2012
IBM System z Technical University
Enabling the infrastructure for smarter computing

IPv6 in z/VSE – Basics and Overview

zDG12

Ingo Franzki – ifranzki@de.ibm.com

Trademarks

The following are trademarks of the International Business Machines Corporation in the United States, other countries, or both.

Not all common law marks used by IBM are listed on this page. Failure of a mark to appear does not mean that IBM does not use the mark nor does it mean that the product is not actively marketed or is not significant within its relevant market.

Those trademarks followed by ® are registered trademarks of IBM in the United States; all others are trademarks or common law marks of IBM in the United States.

For a complete list of IBM Trademarks, see www.ibm.com/legal/copytrade.shtml:

*, AS/400®, e business(logo)®, DBE, ESCO, eServer, FICON, IBM®, IBM (logo)®, iSeries®, MVS, OS/390®, pSeries®, RS/6000®, S/30, VM/ESA®, VSE/ESA, WebSphere®, xSeries®, z/OS®, zSeries®, z/VM®, System i, System i5, System p, System p5, System x, System z, System z9®, BladeCenter®

The following are trademarks or registered trademarks of other companies.

Adobe, the Adobe logo, PostScript, and the PostScript logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.
Cell Broadband Engine is a trademark of Sony Computer Entertainment, Inc. in the United States, other countries, or both and is used under license therefrom.
Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.
Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.
Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.
UNIX is a registered trademark of The Open Group in the United States and other countries.
Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.
ITIL is a registered trademark, and a registered community trademark of the Office of Government Commerce, and is registered in the U.S. Patent and Trademark Office.
IT Infrastructure Library is a registered trademark of the Central Computer and Telecommunications Agency, which is now part of the Office of Government Commerce.

* All other products may be trademarks or registered trademarks of their respective companies.

Notes:

Performance is in Internal Throughput Rate (ITR) ratio based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput 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 improvements equivalent to the performance ratios stated here.

IBM hardware products are manufactured from new parts, or new and serviceable used parts. Regardless, our warranty terms apply.

All customer examples cited or described in this presentation are presented as illustrations of the manner in which some customers have used IBM products and the results they may have achieved. Actual environmental costs and performance characteristics will vary depending on individual customer configurations and conditions.

This publication was produced in the United States. IBM may not offer the products, services or features discussed in this document in other countries, and the information may be subject to change without notice. Consult your local IBM business contact for information on the product or services available in your area.

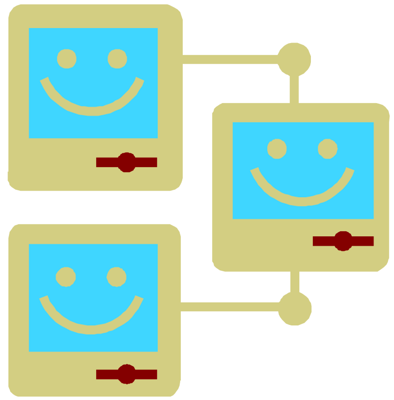
All statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only.

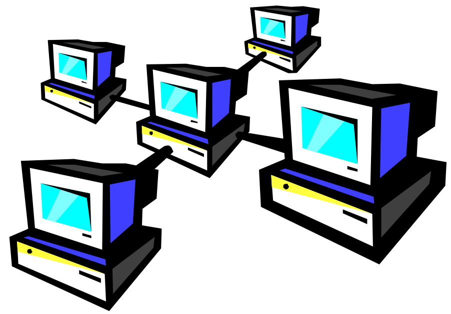
Information about non-IBM products is obtained from the manufacturers of those products or their published announcements. IBM has not tested those products and cannot confirm the performance, compatibility, or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

Prices subject to change without notice. Contact your IBM representative or Business Partner for the most current pricing in your geography.

Agenda

- What is IPv4?
- What is IPv6?
- Why migrate from IPv4 to IPv6?
- IPv6/VSE
- IPv6 enabled applications
- IPv6 enabled APIs





IPv4 Basics

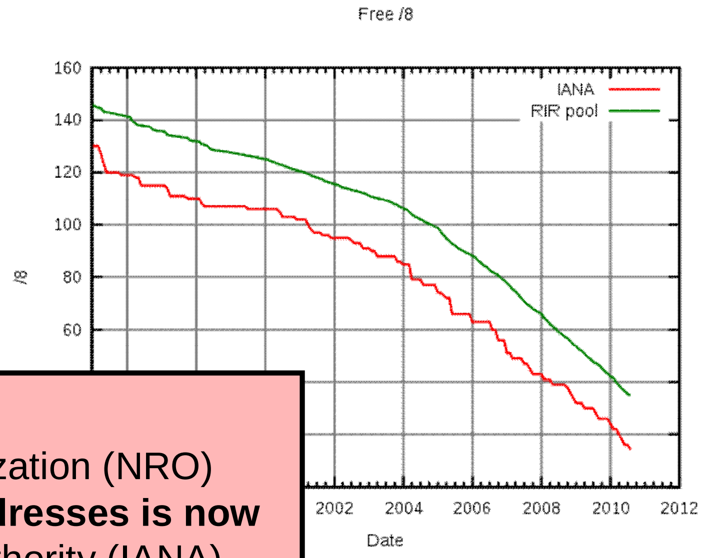
- IPv4 addresses are 32-Bit (4 Bytes) in length
  - Theoretically up to 4.294.967.296 unique addresses
  - IPv4 addresses are usually written in dot-decimal notation, which consists of the four octets of the address expressed in decimal and separated by periods:
    - Example: 207.142.131.235
    - Each block is 8 bits (1 byte). That is, the value range for each block is 0 to 255.

