



IBM Software Group – Enterprise Networking Solutions  
z/OS® V1R12 Communications Server

## *z/OS Communications Server – Application Overview*



© Copyright International Business Machines Corporation 2010. All rights reserved.

This presentation provides an overview of the new functions in z/OS V1R12 Communications Server for applications.

## ***Application integration / data consolidation and standards***

- Configurable default address selection policy table
  - RFC 3484 allows configuration of the IPv6 default address selection algorithm
- Socket API support for source address selection
  - RFC 5014 allows applications to prefer a temporary or public IPv6 source address
- Enhancements to IPv6 router advertisement
  - RFC 4191 Default Router Preferences and More Specific Router
  - RFC 5175 Expanded Flags option
- Resolver support for IPv6 connections to DNS name servers
  - Configure list of name servers in TCPIP.DATA using both IPv4 and IPv6 addresses
- Improved resolver reaction to unresponsive name server

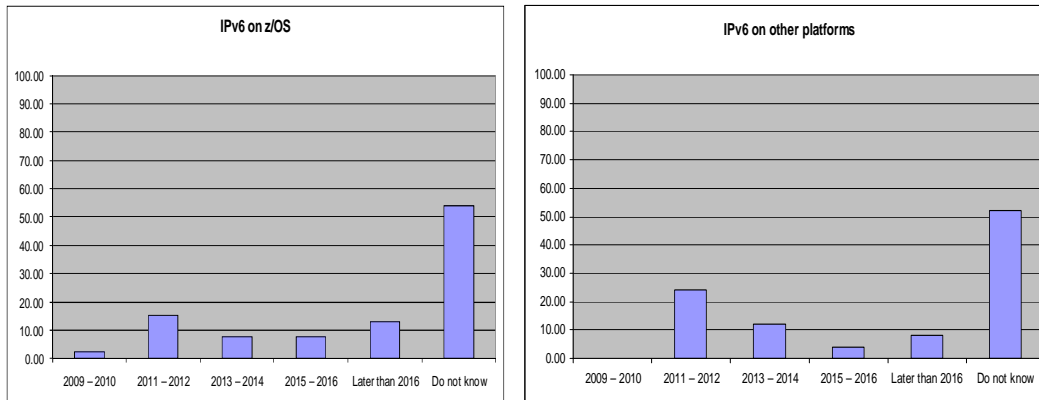
**IPv6 standards compliance –  
meeting the mandates of the  
US government and others**



Several line items fall within the application integration, data consolidation and standards theme. You now have the ability to configure the default address selection policy table that is used for both source address selection and destination address selection. New socket APIs are provided that allow applications to specify whether applications prefer a temporary or public IPv6 source address be assigned. Default routers can now specify a preference field to identify which one or ones should be preferred for default routes. Routers can also specify off-link prefixes which can be reached through the router and, like default routes, can include a preference weight. The z/OS resolver can now send queries to DNS name servers over IPv6 connections. The z/OS resolver is also enhanced to notify the operator when a DNS name server is unresponsive to DNS queries sent to that name server.

## When will z/OS customers need IPv6?

- The majority of z/OS customers do not know
  - Expectations are that it is needed slightly earlier on other platforms than z/OS
- It is time to start thinking, learning, and preparing *now*!



Source: Survey conducted by ENS early 2009 among a selected set of customers (39 responses to this question)

The Enterprise Networking Solutions group conducted a small survey early 2009 among a selected set of customers, primarily z/OS Communications Server Customer Advocate Program customers. The 70 questions included when those customers expected to need IPv6 on z/OS and a question about when they expected to need IPv6 on platforms other than z/OS.

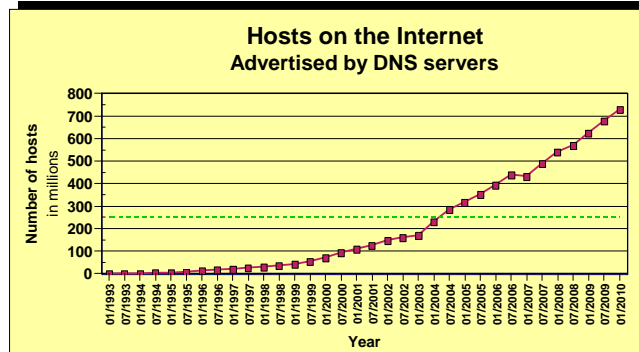
The majority had not thought much about IPv6 at all, which is somewhat alarming.

