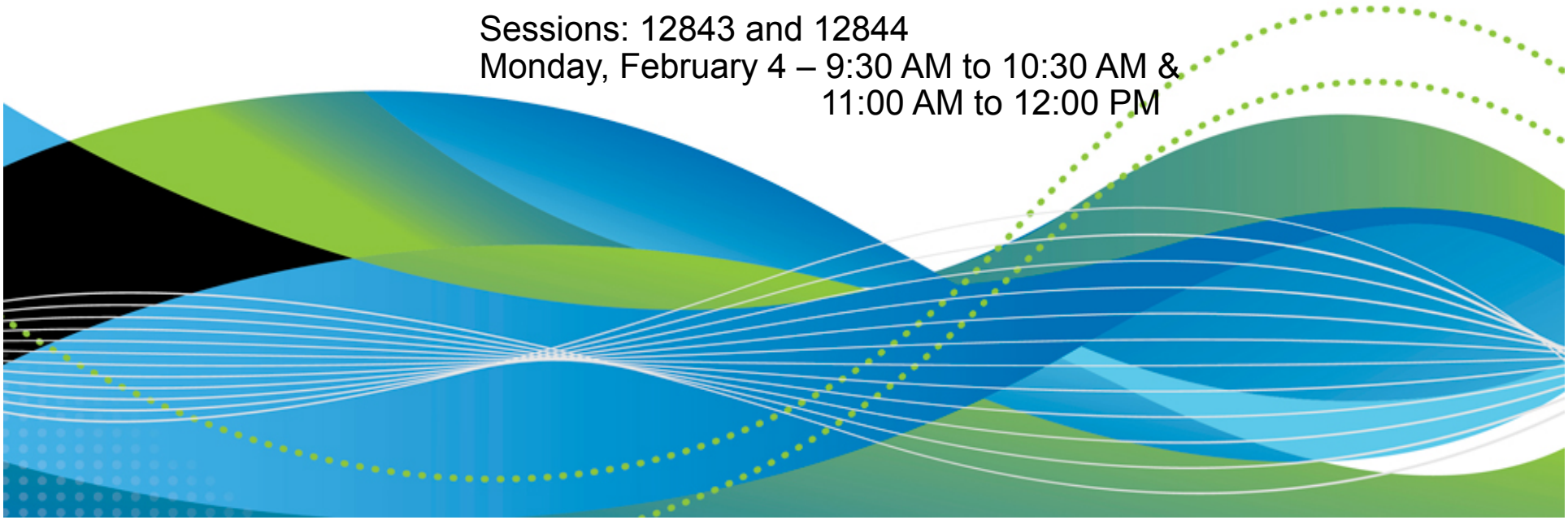
#SHAREorg

SHARE
Technology · Connections · Results

z/OS Communications Server Technical Update

Gus Kassimis - kassimis@us.ibm.com
Sam Reynolds - samr@us.ibm.com
IBM Enterprise Networking Solutions
Raleigh, NC, USA

Sessions: 12843 and 12844
Monday, February 4 – 9:30 AM to 10:30 AM &
11:00 AM to 12:00 PM



z/OS Communications Server Technical Update

Session number:	12843 and 12844
Date and time:	Monday, February 4, 2013 - 9:30 AM – 10:30 and 11:00 AM – 12:00 PM
Location:	Golden Gate 3
Program:	Communications Infrastructure
Project:	Communications Server
Track:	Network Support and Management
Classification:	Technical
Speaker:	Gus Kassimis, IBM Sam Reynolds, IBM
Abstract:	z/OS Communications Server combines TCP/IP and VTAM support to better address the needs of today's complex networks. This two-part session will preview selected content from z/OS V2R1 Communications Server.

Trademarks, notices, and disclaimers

The following terms are trademarks or registered trademarks of International Business Machines Corporation in the United States or other countries or both:

- Advanced Peer-to-Peer Networking®
- AIX®
- alphaWorks®
- AnyNet®
- AS/400®
- BladeCenter®
- Candle®
- CICS®
- DataPower®
- DB2 Connect
- DB2®
- DRDA®
- e-business on demand®
- e-business (logo)
- e business(logo)®
- ESCON®
- FICON®
- GDDM®
- GDPS®
- Geographically Dispersed Parallel Sysplex
- HiperSockets
- HPR Channel Connectivity
- HyperSwap
- i5/OS (logo)
- i5/OS®
- IBM eServer
- IBM (logo)®
- IBM®
- IBM zEnterprise™ System
- IMS
- InfiniBand ®
- IP PrintWay
- IPDS
- iSeries
- LANDP®
- Language Environment®
- MQSeries®
- MVS
- NetView®
- OMEGAMON®
- Open Power
- OpenPower
- Operating System/2®
- Operating System/400®
- OS/2®
- OS/390®
- OS/400®
- Parallel Sysplex®
- POWER®
- POWER7®
- PowerVM
- PR/SM
- pSeries®
- RACF®
- Rational Suite®
- Rational®
- Redbooks
- Redbooks (logo)
- Sysplex Timer®
- System i5
- System p5
- System x®
- System z®
- System z9®
- System z10
- Tivoli (logo)®
- Tivoli®
- VTAM®
- WebSphere®
- xSeries®
- z9®
- z10 BC
- z10 EC
- zEnterprise
- zSeries®
- z/Architecture
- z/OS®
- z/VM®
- z/VSE

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

The following terms are trademarks or registered trademarks of International Business Machines Corporation in the United States or other countries or both:

- 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 there from.
- 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.
- InfiniBand is a trademark and service mark of the InfiniBand Trade Association.
- 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.

Notes:

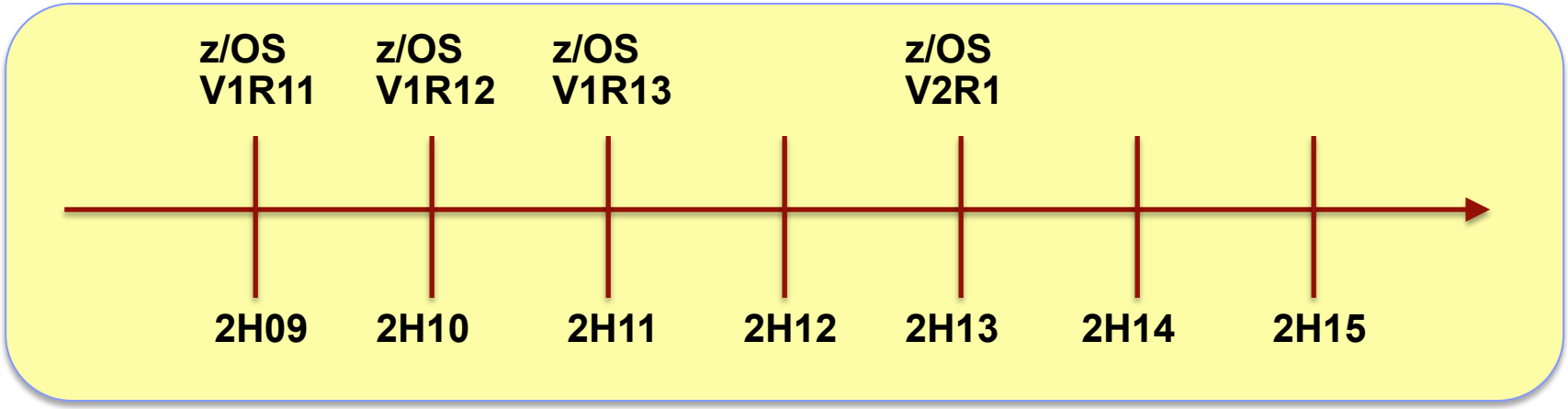
- Performance is in Internal Throughput Rate (ITR) ratio based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput that any user will experience will vary depending upon considerations such as the amount of multiprogramming in the user's job 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.