- In order to facilitate routing a data packet across multiple networks, the address is divided into two parts:
  - Network prefix: A contiguous group of high-order bits that are common among all hosts within a network.
  - Host identifier: The remaining low-order bits of the address that are not designated in the network prefix. This part specifies a particular device in the local network.
  - All endpoints which have the same network prefix are in the same IP network.
    - This implied that the endpoints can communicate directly with each others (e.g. through a switch, a hub or a crosslink cable), without the need to use a router.
  - To communicate between different networks, a router is required.

- In IPv4, subnet masks consist of 32 bits, usually a sequence of ones (1) followed by a block of 0s. The last block of zeros (0) designate that part as being the host identifier.
  - Example: 255.255.255.0

	Dot-decimal notation	Binary form
IP address	192.168.5.130	11000000.10101000.00000101.10000010
Subnet Mask	255.255.255.0	11111111.11111111.11111111.00000000
Network Portion	192.168.5.0	11000000.10101000.00000101.00000000
Host Portion	0.0.0.130	00000000.00000000.00000000.10000010

What’s the problem with IPv4?



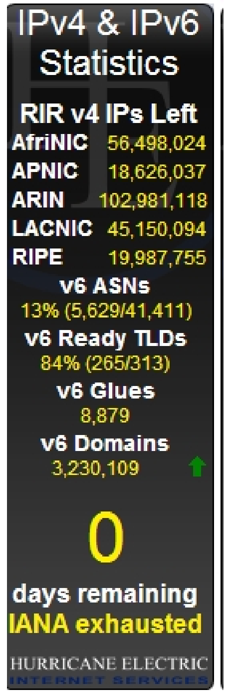
- The depletion of the IPv4 allocation pool has been a concern since the 1980s when the Internet started to experience dramatic growth
  - IPv4 only provided for approximately 4 billion addresses, a limit that is estimated to be reached before 2012
- At the mo
  addresse

On 3 February 2011, the Number Resource Organization (NRO) announced that the free **pool of available IPv4 addresses is now fully depleted.** The Internet Assigned Numbers Authority (IANA) allocated the last two blocks of IPv4 address space to a Regional Internet Registry.

This means that there are **no longer any IPv4 addresses available** for allocation from the IANA to the five Regional Internet Registries.

Reasons

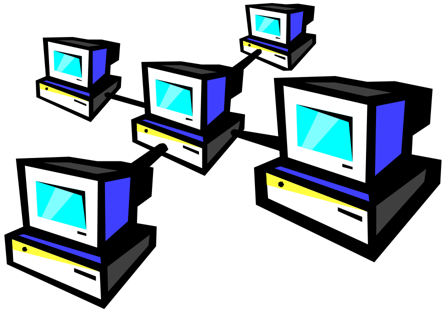
- Unforese
  - Exar
- Every mo
- Always-o IP
  addresse
- Almost every second house has at least one IP address (DSL, etc.)
- Inefficient address allocation: For example, large companies or universities were assigned class A address blocks with over 16 million IPv4 addresses each.
  - Universities and governmental organizations in the US hold about 74 % of the worldwide assigned IPv4 addresses.
  - Example: Genuity is a IP network provider in the US. They have reserved 3 class A networks. That is about 48 million addresses. However, China has only about 20 million addresses, which is not even half of what Genuity uses.



Other disadvantages of IPv4

- Network Address Translation (NAT)
  - Allows to remap multiple internal IP addresses to one external IP address
    - Every DSL router does that
  - But:
    - Quite complicated and processing time consuming
    - Only a limited number of connections possible at a time (limited due to the size of the NAT table)
    - Only connections from internal networks to external networks (outbound) are possible. For inbound connections a service like DynDNS is required.

- IPv4 contains many redundant and inefficient features:
  - Variable Length IP Header Options
    - Very processing time consuming for routers
  - IP Header Checksum
    - TCP Header also contains a checksum
  - Fragmentation of IP packets
    - Very processing time consuming and inefficient
  - Classification of IP packets
    - Still exists, but is not used anymore



The solution: IPv6

- **Wait a second, what about IPv5 ?**
  - Yes, it did exist as a test protocol only
  - Not used anymore

- **Design goals of IPv6**
  - Avoid errors made when designing IPv4
  - Much larger address range (16 bytes instead of only 4 bytes)
  - Increase scalability
    - 'Jumbograms' (up to 4 GB-1 per packet)
  - More efficient processing
    - Fixed Length IPv6 Header
    - No fragmentation
    - No Checksums
    - IPv6 header fields are aligned at 64 bit boundaries
  - Easily extendable
  - Simpler routing
  - 'True' multicasting
  - Auto configuration
    - Neighbor Discovery
    - Router solicitation
  - Support for mobile devices

IPv6 Basics

- IPv6 Addresses
  - 128 Bits in length (16 bytes)
    - 4 times larger than a IPv4 address
  - Up to 2^{128} (about 3.4×10^{38}) unique addresses
    - That's approximately 5×10^{28} (roughly 2^{95}) addresses for each of the roughly 6.8 billion (6.8×10^{9}) people alive in 2010.
    - In another perspective, this is the same number of IP addresses per person as the number of atoms in a metric ton of carbon!
  - IPv6 address are usually written as eight groups of four hexadecimal digits (each group representing 16 bits, or two bytes), where each group is separated by a colon (:).
    - Example: 2001:0db8:85a3:08d3:1319:8a2e:0370:7344
  - Leading zeroes in a group may be omitted (but at least one digit per group must be left):
    - 2001:0db8:0000:08d3:0000:8a2e:0070:7344 is the same as 2001:db8:0:8d3:0:8a2e:70:7344
  - A string of consecutive all-zero groups may be replaced by two colons. In order to avoid ambiguity, this simplification may only be applied once:
    - 2001:db8:0:0:0:0:1428:57ab is the same as 2001:db8::1428:57ab