As you will see from some of the statistics shown on these charts, new growth in the Internet community has to be based on IPv6 within the next two years. IPv6 is an increasing reality for everyone. That does not mean you all have to start migrating your intranets to IPv6. It does, however, mean you will all have to start thinking about how to exist in an Internet environment that consists of IPv4-only nodes, IPv4 and IPv6 capable nodes, and IPv6-only nodes.

A not very well known fact is that an application running on an IPv6-only node cannot communicate directly with an application that runs on an IPv4-only node. So for many years you will all have to start dealing with new types of application-level gateways between IPv6 traffic and IPv4 traffic.

## IPv4 address usage since early 1993

- Projected Internet Assigned Numbers Authority (IANA) Unallocated Address Pool Exhaustion
  - September 2011
- Projected Regional Internet Registries (RIR) Unallocated Address Pool Exhaustion
  - September 2012
- z/OS Communications Server continues to focus on IPv6 standards currency
  - US DoD/NIST
  - IPv6 Forum



- What is the upper practical limit (the ultimate pain threshold) for number of assigned IPv4 addresses? Some predictions said 250,000,000 (250 million), others go up to 1,000,000,000 (one billion or one milliard).
- Source: <https://www.isc.org/solutions/survey>
- Source: <http://www.potaroo.net/tools/ipv4/index.html>
- Source: <http://penrose.uk6x.com/>

**If you want to stay in business after 2011/2012, you'd better start paying attention!**  
 Do not worry too much; the sky isn't falling – IPv4 and IPv6 will coexist for many years to come.  
 Your applications need to be able to use both. If you write directly to the TCP/IP sockets layer, you need to start changing those applications

There are many unknowns around IPv6, but one thing is well known and that is IPv4 addresses are running out. Various Internet-related organizations keep track of the number of assigned IPv4 addresses. They all indicate that within the next two to three years the Internet will run out of IPv4 addresses. First, the Assigned Internet Numbers Authority (IANA) will run out of IPv4 address blocks that can be assigned to the Regional Internet Registries, or RIRs. Shortly after, within one to two years, those regional registries will run out of IPv4 addresses. At that point in time no more IPv4 addresses can be assigned to ISPs or private companies that need Internet accessible IPv4 addresses. Remember that even NAT-based Internet connectivity technologies require a certain amount of public IPv4 addresses.

Various government organizations have begun to plan for a deployment of IPv6 connectivity. Best known are the US Department of Defense, or DoD, and the National Institute of Standards and Technology, or NIST. Both organizations have defined IPv6 compliance requirements that IT solutions must meet. z/OS is tracking these compliance requirements carefully. z/OS V1R10 has achieved the “Special Interoperability Test Certification of the IBM z/OS Version 1.10 Operating System for IBM Mainframe Computer Systems for Internet Protocol Version 6 Capability” of the US Department of Defense. For more information, see [http://jitc.fhu.disa.mil/adv\\_ip/register/certs/ibmzosv110\\_dec08.pdf](http://jitc.fhu.disa.mil/adv_ip/register/certs/ibmzosv110_dec08.pdf)

## Is doomsday approaching?

<http://www.potaroo.net/tools/ipv4/index.html>



### IPv4 Address Report



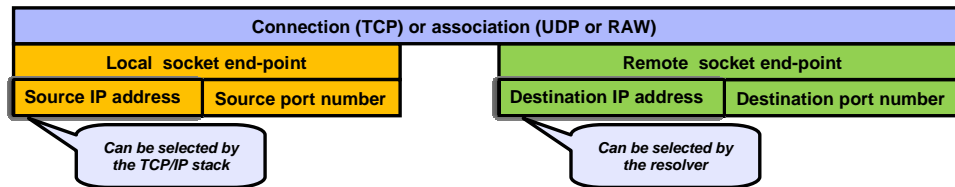
This report is auto-generated by a daily script. The report you are seeing here was generated at 11-Mar-2010 07:58 UTC.