Refer to www.ibm.com/legal/us for further legal information.

z/OS V2R1 Communications Server disclaimer

- Plans for the z/OS Communications Server are subject to change prior to general availability
- Information provided in this presentation may not reflect what is actually shipped by z/OS Communications Server
- This presentation includes an early overview of selected future z/OS Communications Server enhancements
- The focus of this presentation is the Communications Server in z/OS V2R1

Plans may change before GA of z/OS V2R1 CS



Statements regarding IBM future direction and intent are subject to change or withdrawal, and represent goals and objectives only.

Agenda

- z/OS CS Requirements
- Economics and Platform Efficiency
- Application / Middleware / Workload Enablement
- Availability
- Simplification
- Security
- EE / SNA / Miscellaneous

Disclaimer: All statements regarding IBM future direction or intent, including current product plans, are subject to change or withdrawal without notice and represent goals and objectives only. All information is provided for informational purposes only, on an "as is" basis, without warranty of any kind.

z/OS Communications Server and Social Media

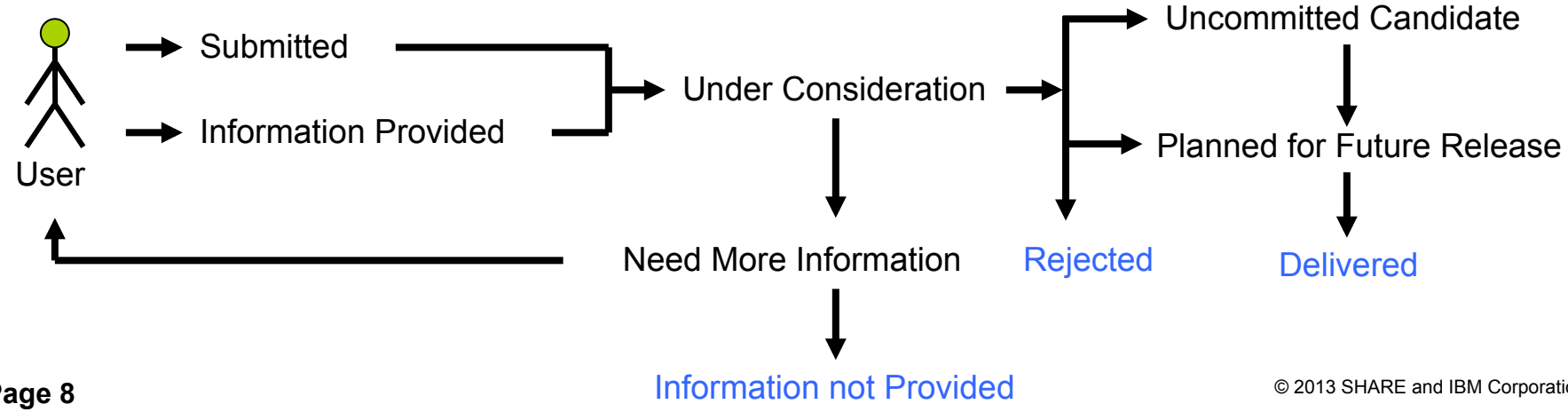
- z/OS Communications Server has a blog, Facebook page, Twitter feed, and YouTube channel.
- We expect to be much more active via these channels in 2013 and beyond.
- Follow us for announcements, hints and tips, screencasts on topics of interest, etc.

What will the z/OS community need from z/OS networking in 2013-2015?