IPv6 Basics - addressing

- IPv6 Addresses gets assigned to interfaces (network adapters)

- One interface (network adapter) can have multiple IPv6 addresses
  - Assigned address
  - Link local address

- Every IPv6 address has a "scope“:
  - Link local
  - Site local
  - Global

- IPv6 addresses are typically composed of two logical parts:
  - Routing prefix
    - The length of the prefix is specified with the address separated by a slash: /64
  - Interface identifier
    - Usually automatically determined from the MAC address of the interface
  - Internet service providers (ISPs) usually get assigned the first 32 bits (or less) as their network from a regional internet registry (RIR)

IPv6 Basics – address types

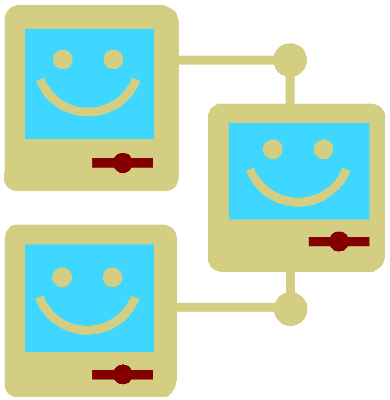
- IPv6 address types:
  - ::1/128 Is the loopback IPv6 address
  - ::/128 Is an unspecified IPv6 address
  - FF00::/8 Is a multicast IPv6 address
  - FE80::/10 Is a link local IPv6 address
  - FEC0::/10 Is a site local IPv6 address
  - FC00::/7 Is a unique local IPv6 address (private address)
  - All others are global unicast IPv6 addresses

- Interfaces (network adapters) have at least 2 IPv6 addresses:
  - Assigned (global) IPv6 address
    - 806::1:2
  - Link local IPv6 address
    - FE80 + Mac Address (020000000008)
    - FE80:0:0:0:0200:0000:0100:0008
    - FE80::200:0:100:8

- Further address types:
  - Site local IPv6 address (not used anymore)
  - Multicast IPv6 address

IPv6 Basics – auto configuration

Goal: Plug ‘n’ Play network

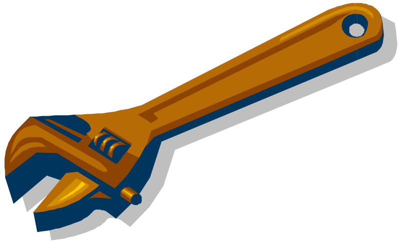


- An IPv6 endpoint needs at least 3 pieces of information to be able to communicate:
  - IPv6 address
  - IPv6 network
  - IPv6 gateway
- Right after the start, an endpoint only knows its link local address
  - E.g. determined from the MAC address of the interface
  - With that, it can only communicate within its local network segment
- The interface then uses Neighbor Discovery Protocols to search for routes in its local network segment
  - It sends requests to the multicast address FF02::2, which all routes are reachable at (Router Solicitation)
  - Available routes then reply with information about the network
- Router also send Router Advertisements in regular intervals to all hosts in the network(s) segment they are responsible for
- ICMPv6 provides essential functions in an IPv6 network
  - Address Resolution Protocol (ARP) is replaced by Neighbor Discovery Protocol (NDP)

Migration from IPv4 to IPv6

- Contrary to popular belief, IPv6 is not backward compatible !
- But: IPv4 and IPv6 networks can be used concurrently over the same cable and with the same endpoint

Transition methods:



- **Dual IP Stacks**
  - That's the easies possibility
  - The IP stack supports both protocols concurrently
    - Examples: Linux since Kernel 2.6, Windows since XP SP1
  - Existing IPv4 applications can continue to run unchanged
    - Applications can be IPv6-enabled over time, one after the other
- **Tunneling**
  - IPv6 packets are sent as payload of other protocols (usually IPv4) to a tunneling broker, which is located in an IPv6 network. The broker extracts the IPv6 packet from the payload and sends it as IPv6 packet through IPv6 routing to the final destination.
    - Example: 6in4 using Tunneling-Broker

Migration from IPv4 to IPv6

Which infrastructure parts needs to be migrated?

- **Layer 1 devices (e.g. hubs)**
  - Those are completely transparent for IPv6
- **Layer 2 devices (switches)**
  - Devices which have been purchased within the last 10 years most likely support IPv6 already
- **Layer 3 devices (routers)**
  - Usually not required for local LANs
  - Today most router manufacturer provide IPv6 capable routers
  - Routers that use Multiprotocol Label Switching (MPLS) are protocol independent
- **Endpoints (PCs, Server, etc.)**
  - Most modern operating systems support IPv6
- **Applications**
  - May have to be adapted (IPv6-enabled) to be able to work with IPv6 addresses

Why should a z/VSE customer care about IPv6?

Independent on your concrete benefits

à You will have to care about IPv6, sooner or later!

Why?

- Your internet service provider (ISP) migrates to IPv6
- On 3 February 2011, the Number Resource Organization (NRO) announced that the free pool of available IPv4 addresses is now fully depleted.
- Your customers or partners are only reachable via IPv6 (e.g. China)
- Governmental organizations may only allow manufacturers of IPv6 capable products and applications to participate in advertised biddings
  - Example: The US Department of Defense (DoD) only allows products that are on the “Unified Capabilities Approved Products List” (UC APL) for its advertised biddings.
    - “This list is used by procurement offices in the DoD and the U.S. Federal agencies for ongoing purchases and acquisitions of IT equipment”

IPv6 Products for z/VSE

IPv6/VSE Version 1 Release 1

IPv6/VSE is a registered trademark of Barnard Software, Inc.