Projected IANA Unallocated Address Pool Exhaustion: 30-Sep-2011

Projected RIR Unallocated Address Pool Exhaustion: 17-Sep-2012

Go to <http://www.potaroo.net/tools/ipv4/index.html> for a daily status of when IPv4 addresses run out. The site is updated daily based on the latest statistics and as you approach the predicted date, the fluctuations are getting smaller. The site currently shows that over 92% of the IPv4 addresses are already assigned, and the IPv4 address pool can be exhausted in a little over two years.

## Source and destination IP address selection overview



- Sockets programs can specify all four elements entirely, but do not have to
- The TCP/IP stack can choose
  - Source IP address
  - Source port number
- The resolver can choose
  - Destination IP address
  - Destination port number
- RFC 3484 “Default Address Selection for Internet Protocol version 6 (IPv6)”
  - Defines configurable rules for how parts of the source and destination IP address selection logic works – the default source and destination IP address selection
  - This rule-based logic kicks in after all the existing z/OS TCP/IP logic for selection of source and destination IP addresses has been exhausted

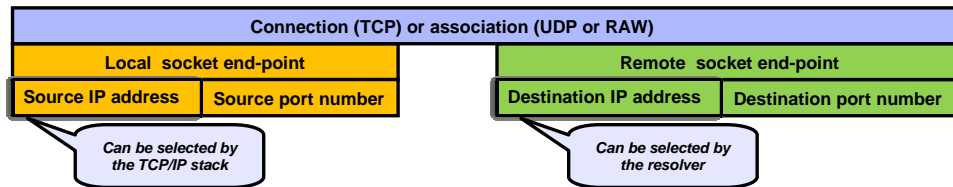
For sockets programs, five elements are used to identify a given socket connection. The five elements are: the protocol, such as TCP; the source IP address; the source port number; the destination IP address; and the destination port number. A sockets program can specify some or all of these elements, or might let the TCP/IP stack and resolver do so for the application.

The TCP/IP stack can select the source IP address for an application that does not explicitly bind to a source IP address. z/OS TCP/IP provides many ways to influence this selection through the source VIPA functions. The TCP/IP stack can also select the source port number from the available pool of ephemeral port numbers.

The resolver can select the destination IP address, using the default address selection algorithm defined in RFC 3484. The resolver can also select the destination port number if the application uses the gerservbyname function.

RFC 3484 defines a set of rules that are used in performing both source IP address selection and destination IP address selection. While the RFC provides a default set of rules to be used, it also suggests that products allow the rules to be customized by the user. This rule-based logic kicks in after all the existing z/OS TCP/IP logic for selection of source and destination IP addresses has been exhausted.

## RFC 3484 implications – primarily implications for IPv6



### ▪ Source address selection

- No impact when destination is an IPv4 address
- For IPv6 destinations, the new configurable rules kick in if neither SOURCEVIPA nor SRCIP selects a source IP address
- New rules configurable by way of a new TCP/IP profile statements