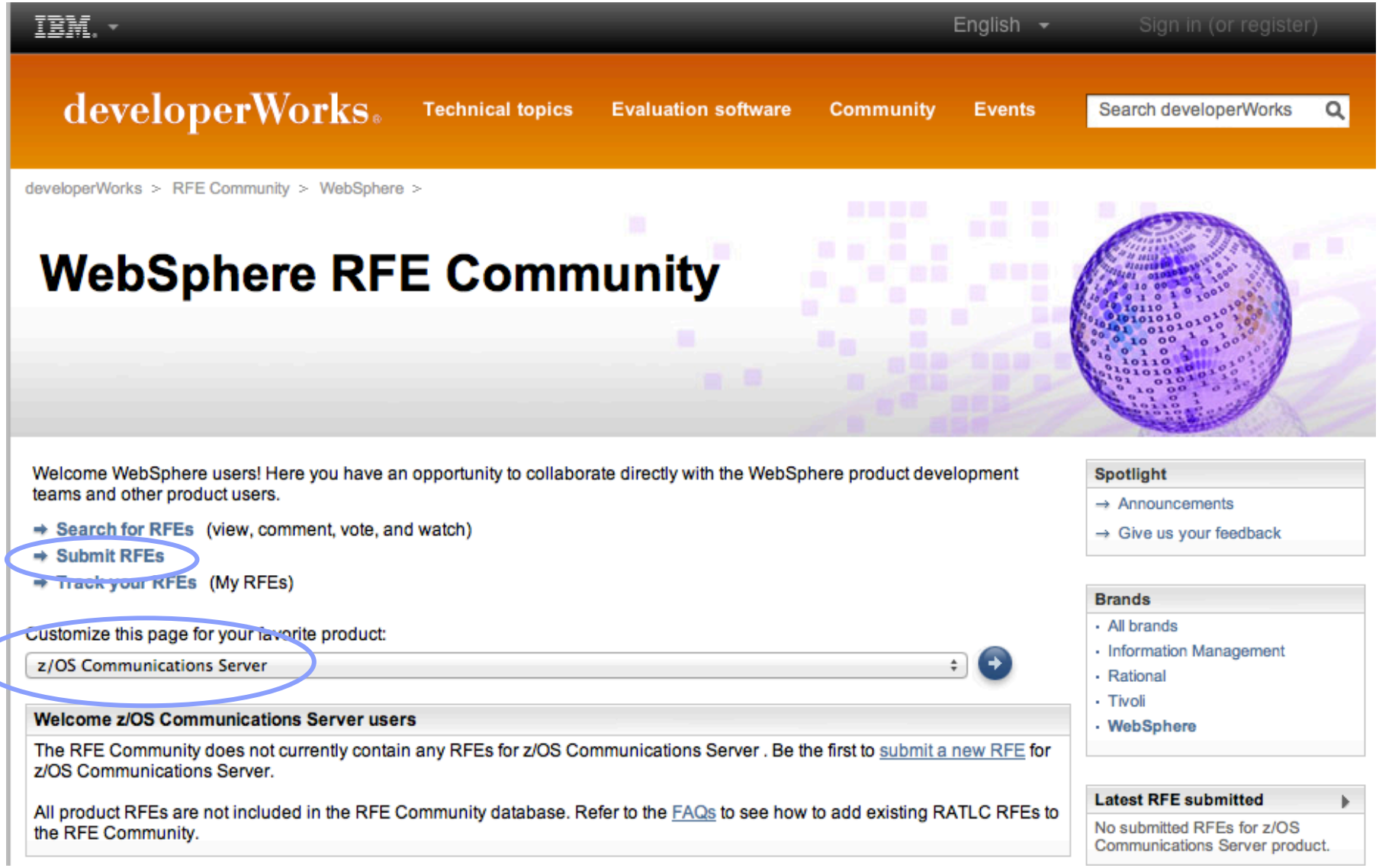
- **System z technology is expected to continue to evolve**
  - Networking software need to support new technologies such as zEnterprise
- **Access to System z system-level skills will continue to be an issue**
  - Retiring existing people, who grew up with system z
  - New people becoming responsible for the overall system z environment – including z/OS networking
  - Note: follow the IBM Academic Initiative
    - https://www.ibm.com/developerworks/university/academicinitiative/
- **Security will continue to be a hot topic**
  - Per customer survey, over 50% of network traffic will need encryption within the next few years
  - Trade organizations and governments continue to establish security and privacy compliance requirements that must be met
- **Price/performance requirements are high priority**
  - Continued demand for reduced cost in combination with increased performance and scalability on system z
- **Demand for increased "autonomic" system integration capabilities**
  - Continued demand for improved integration with other hardware and software platforms for more complex heterogeneous solutions
- **IANA has already run out of IPV4 addresses. Regional registries are also running out (APNIC has already run out, RIPE is expected to be next)**
  - IPv6 compliance (USGv6, IPv6-Ready, TAHI test suite, etc.)

RFE: New Requirements Process for z/OS Communications Server

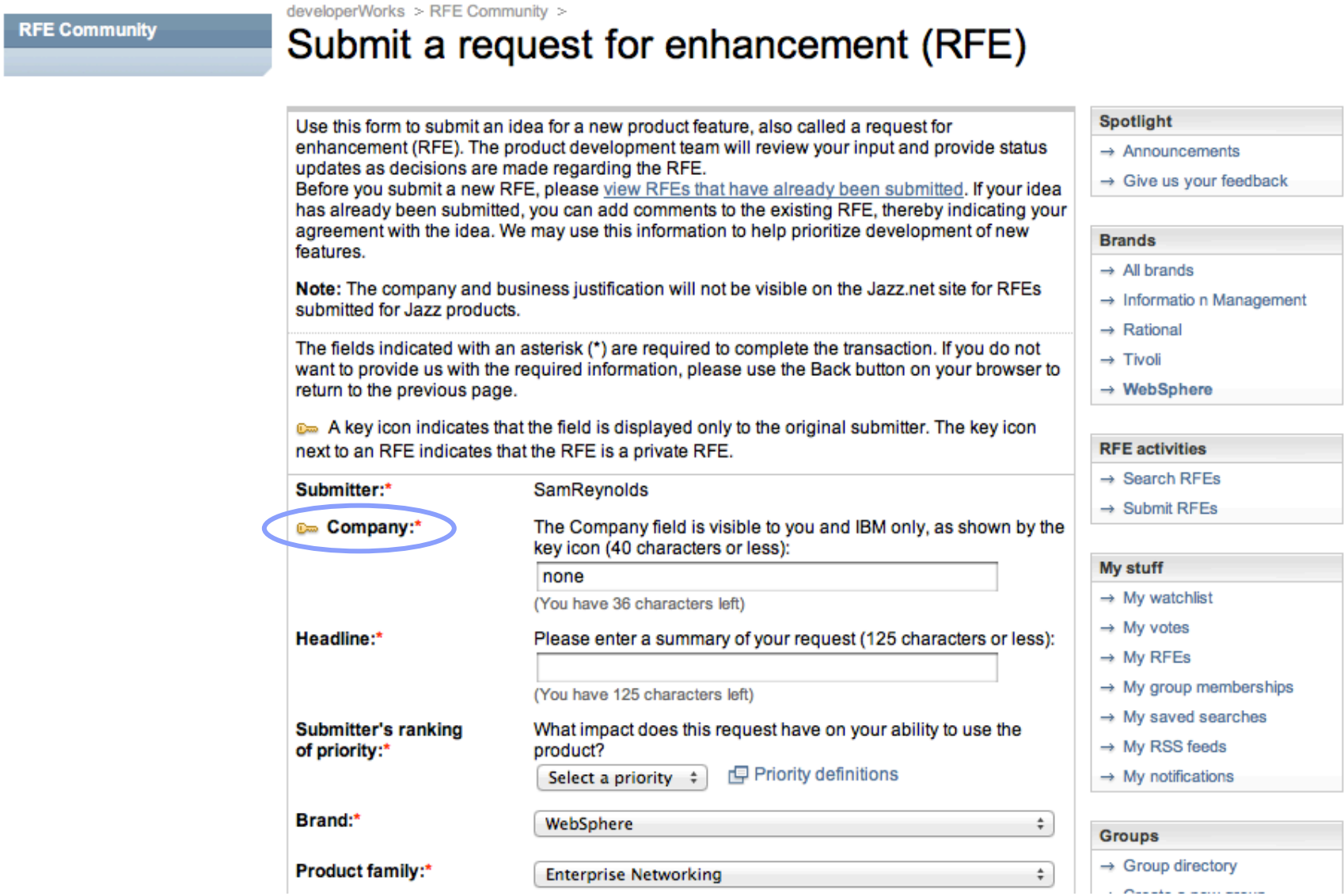
- **RFE (Request for Enhancement)**
  - New web-based interface for submitting requirements to some IBM products
  - Unlike FITS, RFE is available to anyone that signs up for an IBM Universal ID.
  - Can submit requirements, vote on requirements, watch and comment on requirements, and interact with IBM if more information is needed
  - RFEs submitted against z/OS CS will initially be private, but most will be converted to public status after an initial screening.
  - Even public RFEs do have certain fields that will always be private (only viewable by the submitter and IBM):
    - Customer name
    - Business justification
    - Attachments can be private (your choice at submission)
  - Can keep up with requirements you are interested in via watchlists, bookmarks, and/or RSS feeds
  - http://www.ibm.com/developerworks/rfe/?PROD_ID=498 (QR code above)