Extract from
Announcement Letter 210-066

- The IPv6/VSE V1 product is designed to provide an IPv6 solution for z/VSE to:
  - Allow z/VSE users to participate in an IPv6 network
  - Bring the benefits of IPv6 functionality to z/VSE users
  - Help z/VSE users to meet the requirements of the commercial community and governmental agencies and thus fulfills the statement of direction in Software Announcement 209-319, dated October 20, 2009

- IPv6/VSE V1 is designed to provide an IPv6 TCP/IP stack, IPv6 application programming interfaces (APIs), and IPv6-enabled applications.
- The IPv6/VSE product also includes a **full-function IPv4 TCP/IP stack**, IPv4 application programming interfaces and IPv4 applications. The IPv4 TCP/IP stack does not require the IPv6 TCP/IP stack to be active.

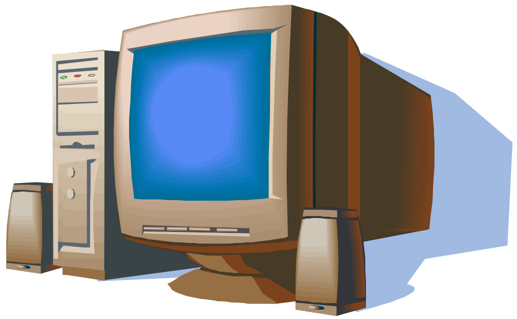
- IPv6/VSE V1 supports the IPv6 and IPv4 protocols, while TCP/IP for VSE/ESA V1.5 supports the IPv4 protocol only.

Available since: May 28, 2010

IPv6 enabled applications

The following applications and tools are part of the IPv6/VSE product:

- FTP Server (POWER queues, VSAM catalogs, SAM file, z/VSE libraries, …)
- Batch FTP Client
- TN3270E server (TN3270/TN3270E Terminal & TN3270E Printer Sessions)
- Network Time Protocol Server (NTP server)
- Network Time Protocol Client (NTP client)
- System Logger Client
- Batch Email Client
- Batch LPR
- Batch Remote Execution client (REXEC)
- Batch PING
- GZIP data compression
- REXX automation



Home grown applications may need to get adapted (IPv6 enabled)

Dual Stack Support

The IPv6/VSE product contains 2 TCP/IP stacks:

- **IPv6 Stack**
  - Provides support for the IPv6 protocol
  - IPv6 application programming interfaces (APIs)
  - IPv6-enabled applications.
  - Supports IPv6 only, no IPv4
- **IPv4 Stack**
  - Provides support for the IPv4 protocol
  - IPv4 application programming interfaces (APIs)
  - IPv4-enabled applications.
  - Supports IPv4 only, no IPv6

To allow applications to use IPv4 and IPv6 at the same time

- Run both stacks (in separate partitions)
- COUPLE the 2 stacks together

→ The 2 coupled stacks act as one dual stack, supporting IPv6 and IPv4

IPv6 Products for z/VSE

CSI International

TCP/IP for VSE (CSI)

Statement of direction from September 13, 2011:

http://www.csi-international.com/csi-products/TCPIP/Statement_of_Direction_for_IPv6_in_TCP-IP_for_VSE_rev%2002_20110913.pdf

Capabilities	Introduced in this release of TCP/IP FOR VSE
IP address parsing for both IPv4 and IPv6	1.5G
IP address de-parsing into the shortest valid form	
Improved debugging and tracing information	
More control of stack processes by applications	
Access to flow-control information	
IPv6-enabled CSI applications: telnet, FTP, etc.	1.5H
Full IPv6 support	2.0

z/VSE V5.1 plus PTFs – OpenSSL Support

- **What is OpenSSL?**
  - OpenSSL is an Open Source project providing an SSL implementation and key management utilities
  - Available for most Unix-style operating systems, MAC, Windows, and IBM System i (OS/400)
  - For details on OpenSSL refer to http://www.openssl.org/

- **Why OpenSSL on z/VSE?**
  - The TCP/IP stack from Connectivity Systems, Inc. has an own SSL implementation
  - What about the other two stacks:
    - IPv6/VSE from Barnard Systems, Inc.
    - Linux Fast Path (LFP) provided by IBM
  - All stacks could use one single SSL implementation: **OpenSSL**
  - OpenSSL is widely used in the industry
  - Latest RFC's implemented
  - One central place for access to crypto hardware software updates, migration to higher versions