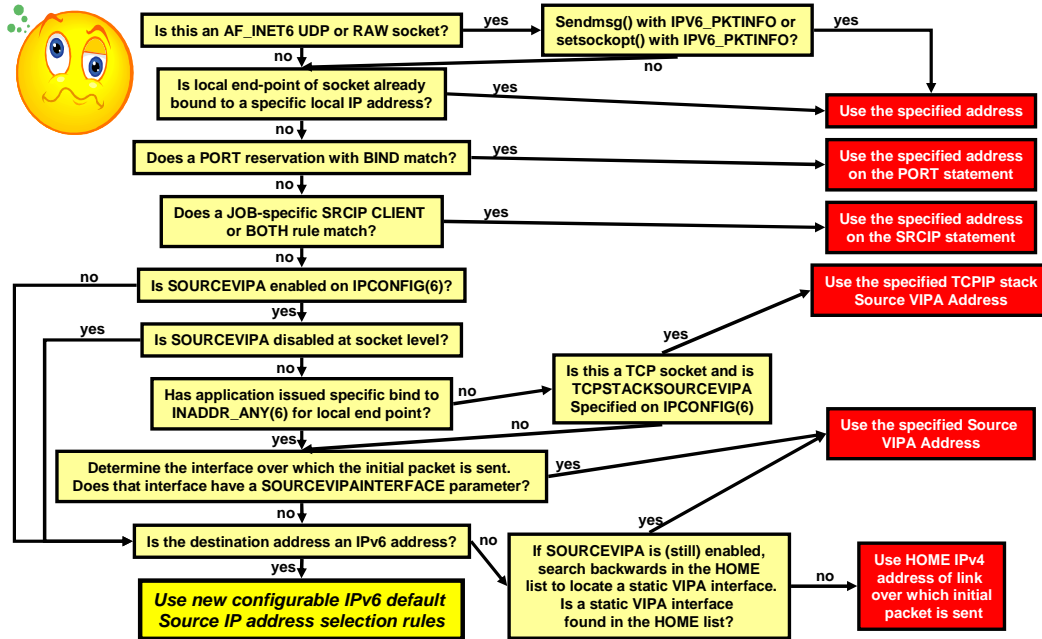
### ▪ Destination address selection

- Governs the order in which IP addresses are returned by the getaddrinfo() resolver call
- No changes for gethostbyname()
- No changes if IPv6 is not enabled
- SORTLIST continues to govern order of IPv4 addresses
- New configurable rules can be used to alter preference for IPv6 over IPv4 addresses to the opposite, but otherwise no impact to IPv4 destinations

The default address selection algorithms defined in RFC 3484 are not always applied. For the source IP address selection, the algorithms only apply to IPv6 destinations. Even then, the algorithms are only used if the application does not bind to an IP address, and neither SOURCEVIPA, SRCIP nor the BIND parameter on the PORT statement selects a source IP address.

For destination IP address selection, the algorithms are only applied if IPv6 is enabled for the TCP/IP stack. It also requires the application to be enabled for IPv6 and use the getaddrinfo function. No changes are made if the application uses the older gethostbyname function. Fortunately, the vast majority of applications are already using getaddrinfo.

### Y'all remember this: z/OS TCP/IP source IP address selection logic (simplified!)



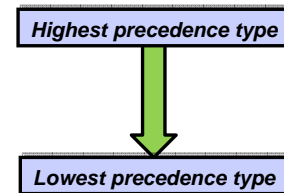
z/OS Communications Server provides many ways to influence the source IP address that is used for a new connection. The application itself can select the source IP address by binding to a specific local IP address. For UDP and RAW sockets, the application can select the source IP address by using the IPV6\_PKTINFO option on either the sendmsg or the setsockopt functions. You can use the BIND parameter on the PORT reservation statement, or use the SRCIP statement, to select the source IP address. SOURCEVIPA, TCPSTACKSRCVIPA, or SOURCEVIPAINTERFACE can be used to select a source VIPA address.

If none of these selects a source IP address, and the destination is an IPv6 address, then the TCP/IP stack applies the default source address selection algorithms as defined in RFC 3484.



### **Route precedence before z/OS V1R12**

- Which route is installed in the routing table when routes to the same destination are received from multiple sources?
  1. Non-replaceable static routes
  2. OSPF routes
  3. RIP routes
  4. Router advertisement routes (IPv6)
  5. Replaceable static routes
- Managed by the TCP/IP stack and OMPROUTE in combination



The TCP/IP stack can learn routes from a variety of sources. The routes can be static routes, either replaceable or non-replaceable. They can also be router advertisement routes (for IPv6). Finally, they can be routes from OMPROUTE, either OSPF or RIP. When routes for the same destination are available from multiple sources, the stack uses precedence rules to determine the routes to install in the route table.