RFE: New Requirements Process for z/OS Communications Server …



RFE: New Requirements Process for z/OS Communications Server …



RFE Community

developerWorks > RFE Community >

Submit a request for enhancement (RFE)

Use this form to submit an idea for a new product feature, also called a request for enhancement (RFE). The product development team will review your input and provide status updates as decisions are made regarding the RFE.
Before you submit a new RFE, please view RFEs that have already been submitted. If your idea has already been submitted, you can add comments to the existing RFE, thereby indicating your agreement with the idea. We may use this information to help prioritize development of new features.

Note: The company and business justification will not be visible on the Jazz.net site for RFEs submitted for Jazz products.

The fields indicated with an asterisk (*) are required to complete the transaction. If you do not want to provide us with the required information, please use the Back button on your browser to return to the previous page.

A key icon indicates that the field is displayed only to the original submitter. The key icon next to an RFE indicates that the RFE is a private RFE.

Submitter:* SamReynolds

Company:* The Company field is visible to you and IBM only, as shown by the key icon (40 characters or less):
none
(You have 36 characters left)

Headline:* Please enter a summary of your request (125 characters or less):
(You have 125 characters left)

Submitter's ranking of priority:* What impact does this request have on your ability to use the product?
Select a priority | Priority definitions

Brand:* WebSphere

Product family:* Enterprise Networking

Spotlight
- Announcements
- Give us your feedback

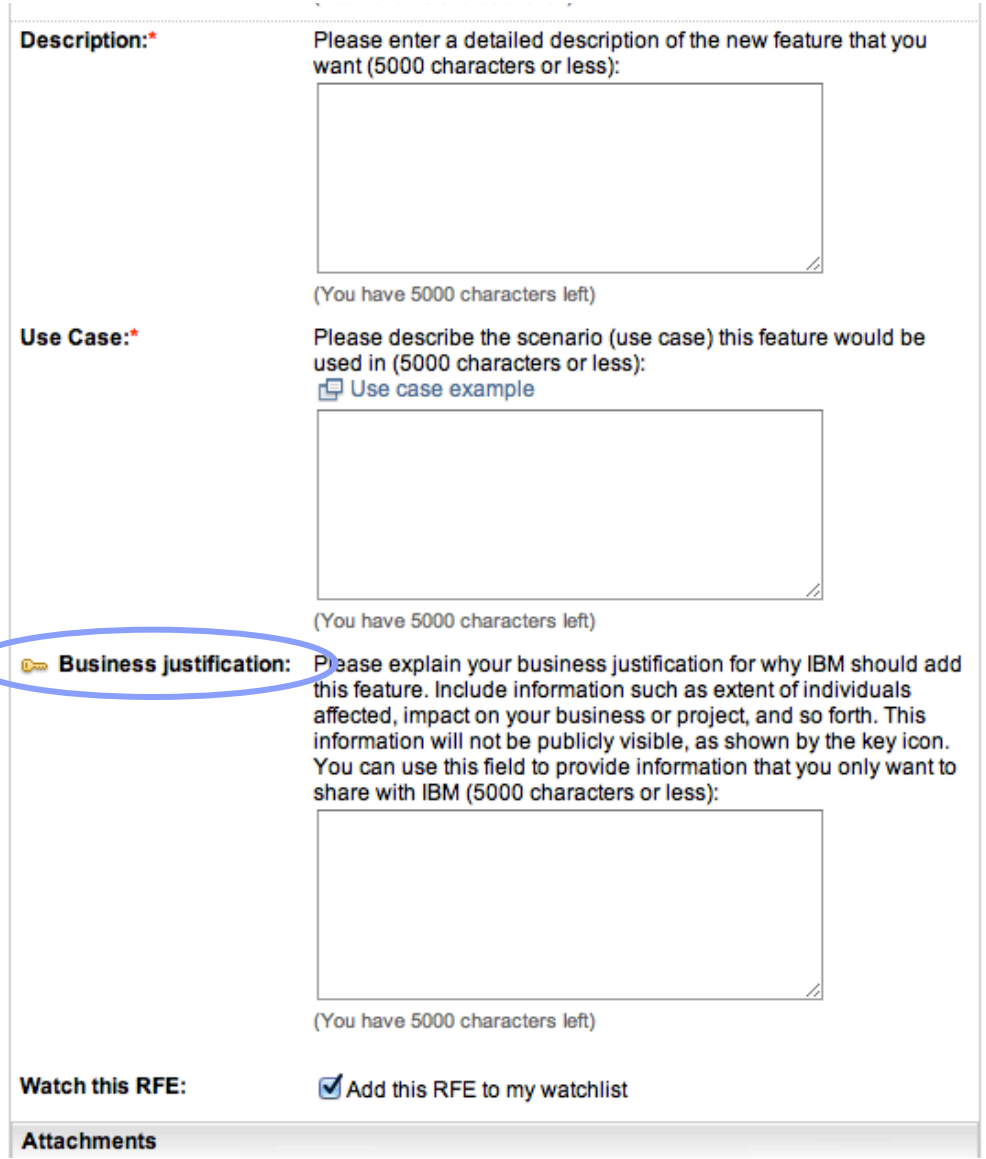
Brands
- All brands
- Informatio n Management
- Rational
- Tivoli
- WebSphere

