**
 - **IPWIZARD does not support line mode**
- **A completed “Basic IP Connectivity Worksheet”**
 - **See the “Plan Your Installation” chapter in “z/VM Guide for Automated Installation and Service”**

What You Don't Need

- **Knowledge or experience with z/VM TCP/IP configuration file formats**

Host Configuration Panel

*** z/VM TCP/IP Configuration Wizard ***

The items that follow describe your z/VM host

User ID of VM TCP/IP Stack Virtual Machine: TCPIP06_

Host Name: MIGUELD_____

Domain Name: IBM.COM_____

Gateway IP Address: 10.10.0.1_____

DNS Addresses:

1) 10.10.0.2_____

2) _____

3) _____

PF1 = HELP PF3 = QUIT PF8 = Continue ENTER = Refresh

IPv4 Interface Configuration Panel

*** General Interface Configuration Panel ***

Interface Name: INT0_____ Device Number: 4712

IP Address: 10.10.0.6_____

Subnet Mask: 255.255.255.0__

Interface Type (Select one):

<input checked="" type="checkbox"/>	QDIO	<input type="checkbox"/>	LCS	<input type="checkbox"/>	HiperSockets
<input type="checkbox"/>	CLAW	<input type="checkbox"/>	CTC		

PF1 = HELP PF3 = QUIT PF7 = Backward PF8 = Continue ENTER = Refresh

QDIO Configuration Panel (IPv4)

*** QDIO Interface Configuration Panel ***

Network Type (Select one):

Ethernet Token Ring

Port Name (optional): _____

Router Type (Select one):

Primary Secondary None

Maximum Transmission Unit (MTU) size: 1500_

PF1 = HELP PF3 = QUIT PF5 = Process PF7 = Backward ENTER = Refresh

LCS Configuration Panel

*** LCS Interface Configuration Panel ***

Network Type (Select one):

X Ethernet Token Ring FDDI

Port/Adapter Number 3

Maximum Transmission Unit (MTU) size: 1500

PF1 = HELP PF3 = QUIT PF5 = Process PF7 = Backward ENTER = Refresh

HiperSocket Configuration Panel (IPv4)

*** HiperSockets Interface Configuration Panel ***

Maximum Frame Size (MFS): 64 K

PF1 = HELP PF3 = QUIT PF5 = Process PF7 = Backward ENTER = Refresh

CLAW Configuration Panel

*** CLAW Interface Configuration Panel ***

The items that follow must match the values configured on the CLAW device.

CLAW Host Name: CLAWHOST

CLAW Adapter Name: CLAWADAP

Maximum Transmission Unit (MTU) size: 4096

PF1 = HELP PF3 = QUIT PF5 = Process PF7 = Backward ENTER = Refresh

CTC Configuration Panel

*** CTC Interface Configuration Panel ***

Write Channel Device Number (Select one):

X 4712 _ 4713

Maximum Transmission Unit (MTU) size: 32760

Peer IP Address: 10.10.0.5_____

PF1 = HELP PF3 = QUIT PF5 = Process PF7 = Backward ENTER = Refresh

Initial Panel - Revisited

```
*** z/VM TCP/IP Configuration Wizard ***
```

The items that follow describe your z/VM host

User ID of VM TCP/IP Stack Virtual Machine: TCPIP06_

Host Name: MIGUELD_____

Domain Name: IBM.COM_____

Gateway IP Address: 50C0:C2C1:1010::1_____

DNS Addresses:

- 1) _____
- 2) _____
- 3) _____

PF1 = HELP PF3 = QUIT PF8 = Continue ENTER = Refresh

IPv6 Interface Configuration Panel

*** General Interface Configuration Panel ***

Interface Name: INT0 Device Number: 4712

IP Address: 50C0:C2C1:1010::6

Prefix Length: 64_

Interface Type (Select one):

QDIO HiperSockets

PF1 = HELP PF3 = QUIT PF7 = Backward PF8 = Continue ENTER = Refresh

QDIO Configuration Panel (IPv6)

*** QDIO Interface Configuration Panel ***

Port Name (optional): _____

Router Type (Select one):

Primary Secondary None

Maximum Transmission Unit (MTU) size: 1500_

Send Router Advertisements (Select One): On Off

PF1 = HELP PF3 = QUIT PF5 = Process PF7 = Backward ENTER = Refresh

HiperSockets Configuration Panel (IPv6)