The routes to be installed are decided based on route type. First, any non-replaceable static routes are installed. If there are no non-replaceable static routes, any OSPF routes are installed. If there are no OSPF routes, any RIP routes are installed. If there are no RIP routes, any router advertisement routes are installed. Finally, if there are no routes of any other type, any replaceable static routes are installed.

Route precedence is managed by TCP/IP and OMPROUTE working together.

## ***Route precedence improvements***

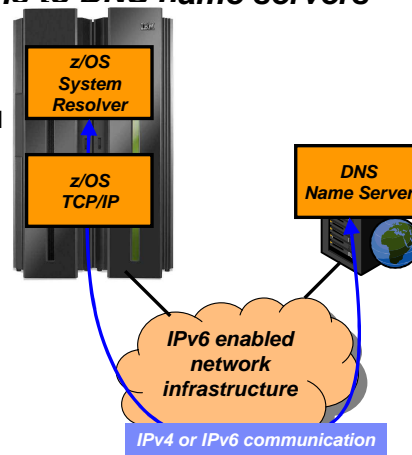
- IPv6 default router advertisements have been expanded with metrics
  - Router advertisement routes can now have a precedence associated
  - Allows for differentiation among multiple routers that all provide a default route
  - All router advertisements are kept by TCP/IP in case a higher precedence routes goes away
    - These currently unused router advertisements can now be displayed by netstat
- IPv6 router advertisement allows a router to inform about off-link destinations (network prefixes) that can be reached through the router
  - These are also associated with precedence information

IPv6 default router advertisements can include a precedence associated with default routes. This allows a network administrator to indicate which routers attached to a LAN are to be used as the preferred default router, while providing backup default routers for redundancy. The TCP/IP stack will install a default route with the highest precedence available.

IPv6 router advertisements have also been expanded to allow a router to specify which off-link destinations can be reached through the router. As with default routes, these routes also include a precedence.

## Resolver support for IPv6 connections to DNS name servers

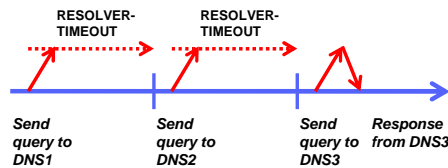
- Allows the system resolver to send requests to DNS name servers using IPv6 communication
  - Specify IPv6 addresses on the NSINTERADDR and NAMESERVER configuration statements
  - Resolver sends queries using IPv4, IPv6 or both based on the configuration
- Applications cannot manipulate IPv6 addresses using low-level resolver API calls, such as `res_query` and `res_search`
  - Only IPv4 addresses are supported on these APIs
  - The entire list, containing IPv4 and IPv6 addresses, is used for searching
    - Unless the application modifies the list, in which case only the returned IPv4 addresses are used
- The type of address returned (IPv4/IPv6) is not tied to the transport between the resolver and the name server. IPv6 addresses can be returned before z/OS V1R12



IPv6 support was first introduced in z/OS V1R4. Since that release, the resolver has provided the ability to return both IPv4 and IPv6 addresses to an application. However, the requests to perform the mapping between a host name and IP address were required to use an IPv4 connection between the resolver and the DNS name server. In V1R12, these requests can now flow over an IPv6 connection in addition to an IPv4 connection by specifying IPv6 addresses on the NSINTERADDR and NAMESERVER configuration statements.

Applications that use low-level resolver API calls, such as `res_query` and `res_search`, cannot manipulate IPv6 addresses. These APIs were defined a long time before IPv6, and use hard-coded structures that only allow 32-bit IPv4 addresses to be passed. As long as the application does not modify the list of addresses passed in these APIs, then resolver will use both IPv4 and IPv6 connections when communicating with a DNS name server. However, if an application does modify the list of addresses, then the resolver will only be able to search the IPv4 addresses that are returned. Fortunately, very few applications use these low-level APIs.