RFE activities
- Search RFEs
- Submit RFEs

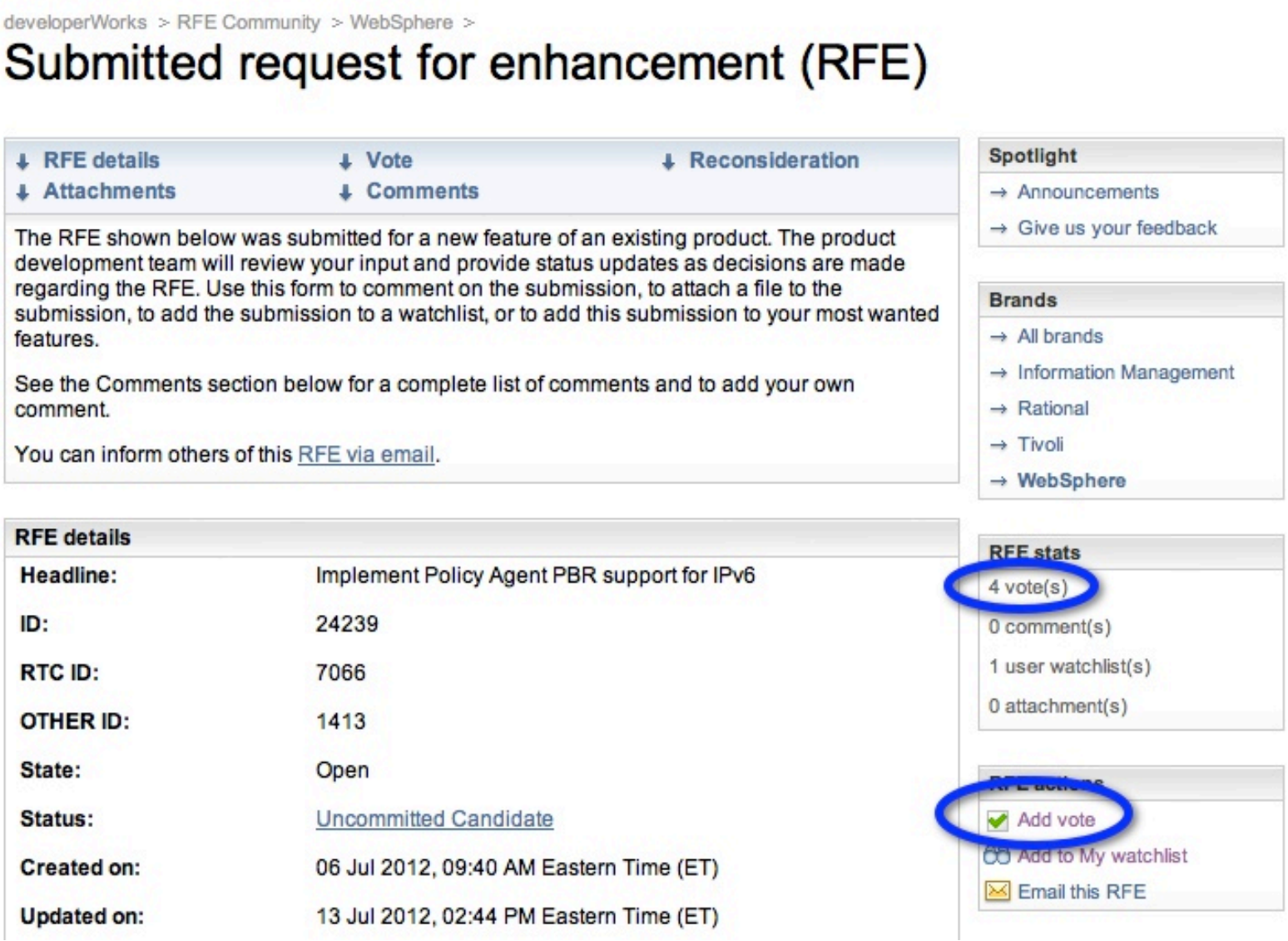
My stuff
- My watchlist
- My votes
- My RFEs
- My group memberships
- My saved searches
- My RSS feeds
- My notifications

Groups
- Group directory

RFE: New Requirements Process for z/OS Communications Server …



RFE: New Requirements Process for z/OS Communications Server …



developerWorks > RFE Community > WebSphere >

Submitted request for enhancement (RFE)

RFE details | Vote | Reconsideration
Attachments | Comments

The RFE shown below was submitted for a new feature of an existing product. The product development team will review your input and provide status updates as decisions are made regarding the RFE. Use this form to comment on the submission, to attach a file to the submission, to add the submission to a watchlist, or to add this submission to your most wanted features.

See the Comments section below for a complete list of comments and to add your own comment.

You can inform others of this RFE via email.

RFE details

Headline:	Implement Policy Agent PBR support for IPv6
ID:	24239
RTC ID:	7066
OTHER ID:	1413
State:	Open
Status:	Uncommitted Candidate
Created on:	06 Jul 2012, 09:40 AM Eastern Time (ET)
Updated on:	13 Jul 2012, 02:44 PM Eastern Time (ET)

Spotlight
- Announcements
- Give us your feedback

Brands
- All brands
- Information Management
- Rational
- Tivoli
- WebSphere

RFE stats
- 4 vote(s)
- 0 comment(s)
- 1 user watchlist(s)
- 0 attachment(s)

RFE actions
- Add vote
- Add to My watchlist
- Email this RFE

RFE: New Requirements Process for z/OS Communications Server …

- **Requirement Tips:**
  - Explain the problem you need solved, not just the requested solution
    - A request for a particular solution may not be feasible, or might take a very long time to deliver
    - By describing the problem to be addressed, we may be able to suggest alternatives that will be immediately beneficial, or a solution that we will be more likely able to implement in the reasonable future
    - Please understand that even in the best case, the delivery of a new function will likely be 2 to 3 years off due to our 2-year release cycle
  - We try to disposition a requirement within a few weeks (typically marking it as an “uncommited candidate” or closing it). During the time it is “Under Consideration” please monitor for our updates. We sometimes need to request more information, recommend alternate solutions, etc., before we can disposition it.
  - If you see other z/OS Communications Server requirements that would be beneficial to your organization, please consider voting for them (as shown on the previous chart).

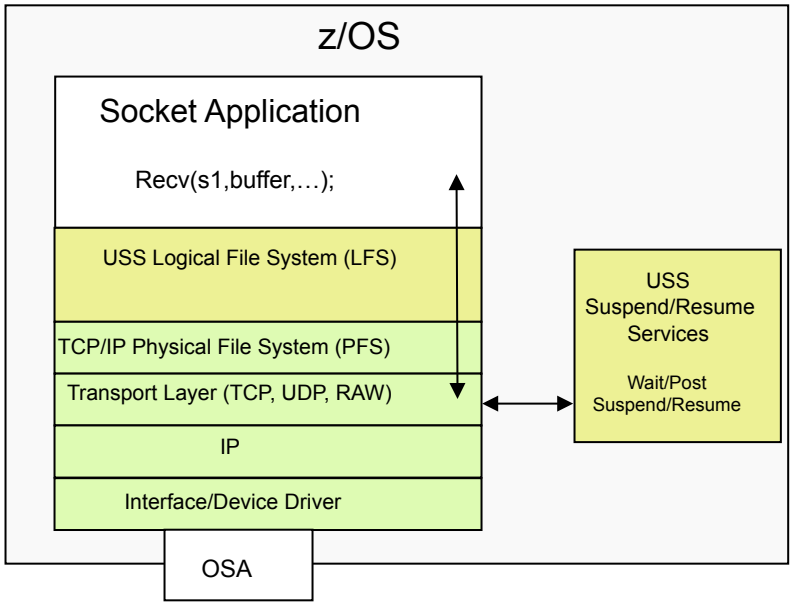
Economics and platform efficiency

Removal of BIND DNS from z/OS

- The z/OS BIND implementation is not current on standards
  - There are RFCs to address BIND-related issues, and those RFCs are incompatible with the z/OS version of BIND
- Most z/OS customers run the DNS function on other platforms, since there is no differentiating advantage to running it on z/OS
- z/OS V1R11's resolver caching function eliminated the need for caching-only name servers, which was the most common use of DNS on z/OS
- We have previously issued Statements of Direction (SOD) for removal of BIND DNS from z/OS
  - V1R11 preview RFA indicated BIND would be removed from a "future" z/OS release
  - V1R13 preview RFA announced that V1R13 was the last release that would ship with BIND DNS
- We will continue to ship the DNS utility programs (nslookup, dig, etc.)