z/VSE V5.1 plus PTFs – OpenSSL Support

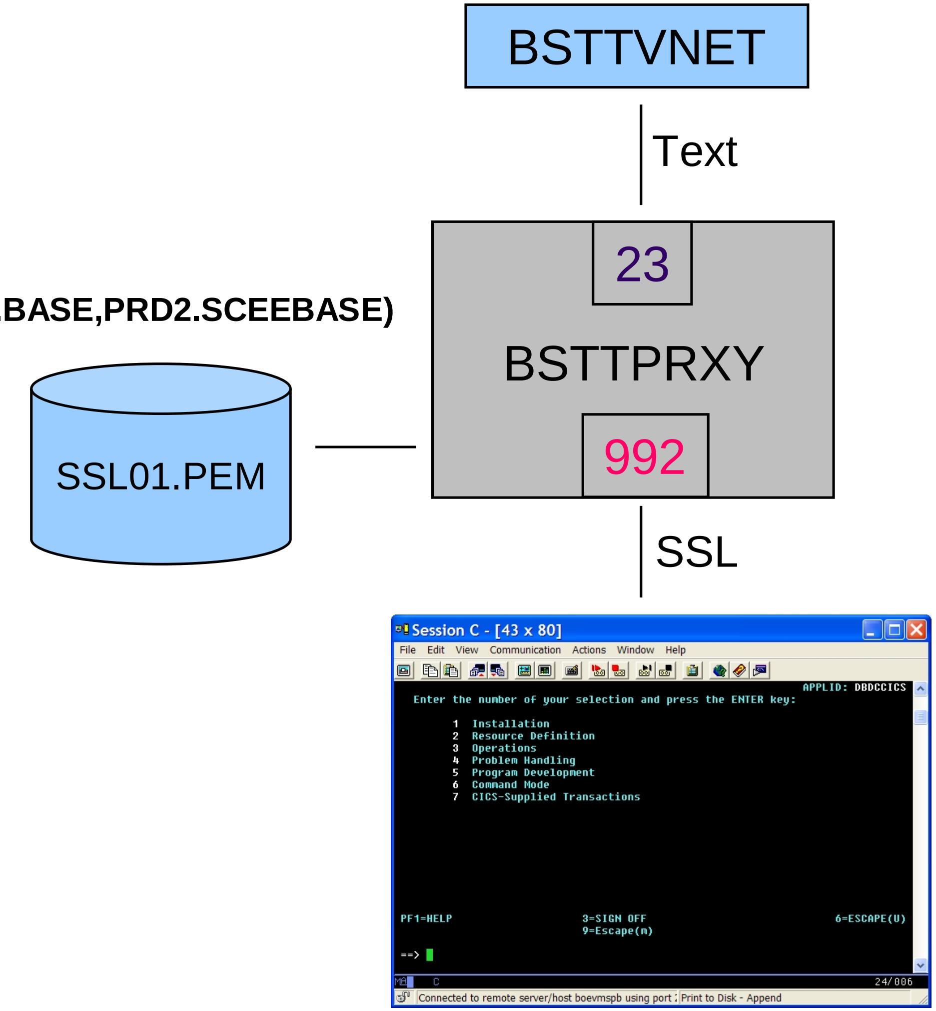
- **What is available on z/VSE?**
  - OpenSSL 1.0.0d runtime library
  - New component: z/VSE cryptographic services, 5686-CF9-17-51S
  - Available on z/VSE 5.1 plus PTFs (DY47397/UD53864 and DY47414/UD53863)
  - Software implementations for <u>all</u> algorithms with <u>all</u> key lengths
  - Hardware Crypto Support (Crypto Express cards and CPACF)
  - Programming APIs:
    - OS390 / z/OS compatible SSL API (gsk_initialize(), gsk_secure_soc_init(), etc.)
    - Subset of the OpenSSL API (LE/C)

- **OpenSSL Exploitation**
  - IPv6/VSE product exploits OpenSSL
    - **SSL Proxy Server** (BSTTPRXY)
      - Proxies a clear text connection (IPv4 or IPv6) into an SSL/TLS connection (IPv4 or IPv6) and vice versa
    - **Automatic TLS Facility** (BSTTATLS)
      - Automatically converts any application into SSL/TLS application

z/VSE V5.1 plus PTFs – OpenSSL Support

BSTTPRXY example:

```
// JOB BSTTPRXY - BSI SSL PROXY SERVER
// OPTION SYSPARM='02'
// SETPARM SSL$DBG='NO'
// SETPARM SSL$ICA='YES'
// LIBDEF *,SEARCH=(PRD2.CONFIG,PRD2.TCPIPB,PRD1.BASE,PRD2.SCEEBASE)
// EXEC BSTTPRXY,SIZE=BSTTPRXY
ID 02
*
KEYRING PRD2.CONFIG
KEYFILE SSL01
SECTYPE TLSV1
OPTION SERVER
HANDSHAKE_SERVER 1
*
PROXY TCP V4   992 SSL * TO V4   23 TXT * 127.0.0.1
/*
/&
```



z/VSE V5.1 plus PTFs – OpenSSL Support

BSTTATLS example:

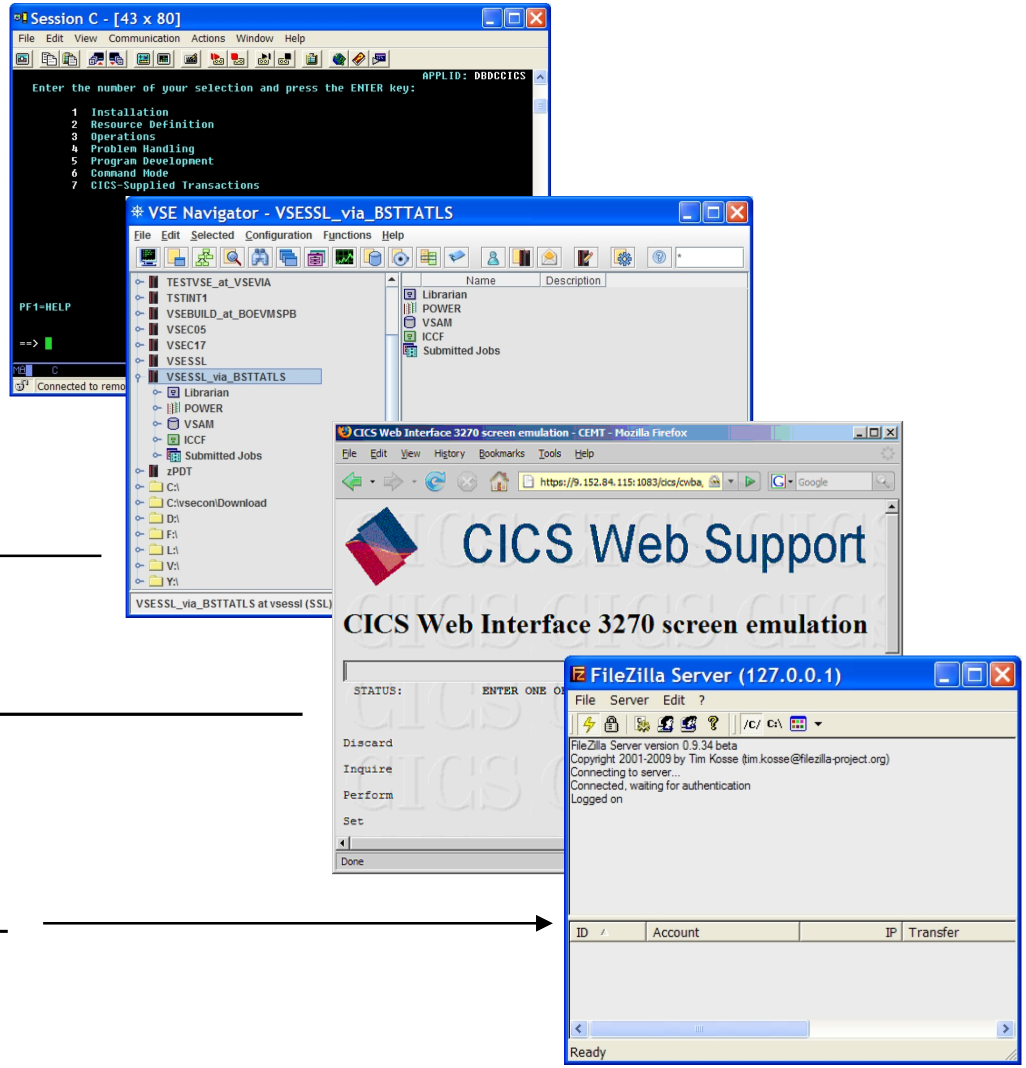
BSTTVNET

Conn. Server

CICS

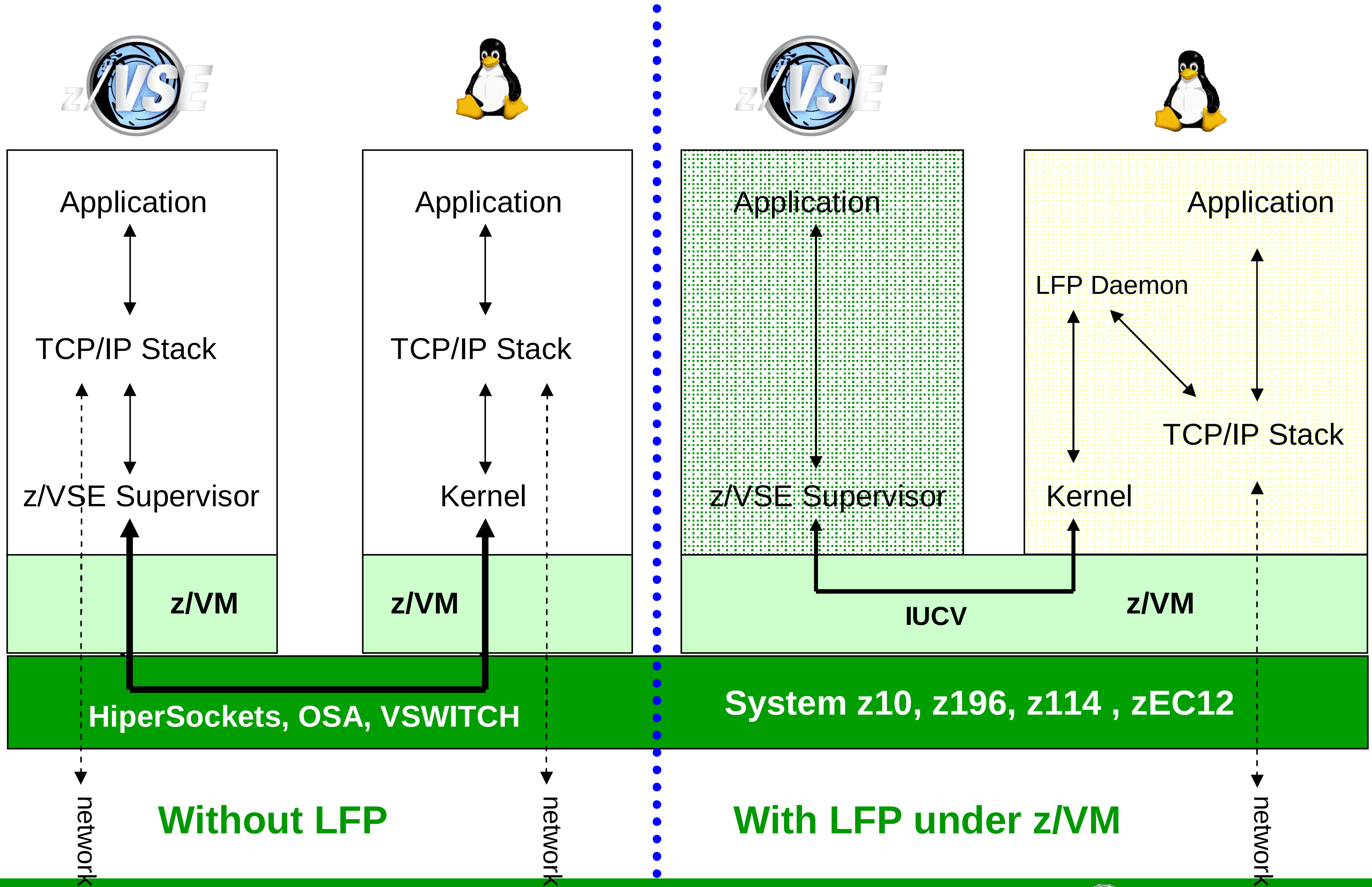
BSTTFTPC

```
// EXEC BSTTATLS,SIZE=BSTTATLS
ID 02
*
KEYRING PRD2.CONFIG
*
KEYFILE SSL01
SECTYPE SSL30
OPTION SERVER
ATTLS 23 AS 992 SSL
*
KEYFILE SSL02
OPTION SERVER
SECTYPE TLSV1
HANDSHAKE_SERVER 2
ATTLS 2893 AS 2894 SSL
*
KEYFILE SSL03
SECTYPE TLSV1
OPTION SERVER
ATTLS 80 AS 443 SSL
*
KEYFILE SSL01
SECTYPE TLSV1
OPTION CLIENT
OPTION FTP
ATTLS 21 TO 9.152.222.71 AS 990 SSL
*
/*
```