*** HiperSockets Interface Configuration Panel ***

Maximum Frame Size (MFS): 64 K

Send Router Advertisements (Select One): On Off

PF1 = HELP PF3 = QUIT PF5 = Process PF7 = Backward ENTER = Refresh

IPWIZARD Panels

- **Input all required fields**
 - You will be prompted if you miss one
- **Press PF5 to process your input**
- **The following files are created for you:**
 - **On TCPMAINT's 198 disk:**
 - PROFILE TCPIP
 - SYSTEM DTCPARMS
 - **On TCPMAINT's 592 disk:**
 - TCPIP DATA
- **Basic connectivity tests performed**

Generated PROFILE TCPIP

```
ASSORTEDPARMS
PROXYARP
ENDASSORTEDPARMS
; -----
OBEY
OPERATOR TCPMAINT MAINT MPROUTE DHCPD REXECD SNMPD SNMPQE TCPMNT06
ENDOBEY
; -----
PORT
  23    TCP INTCLIEN          ; TELNET Server
; -----
DEVICE DEV@4712  OSD 4712  PRIROUTER
LINK INTO QDIOETHERNET DEV@4712  MTU 1500
; -----
HOME
10.10.0.6 255.255.255.0 INTO
; -----
GATEWAY
DEFAULTNET          10.10.0.1          INTO          1500
; -----
START DEV@4712
```

Generated SYSTEM DTCPARMS

```
:nick.TCPIP06 : type.server  
              : class.stack  
              : attach.4712-4714
```

Generated TCPIP DATA

```
TCPIPUSERID TCPIP06
```

```
;
```

```
HOSTNAME MIGUELD
```

```
;
```

```
DOMAINORIGIN IBM.COM
```

```
;
```

```
NSINTERADDR 10.10.0.2
```

Solution #2: The **IFCONFIG** Command

OBEYFILE: It Looked Like Such a Nice Toy...

- **OBEYFILE can be used to make dynamic changes to statements configured in PROFILE TCPIP**
- **It is not necessary to recycle your TCP/IP server to pick up these changes**
- **Usable by anyone in the TCP/IP server's OBEY list**

...Until I Tried to Use It

- **Identical syntax to PROFILE TCPIP**
 - **Not good for a Unix/Linux user**
- **Some statements require full replacement**
 - **GATEWAY**
 - **HOME**
- **Need to issue CP commands to inform stack of new devices prior to issuing OBEYFILE**
 - **The CP commands differ for Real and Virtual devices**

The OBEYFILE Process - “Simplified”

- Issue CP ATTACH to attach a new device to your stack
- Copy your current configuration file (PROFILE TCPIP) to a new file (e.g. NEWLINK TCPIP A)
- Delete everything but the HOME and GATEWAY statements
- Add new DEVICE and LINK statements at the top
- Insert new HOME and GATEWAY entries
- Add new START statement at the bottom
- Issue OBEYFILE NEWLINK TCPIP

The IFCONFIG “Process”

- Issue IFCONFIG command with the appropriate parameters
- Go get a cup of coffee

IFCONFIG — Goodies

- Can be used to display device information, define a new device or modify an existing device
- Most device types are supported
- Command syntax is similar to Linux/Unix
 - Keyword driven
 - Command options are preceded by a —
- It can create Guest LANs
 - Specify a LAN that doesn't exist when creating a virtual device

IFCONFIG — Gotchas

- **Changes made are NOT permanent!**
 - **An IPL will revert to configuration in PROFILE TCPIP**
 - **Commands could be issued by a service machine at IPL to setup network environment**
- **Does not fully support dynamic routing (MPRoute)**
 - **Queries work fine**
 - **Adding or modifying interfaces requires more work on your part**

Useful IFCONFIG Options

■ —SHOW

- Displays the TCP/IP server configuration file statements that are required to make the changes specified by the command, but does NOT change the running system

■ —VERBOSE

- Displays any CP or NETSTAT commands that are used while changing the running system

■ —FORCE

- Specifies that IFCONFIG should ignore the sense data returned from the device when attempting to create a new interface

More Useful IFCONFIG Options

- **—MDISKPW *password***
 - Specifies the read password for the 'A' disk of the user issuing the IFCONFIG command
- **—TCP *userid***
 - Directs the IFCONFIG command to the specified TCP/IP server
- **—ALL**
 - Displays configuration information for all interfaces, regardless of their status
- **—REMOVE (New in z/VM 5.3.0)**
 - Deletes an interface from the TCP/IP stack's configuration