Enhanced Fast Path socket support

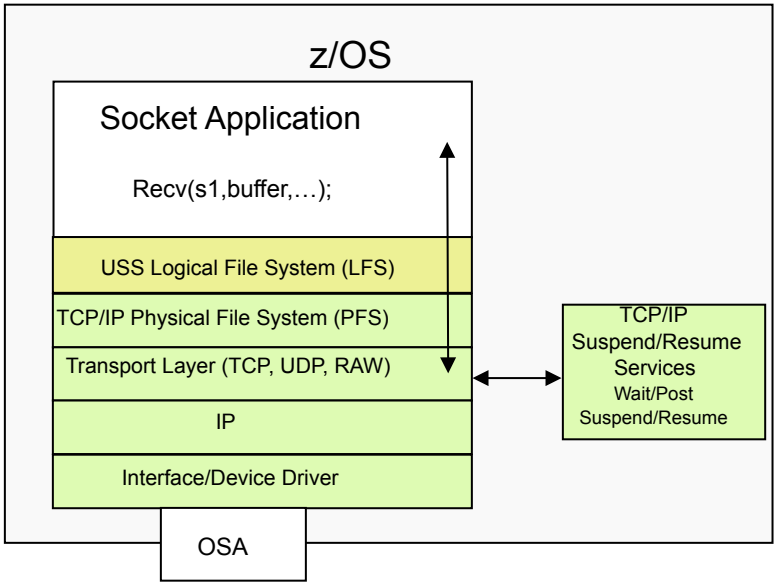
TCP/IP sockets (normal path)



Key attributes:

- Full function support for sockets, including support for Unix signals, POSIX compliance
- When TCP/IP needs to suspend a thread waiting for network flows, USS suspend/resume services are invoked.
  - These services require a space switch into OMVS address
  - Allows thread to be woken up when data arrives or when a signal needs to be delivered

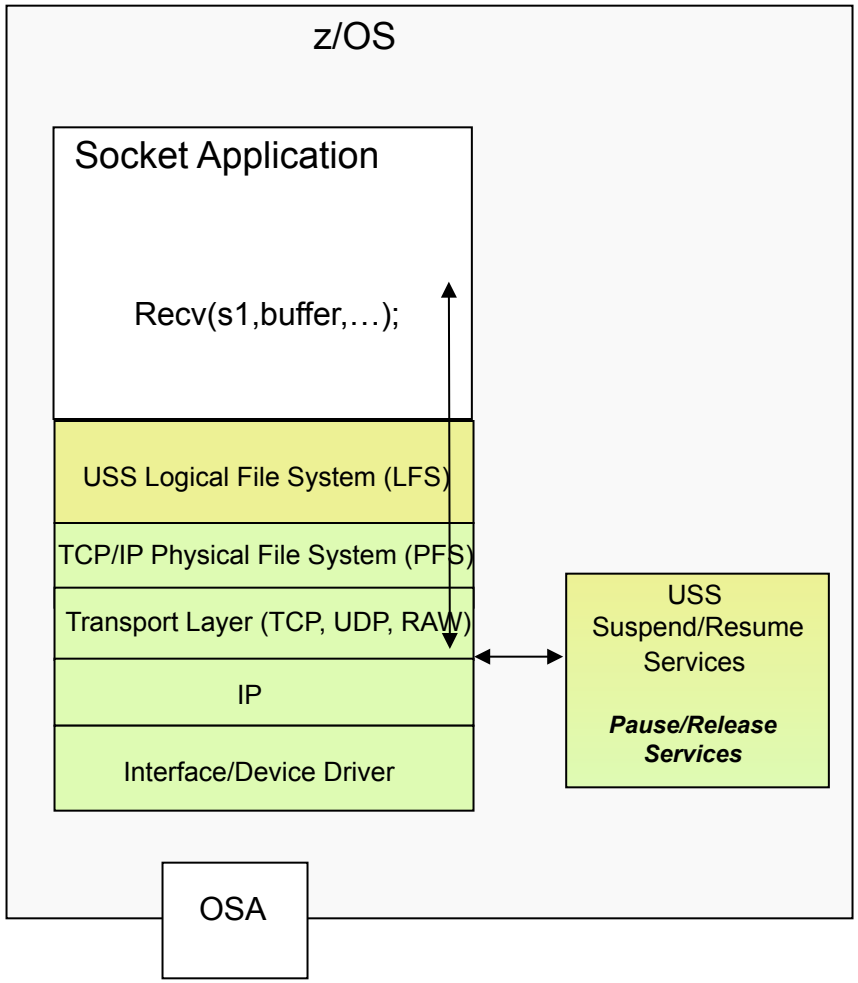
TCP/IP fast path sockets (existing support)



Key attributes:

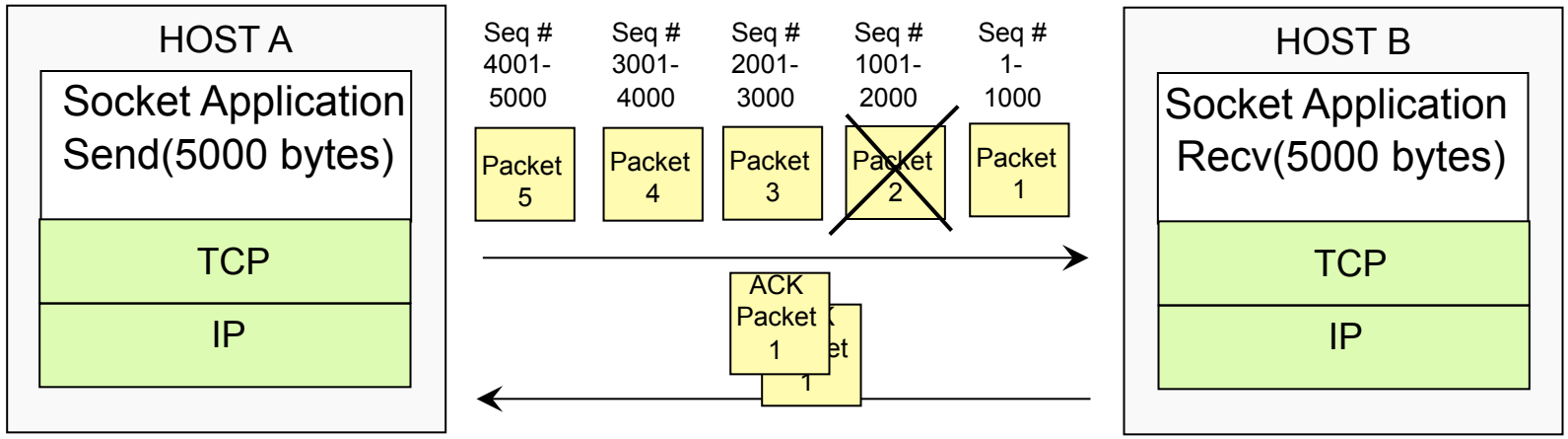
- Streamlined path through USS LFS (reduced path length) for selected socket APIs (send/recv)
- When TCP/IP needs to suspend a thread waiting for network flows, TCP/IP performs the wait/post or suspend/resume inline using its own services
- Not POSIX compliant, no support for Unix signals (other than SIGTERM), no DBX support
- But significant reduction in path length for request/response workloads
- Must be explicitly enabled via socket API options (Iocc#FastPath IOCTL) or via Environment variables *(_BPXK_INET_FASTPATH)*

Enhanced Fast Path socket support (new in z/OS V2R1)



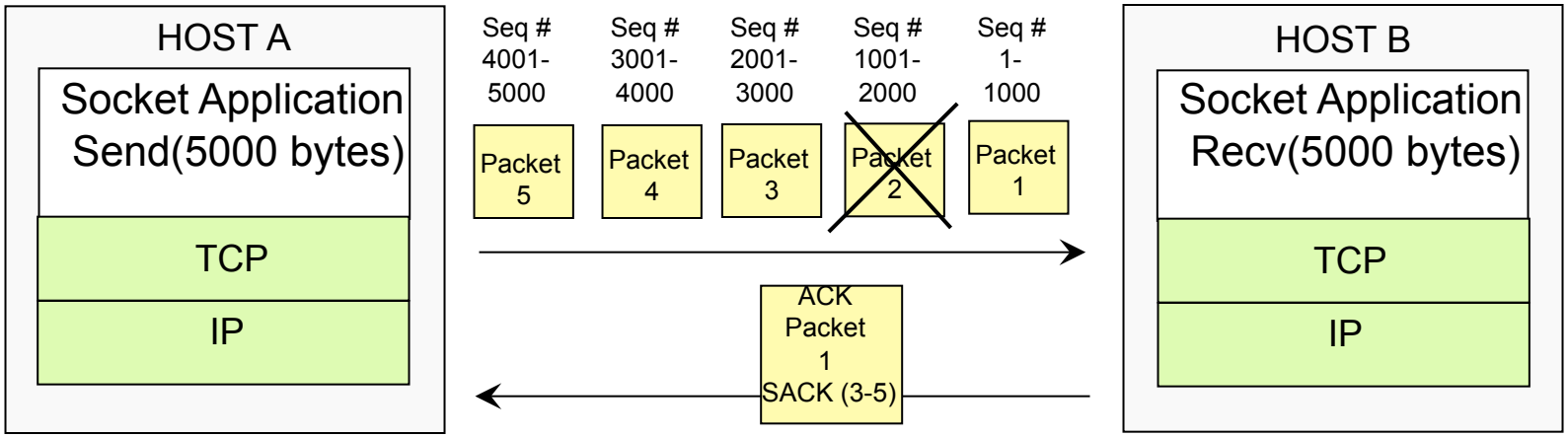
- Provide fast path sockets like performance for *all* sockets applications
  - *Without requiring explicit enablement by the application or the administrator*
  - With POSIX compliance, signals support and DBX support
  - Valid of all socket APIs (except Pascal Socket API)
- Streamlined path through USS LFS for *recv/send* set of APIs
  - Enabled for Recv, recvmsg, recvfrom, send, sendmsg, sendto)
  - But not enabled for read, readv, write,writev, etc.
- New efficient Unix Services for suspend/ resume processing
  - Runs as an extension of TCP/IP stack
  - No need to space switch into OMVS address space
  - Exploits MVS Pause/Release services
    - Optimized path length
    - Minimizes local lock contention for address spaces with large number of threads performing socket calls

Existing TCP acknowledgement/retransmission processing



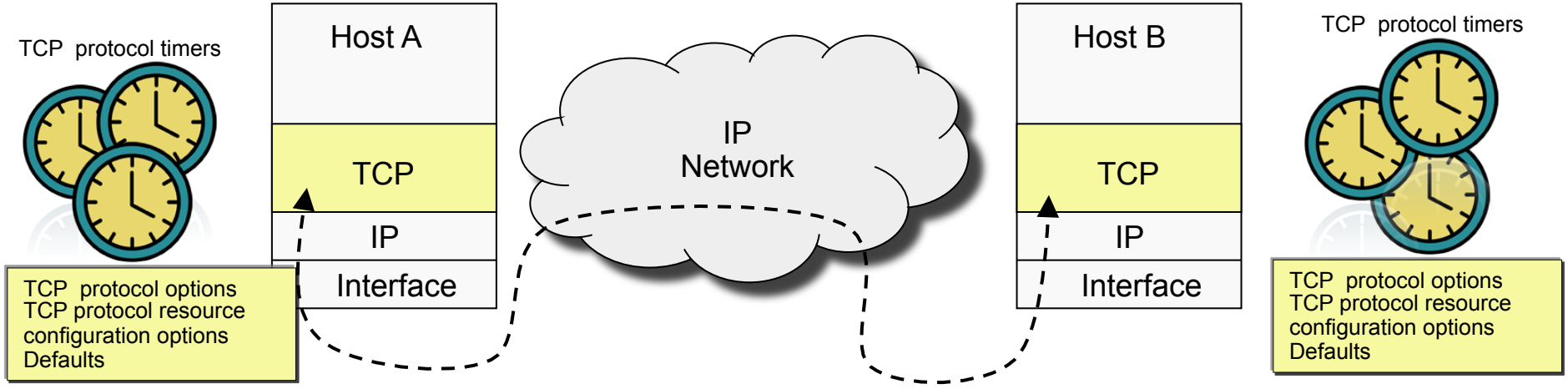
- TCP ack/retransmit processing
  - Receiving host detects gap in data and ACKs the last packet that was received in order (multiple duplicate ACKs sent)
  - Sending host retransmits all packets after the acknowledged packet
- Results in unnecessary overhead
  - Sender retransmits more packets than necessary
  - Receiver burns cycles processing packets it has already received