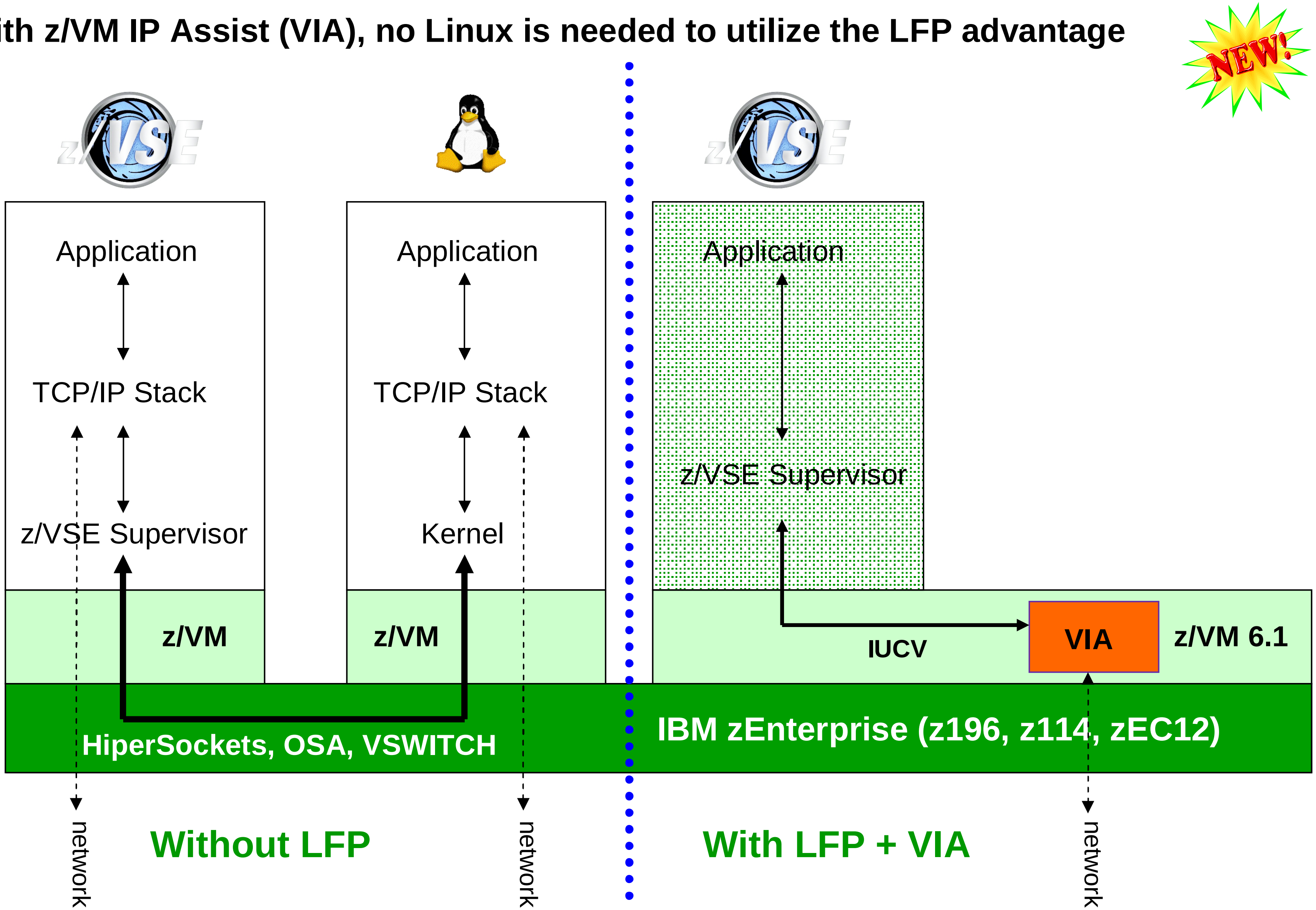
Linux Fast Path in a z/VM environment (z/VSE 4.3 or later)

Faster communication between z/VSE and Linux applications



New: z/VSE z/VM IP Assist (VIA) (z/VSE 5.1 + z/VM 6.1)