Displaying Interfaces With IFCONFIG

ifconfig

```
ETH0      inet addr: <NONE> mask: ?  
          UP BROADCAST MULTICAST MTU: 1500  
          vdev: 3300 type: QDIO ETHERNET portname: UNASSIGNED  
          ipv4 router type: NONROUTER  
          ipv6 router type: NONROUTER ipv6: ENABLED  
          LAN owner: TCPIP06 name: LAN1  
          cpu: 0 forwarding: ENABLED  
          RX bytes: 0 TX bytes: 816  
          IPv6 Addresses:  
            10:0:0:0:0:0:0:1  
            FE80:0:0:0:0:209:5700:100:3D  
Ready; T=0.04/0.05 10:37:28
```

ifconfig eth1

```
ETH1      inet addr: 10.10.0.1 mask: 255.255.255.0  
          DOWN MTU: 4000  
          vdev: FF00 type: HIPERS  
          ipv6: DISABLED  
          LAN owner: TCPIP06 name: LAN2  
          cpu: 0 forwarding: ENABLED  
          RX bytes: 0 TX bytes: 0  
Ready; T=0.04/0.05 10:37:38
```

Displaying All Interfaces

```
ifconfig -all
```

```
ETH0      inet addr: <NONE> mask: ?  
          UP BROADCAST MULTICAST MTU: 1500  
          vdev: 3300 type: QDIO ETHERNET portname: UNASSIGNED  
          ipv4 router type: NONROUTER  
          ipv6 router type: NONROUTER ipv6: ENABLED  
          LAN owner: TCPIP06 name: LAN1  
          cpu: 0 forwarding: ENABLED  
          RX bytes: 0 TX bytes: 816  
          IPv6 Addresses:  
            10:0:0:0:0:0:0:1  
            FE80:0:0:0:209:5700:100:3D  
  
ETH1      inet addr: 10.10.0.1 mask: 255.255.255.0  
          DOWN MTU: 4000  
          vdev: FF00 type: HIPERS  
          ipv6: DISABLED  
          LAN owner: TCPIP06 name: LAN2  
          cpu: 0 forwarding: ENABLED  
          RX bytes: 0 TX bytes: 0  
Ready; T=0.05/0.07 10:39:05
```

Creating a Virtual HiperSockets Device

```
ifconfig vhsi0 10.2.0.1/25 vhs 3904 system lantest mtu 8192 -v -s
* NETSTAT TCP TCPIP06 CP DEFINE LAN LANTEST OWNER SYSTEM TYPE HIPERS MFS 16K
* NETSTAT TCP TCPIP06 CP DEFINE NIC 3904 HIPERS
* NETSTAT TCP TCPIP06 CP COUPLE 3904 SYSTEM LANTEST
; Generated by <IFCONFIG vhsi0 10.2.0.1/25 vhs 3904 system lantest mtu 8192 -v
; -s>
; 4 Aug 2006 10:42:21
DEVICE DEV@3904 HIPERS 3904
LINK VHSIO QDIOIP DEV@3904 MTU 8192
HOME
10.2.0.1 255.255.255.128 VHSIO
START DEV@3904
Ready; T=0.05/0.06 10:42:21
```

Creating a Virtual CTC Device

```
ifconfig vctc0 10.14.6.1/30 vctc 800 laplace1 3600 ptp 10.14.6.2 portnumber 0 -s -v
* NETSTAT TCP TCPIP06 CP DEFINE 3088 800
* NETSTAT TCP TCPIP06 CP DEFINE 3088 801
* NETSTAT TCP TCPIP06 CP COUPLE 800 LAPLACE1 3600
* NETSTAT TCP TCPIP06 CP COUPLE 801 LAPLACE1 3601
; Generated by <IFCONFIG vctc0 10.14.6.1/30 vctc 800 laplace1 3600 ptp 10.14.6.2
; portnumber 0 -s -v>
; 31 Jan 2007 13:50:32
DEVICE DEV@800 CTC 800
LINK VCTC0 CTC 0 DEV@800 MTU 0
HOME
10.14.6.1 255.255.255.252 VCTC0
GATEWAY
10.14.6.2 HOST = VCTC0 0
START DEV@800
Ready; T=0.02/0.02 13:50:32
```