## Improved resolver reaction to unresponsive name servers



### Assume:

- 3 name servers in TCPIP.DATA
- 2 first are un-responsive
- RESOLVERTIMEOUT 30 seconds

*It takes 60+ seconds to get a response, and it will do so for every query made to the resolver*

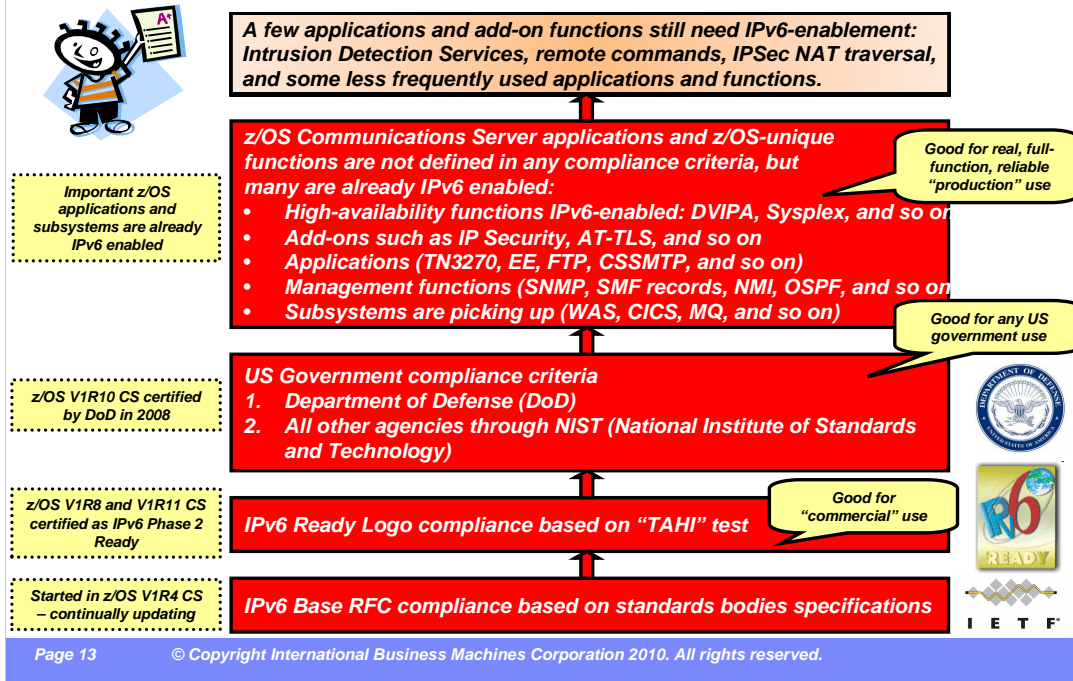
- Un-responsive name servers can impact performance significantly
  - Based on the setting of number of name servers, timeout, and retry limit in TCPIP.DATA
    - Beware that default RESOLVERTIMEOUT is five seconds – should be lowered to seconds or sub-seconds!
    - Before z/OS V1R12, the default RESOLVERTIMEOUT was 30 seconds
- So far, no warning messages have been issued when name servers repetitively time out
- z/OS V1R12 adds messages to the console when name servers are un-responsive
- Configurable un-responsiveness threshold: percentage of failed queries over a 5-minute period
  - Default 25%
- A message will also be issued when a name server is deemed to have become responsive again

```
EZZ9308E UNRESPONSIVE NAME SERVER DETECTED AT IP ADDRESS 9.43.25.200
EZZ9310I NAME SERVER 9.43.25.200
          TOTAL NUMBER OF QUERIES SENT           6000
          TOTAL NUMBER OF FAILURES                2100
          PERCENTAGE                               35%
```

You can specify the UNRESPONSIVETHRESHOLD resolver setup statement to customize the threshold value when the resolver declares a name server to be unresponsive. By default, the resolver starts with a threshold setting of 25%. This means that if a name server fails to respond to more than 25% of the queries sent to it over a five-minute period, the resolver will declare the name server to be unresponsive. The threshold applies to all name servers in the network, and either all name servers are monitored, or no name servers are monitored.