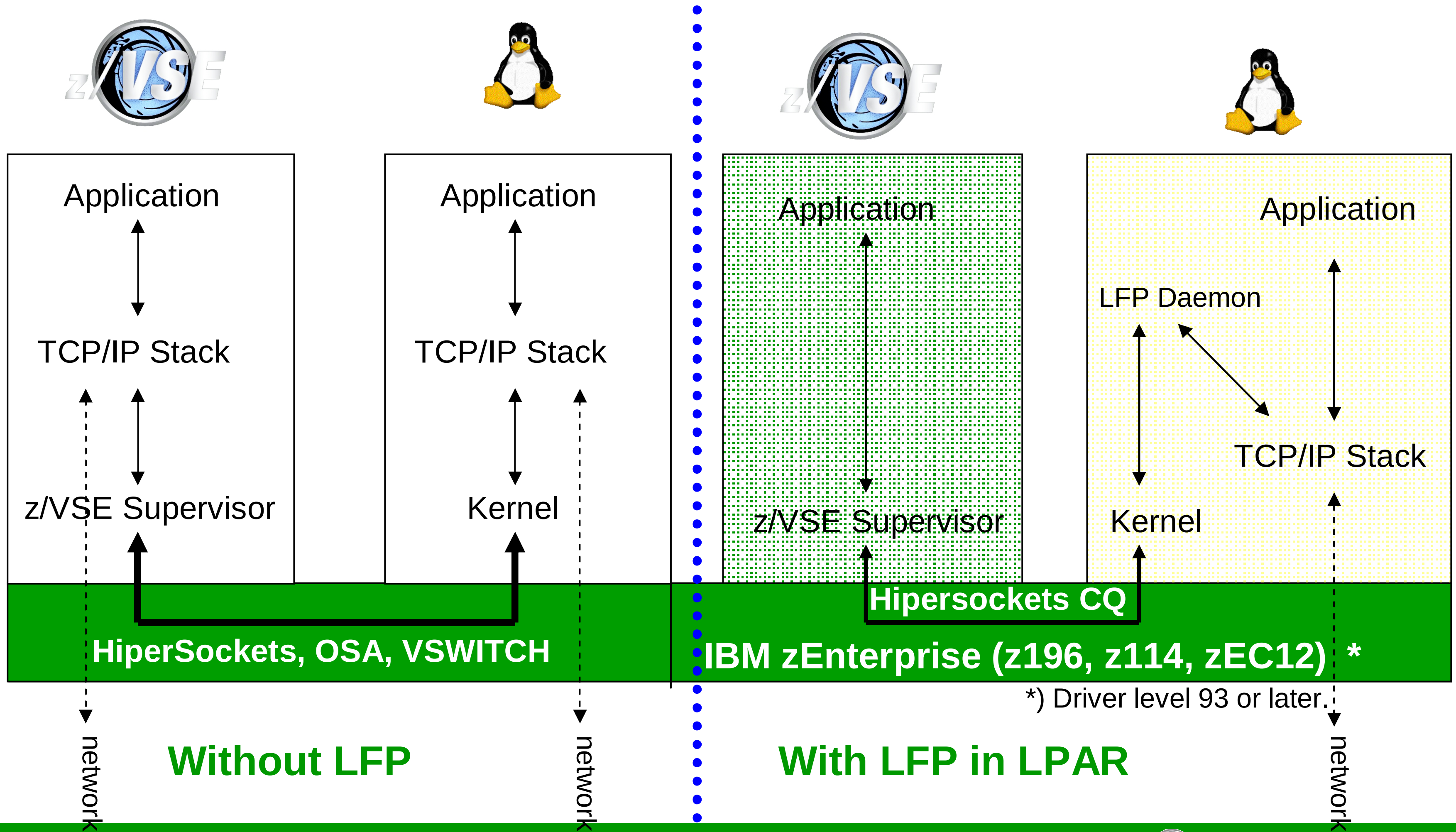
With z/VM IP Assist (VIA), no Linux is needed to utilize the LFP advantage



New: Linux Fast Path in an LPAR environment (z/VSE 5.1 + PTFs)

Faster communication between z/VSE and Linux applications

à Exploits the HiperSockets Completion-Queue support of IBM zEnterprise (z196, z114, zEC12)



z/VSE Fast Path to Linux on System z (LFP)

- **Most existing applications run unchanged with Linux Fast Path**
  - Provided they use one of the supported Socket API (LE/C, EZA or ASM SOCKET)
    - And they do not use any CSI or BSI specific interface, features or functions
    - Since z/VSE V5.1: LFP supports IPv6 NEW!

- **IBM Applications supporting Linux Fast Path**
  - VSE Connector Server
  - CICS Web Support
  - VSE Web Services (SOAP) support (client and server)
  - CICS Listener
  - DB2/VSE Server and Client
  - WebSphere MQ Server and Client
  - VSAM Redirector
  - VSE VTAPE
  - VSE LDAP Support
  - VSE Script Client
  - POWER PNET
  - TCP/IP-TOOLS included in IPv6/VSE product (e.g. FTP Server/Client)

- **Customer applications should run unchanged:**
  - Provided they use one of the supported Socket API (LE/C, EZA or ASM SOCKET)

IPv6 programming interfaces (APIs)

Existing APIs have been extended:

- **EZASOCKET and EZASMI API (APAR DY47077 for z/VSE 4.2)**
  - New functions:
    - GETADDRINFO
    - FREEADDRINFO
    - GETNAMEINFO
    - NTOP
    - PTON
  - New Address-Family: AF_INET6

- **LE/C Socket API**
  - Has been made IPv6 capable with z/VSE 5.1

- **CSI's Assembler SOCKET Macro**
  - BSI: transparent IPv6 extension using the existing SOCKET Macros

Summary

z/VSE is IPv6 ready

- **The TCP/IP stack is IPv6 ready**
  - IPv6/VSE Product
  - Fast Path to Linux on System z
    - IPv6 enabled since z/VSE 5.1
- **Many standard applications are IPv6 ready**
  - FTP, Telnet, e-Mail, ...
    - Part of the IPv6/VSE Products
- **Programming interfaces (APIs) are IPv6 ready**
  - EZASMI and EZASOCKET
  - LE/C

But: Is your network environment already IPv6 ready ?

Questions ?

?

Please also see the IPv6 Flyer:
http://public.dhe.ibm.com/common/ssi/ecm/en/zss03058usen/ZSS03058USEN.PDF

IBM Systems Lab Services and Training

Helping you gain the IBM Systems skills needed for smarter computing

- Comprehensive education, training and service offerings
- Expert instructors and consultants, world-class content and skills
- Multiple delivery options for training and services
- Conferences explore emerging trends and product strategies

www.ibm.com/training

Special Programs:

- IBM Systems ‘Guaranteed to Run’ Classes -- ***Make your education plans for classes with confidence!***
- Instructor-led online (ILO) training ***The classroom comes to you.***
- Customized, private training
- Lab-based services assisting in high tech solutions