Modifying an Interface

```
ifconfig eth1
```

```
ETH1      inet addr: 10.10.0.1 mask: 255.255.255.0  
          UP BROADCAST MULTICAST MTU: 4000  
          vdev: FF00 type: HIPERS  
          ipv6: DISABLED  
          LAN owner: TCPIP06 name: LAN2  
          cpu: 0 forwarding: ENABLED  
          RX bytes: 0 TX bytes: 0
```

```
Ready; T=0.06/0.07 11:09:17
```

```
ifconfig eth1 mask 255.255.255.128 mtu 1500
```

```
Ready; T=0.12/0.14 11:09:28
```

```
ifconfig eth1
```

```
ETH1      inet addr: 10.10.0.1 mask: 255.255.255.128  
          UP BROADCAST MULTICAST MTU: 1500  
          vdev: FF00 type: HIPERS  
          ipv6: DISABLED  
          LAN owner: TCPIP06 name: LAN2  
          cpu: 0 forwarding: ENABLED  
          RX bytes: 0 TX bytes: 0
```

```
Ready; T=0.06/0.07 11:09:32
```

Bringing an Interface UP or DOWN

```
ifconfig eth1
```

```
ETH1      inet addr: 10.10.0.1 mask: 255.255.255.128  
          UP BROADCAST MULTICAST MTU: 1500  
          vdev: FF00 type: HIPERS  
          ipv6: DISABLED  
          LAN owner: TCPIP06 name: LAN2  
          cpu: 0 forwarding: ENABLED  
          RX bytes: 0 TX bytes: 0
```

```
Ready; T=0.06/0.07 11:12:55
```

```
ifconfig eth1 down
```

```
Ready; T=0.12/0.14 11:13:15
```

```
ifconfig eth1
```

```
ETH1      inet addr: 10.10.0.1 mask: 255.255.255.128  
          DOWN MTU: 1500  
          vdev: FF00 type: HIPERS  
          ipv6: DISABLED  
          LAN owner: TCPIP06 name: LAN2  
          cpu: 0 forwarding: ENABLED  
          RX bytes: 0 TX bytes: 0
```

```
Ready; T=0.06/0.07 11:13:18
```

```
ifconfig eth1 up
```

```
Ready; T=0.12/0.14 11:13:22
```

```
ifconfig eth1
```

```
ETH1      inet addr: 10.10.0.1 mask: 255.255.255.128  
          UP BROADCAST MULTICAST MTU: 1500  
          vdev: FF00 type: HIPERS  
          ipv6: DISABLED  
          LAN owner: TCPIP06 name: LAN2  
          cpu: 0 forwarding: ENABLED  
          RX bytes: 0 TX bytes: 0
```

```
Ready; T=0.06/0.07 11:13:23
```

IP Version 6 Support

- **Support for creating and displaying IPv6 interfaces**
 - **Dual protocol (IPv4/IPv6) support also available**
- **Support for multiple IP addresses per interface**
 - **Displaying all IP addresses for an interface**
 - **Adding or deleting IP addresses to or from an interface**

Creating an IPv6 Interface

```
ifconfig eth2 50c0:c2c1:1010::6/64 veth fe00 tcpip06 lan1 -v -s
* NETSTAT TCP TCPIP06 CP DEFINE NIC FE00 QDIO
* NETSTAT TCP TCPIP06 CP COUPLE FE00 TCPIP06 LAN1
; Generated by <IFCONFIG eth2 50c0:c2c1:1010::6/64 veth fe00 tcpip06 lan1 -v -s>
; 4 Aug 2006 10:43:26
DEVICE DEV@FE00  OSD FE00
LINK ETH2 QDIOETHERNET DEV@FE00  MTU 0  ENABLEIPV6
HOME
50C0:C2C1:1010:0:0:0:0:6                ETH2
START DEV@FE00
Ready; T=0.05/0.06 10:43:26
```

Adding An IP Address

```
ifconfig eth0
```

```
ETH0      inet addr: <NONE> mask: ?  
          UP BROADCAST MULTICAST MTU: 1500  
          vdev: 3300 type: QDIO ETHERNET portname: UNASSIGNED  
          ipv4 router type: NONROUTER  
          ipv6 router type: NONROUTER ipv6: ENABLED  
          LAN owner: TCPIP06 name: LAN1  
          cpu: 0 forwarding: ENABLED  
          RX bytes: 752 TX bytes: 1170  
          IPv6 Addresses:  
            10:0:0:0:0:0:0:1  
            FE80:0:0:0:209:5700:100:3D  
Ready; T=0.05/0.06 10:46:15
```

```
ifconfig eth0 add 50c0:c2c1:1010::1/64
```

```
Ready; T=0.10/0.12 10:46:57
```

```
ifconfig eth0
```

```
ETH0      inet addr: <NONE> mask: ?  
          UP BROADCAST MULTICAST MTU: 1500  
          vdev: 3300 type: QDIO ETHERNET portname: UNASSIGNED  
          ipv4 router type: NONROUTER  
          ipv6 router type: NONROUTER ipv6: ENABLED  
          LAN owner: TCPIP06 name: LAN1  
          cpu: 0 forwarding: ENABLED  
          RX bytes: 752 TX bytes: 1266  
          IPv6 Addresses:  
            10:0:0:0:0:0:0:1  
            50C0:C2C1:1010:0:0:0:0:1  
            FE80:0:0:0:209:5700:100:3D  
Ready; T=0.05/0.06 10:47:01
```