Because the value represents a percentage, any integer between 0 and 100 can be specified for UNRESPONSIVETHRESHOLD. If you code UNRESPONSIVETHRESHOLD(0), that disables the monitoring function, and the resolver maintains no statistics for any name servers in this situation. You can dynamically modify the setting for UNRESPONSIVETHRESHOLD using the MODIFY RESOLVER command. If you specify UNRESPONSIVETHRESHOLD(0), the function is disabled immediately by the MODIFY command. If you specify a non-zero value, the percentage is used by the resolver for comparison purposes at the next one-minute checkpoint.

## IPv6 – state of z/OS and z/OS Communications Server



z/OS has included IPv6 support since z/OS V1R4. Since then, z/OS Communications Server has been continually updating support to address the evolving IPv6 standards. z/OS also works to ensure the implementation is both standards-compliant and interoperates with a wide range of IPv6 products. In addition to extensive internal testing, the IPv6 Ready Logo compliance test includes checks for RFC compliance in addition to interoperability tests. z/OS V1R11 was certified as IPv6 Phase 2 Ready in 2009. The US Department of Defense and the US National Institute of Standards and Technology (NIST) produce IPv6 requirement profiles. These profiles specify a set of requirements that must be met by any product that is purchased by the US Government. z/OS V1R10 was certified as meeting the Department of Defense requirements in 2008. The Department of Defense has since stopped performing IPv6 certification, so subsequent releases of z/OS no longer need to be certified. However, NIST has instituted its own certification program to be used for US government purchases. z/OS plans to take z/OS V1R12 through the certification process once z/OS V1R12 becomes generally available.

While most applications and z/OS-unique functions are already IPv6 enabled, there are a few which are not, such as Intrusion Detection Services and IPSec NAT traversal. When you are considering deploying IPv6 on z/OS, you should look at the list of functions and applications that are not enabled for IPv6 support to determine whether you use any. If so, verify that it is acceptable for you to use them only in an IPv4 environment.

## **Feedback**

Your feedback is valuable

You can help improve the quality of IBM Education Assistant content to better meet your needs by providing feedback.

- Did you find this module useful?
- Did it help you solve a problem or answer a question?
- Do you have suggestions for improvements?

Click to send e-mail feedback:

[mailto:iea@us.ibm.com?subject=Feedback\\_about\\_wnapps.ppt](mailto:iea@us.ibm.com?subject=Feedback_about_wnapps.ppt)

This module is also available in PDF format at: [../wnapps.pdf](#)

You can help improve the quality of IBM Education Assistant content by providing feedback.

## Trademarks, copyrights, and disclaimers

IBM, the IBM logo, ibm.com, CICS, IBM, and z/OS are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of other IBM trademarks is available on the Web at "[Copyright and trademark information](http://www.ibm.com/legal/copytrade.shtml)" at <http://www.ibm.com/legal/copytrade.shtml>

THE INFORMATION CONTAINED IN THIS PRESENTATION IS PROVIDED FOR INFORMATIONAL PURPOSES ONLY. WHILE EFFORTS WERE MADE TO VERIFY THE COMPLETENESS AND ACCURACY OF THE INFORMATION CONTAINED IN THIS PRESENTATION, IT IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. IN ADDITION, THIS INFORMATION IS BASED ON IBM'S CURRENT PRODUCT PLANS AND STRATEGY, WHICH ARE SUBJECT TO CHANGE BY IBM WITHOUT NOTICE. IBM SHALL NOT BE RESPONSIBLE FOR ANY DAMAGES ARISING OUT OF THE USE OF, OR OTHERWISE RELATED TO, THIS PRESENTATION OR ANY OTHER DOCUMENTATION. NOTHING CONTAINED IN THIS PRESENTATION IS INTENDED TO, NOR SHALL HAVE THE EFFECT OF, CREATING ANY WARRANTIES OR REPRESENTATIONS FROM IBM (OR ITS SUPPLIERS OR LICENSORS), OR ALTERING THE TERMS AND CONDITIONS OF ANY AGREEMENT OR LICENSE GOVERNING THE USE OF IBM PRODUCTS OR SOFTWARE.

© Copyright International Business Machines Corporation 2010. All rights reserved.