Putting It All Together

```
ifconfig eth4
DTCIFC2612E Unknown interface: eth4
Ready(00012); T=0.04/0.05 10:47:58
ifconfig eth4 10.0.0.1/27 add 50c0:c2c1:1010::7/64 veth fc00 tcpip06 lan3
Ready; T=0.14/0.17 10:48:57
ifconfig eth4
ETH4      inet addr: 10.0.0.1 mask: 255.255.255.224
          UP BROADCAST MULTICAST MTU: 1500
          vdev: FC00 type: QDIO ETHERNET portname: UNASSIGNED
          ipv4 router type: NONROUTER
          ipv6 router type: NONROUTER ipv6: ENABLED
          LAN owner: TCPIP06 name: LAN3
          cpu: 0 forwarding: ENABLED
          RX bytes: 0 TX bytes: 2376
          IPv6 Addresses:
              50C0:C2C1:1010:0:0:0:0:7
              FE80:0:0:0:209:5700:100:40
Ready; T=0.06/0.07 10:49:04
```

Removing an Interface – z/VM 5.3.0

```
ifconfig eth1
```

```
ETH1      inet addr: 10.10.0.1 mask: 255.255.255.128
          UP BROADCAST MULTICAST MTU: 1500
          vdev: FF00 type: HIPERS
          ipv6: DISABLED
          LAN owner: TCPIP06 name: LAN2
          cpu: 0 forwarding: ENABLED
          RX bytes: 0 TX bytes: 0
```

```
Ready; T=0.02/0.02 12:56:56
```

```
netstat devlinks
```

```
VM TCP/IP Netstat Level 530
```

```
Device DEV@FF00          Type: HIPERS          Status: Ready
  Queue size: 0          CPU: 0          Address: FF00          Port name: UNASSIGNED
  IPv4 Router Type: NonRouter  Arp Query Support: Yes
  Link ETH1              Type: QDIOIP          Net number: 0
  BytesIn: 0              BytesOut: 0
  Forwarding: Enabled      MTU: 1500             IPv6: Disabled
  Maximum Frame Size   : 16384
  Broadcast Capability: Yes
  Multicast Capability: Yes
```

```
Group                      Members
-----                      -
```

Group	Members
224.0.0.1	1

```
Ready; T=0.01/0.01 12:57:01
```

```
ifconfig eth1 -remove
```

```
DTCIFC2668E -REMOVE cannot be specified for an active interface
```

```
Ready(00008); T=0.01/0.01 12:57:07
```

Removing an Interface – z/VM 5.3.0 (Continued)

```
ifconfig eth1 down
Ready; T=0.03/0.04 12:59:40
ifconfig eth1 -remove
Ready; T=0.02/0.02 12:59:48
ifconfig eth1
DTCIFC2612E Unknown interface: eth1
Ready(00012); T=0.01/0.01 12:59:54
netstat devlinks
VM TCP/IP Netstat Level 530

Ready; T=0.01/0.01 13:00:00
```

■ Details

- Uses the new SIOCDINTERFACE ioctl() subcommand
 - Available to both REXX & C programs
 - Removes control block definitions and releases associated memory in the stack

Read More About It

- **IPWIZARD**

- **z/VM Guide for Automated Installation and Service, GC24-6099**

- **IFCONFIG**

- **z/VM TCP/IP Planning and Customization, SC24-6125**

Contact Information

■ By e-mail: **migueld@us.ibm.com**

■ In person: **USA 503.578.3561**

■ Mailing lists:

A number of listservs relevant to z/VM are available. Information on how to subscribe and view/search archives can be found at the following website:

<http://www.vm.ibm.com/techinfo/listserv.html>

Of particular interest:

IBMVM@listserv.uark.edu

IBMTCP-L@vm.marist.edu

LINUX-390@vm.marist.edu