TCP selective acknowledgements and retransmission processing



- TCP ack/retransmit processing
  - Receiving host detects gap in data and ACKs the last packet that was received in order but includes information about other packets in the sequence that have been received (out of order segments)
  - Sending host can use the selective ACK information to only retransmit the missing packets (i.e. the gaps)
- Based on IETF standards
  - RFC 2018 (Selective TCP Selective Acknowledgment Options) and RFC 3517 (A Conservative Selective Acknowledgment (SACK)-based Loss Recovery Algorithm for TCP)
- More efficient retransmission processing in networks with packet loss
  - But some additional CPU overhead under certain conditions
- SACK capability dynamically negotiated by TCP peers during connection establishment
- New configuration statements to enable/disable SACK processing
  - TCPCONFIG SELECTIVEACK|NOSELECTIVEACK option
  - Default setting: TBD based on performance testing

Enhanced TCP protocol configuration options and default settings



- Background: The TCP protocol has several timers and resource utilization options
  - Influence the protocol's behavior in terms of resources consumed (e.g. memory for buffers, queued connections, etc.)
  - Timers that influence how long the protocol waits before taking an action (timing out connections, retransmitting packets, etc.)
- Problem:
  - Some of these timers and resource utilization options cannot be modified by users
  - Default settings for some of these options are not optimal for today's networks and workloads
- Solution:
  - Externalize key timers and controls via new configuration options in the TCP/IP profile
  - Modify default settings where possible to match best practices recommendations

New TCPCONFIG parameters in TCPIP profile

New TCPCONFIG parameter	*Function*	*Range*	*Default*
FINEWAIT2 (modified, not new)	***Existing Parameter,*** controls how long (number of seconds) TCP connections stay in a FINWAIT2 state. Large number of connections in this state consume system memory. Change is in the range of values that can be specified.	1-3,600 seconds (previous range was 60-3,600 seconds)	600 seconds (unchanged)
TIMEWAITINTERVAL	Controls how long connections are kept in a TIMEWAIT state (occurs after a TCP connection is closed – close initiator goes into this state). Large numbers of connection in TIMEWAIT state consume system memory	0-120 seconds	60 seconds
MAXIMUMRETRANSMITTIME	TCP/IP *stack wide* control for maximum TCP retransmission interval. Effective if no other existing retransmission configuration options (i.e. MAXIMUMRETRANSMITTIME on BEGINROUTES/GATEWAY, ROUTETABLE or Max_Xmit_Time on OSPF_INTERFACE and RIP_INTERFACE)	0-999.99 seconds	120 seconds 15 attempts
RETRANSMITATTEMPTS	New control, specifies the total number of times a segment is retransmitted before aborting a TCP connection	0-15 attempts	15 attempts (same as today)

New TCPCONFIG parameters in TCPIP profile (cont)

New TCPCONFIG parameter	*Function*	*Range*	*Default*
CONNECTTIMEOUT	Specifies the total amount of time to allow before the initial connection times out (for outbound connections from z/OS).	5-190 seconds	75 seconds (previous internal setting was around 3 minutes)
CONNECTINITINTERVAL	Specifies the initial retransmission time out interval in milliseconds	100-3,000 ms	3,000 ms (3 seconds)
KEEPALIVEPROBES	Additional control related to TCP *Keepalive Interval* that can be specified on TCPCONFIG. This new control limits the number of keepalive probes that are sent after the keepalive timer has expired and the connection is not responsive. ***Note:*** Applications issuing setsockopt() TCP_KEEPALIVE will override this TCPCONFIG setting	1-10 probes	10 probes (same as previous releases)
KEEPALIVEPROBEINTERVAL	Related to KEEPALIVEPROBES, indicates the interval between probes. ***Note:*** Applications issuing setsockopt() TCP_KEEPALIVE will override this TCPCONFIG setting	1-75 seconds	75 seconds (same as previous releases)

New TCPCONFIG parameters in TCPIP profile (cont)

New TCPCONFIG parameter	*Function*	*Range*	*Default*
NONAGLE and NAGLE	Nagle's algorithm prevents TCP from sending data packets which are smaller than a full segment unless all previously sent data has been ACKed. For most workloads this is not a problem. However, for some workloads that perform multiple small sends without receiving any inbound data this can lead to significant latency issues. The setsockopt() TCP_NODELAY option allows applications to disable Nagle's algorithm for a given connection. NONAGLE disables Nagle's algorithm for all TCP connections. NAGLE enables Nagle's algorithm for all TCP connections (unless explicitly disabled via the TCP_NODELAY setsockopt()) Notes: ▪ z/OS implements a "Relaxed" Nagle algorithm that allows some small segments to be sent under certain conditions. ▪ This parameter is related to the DELAYACKS/ NODELAYACKS parameter setting. NODELAYACKS is another method for solving latency issues related to Nagle's algorithm on the receiving TCP end point.	N/A	*NAGLE*

New TCPCONFIG parameters in TCPIP profile (cont)

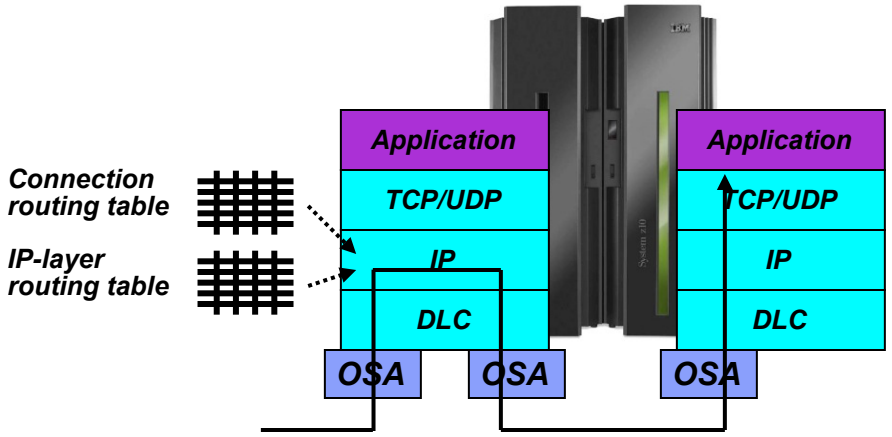
New TCPCONFIG parameter	*Function*	*Range*	*Default*
QUEUEDRTT	Outbound serialization is used when a high latency network is detected. ▪ avoids out of order outbound packets ▪ high latency is defined as an RTT and SRTT of 20 ms This parameter specifies the latency (in milliseconds) used for outbound serialization. Use with care, typically under the direction of IBM service	0-50 ms	20 ms (same as internal setting on previous releases)
FRRTHRESHOLD	Fast retransmit/fast recovery (FRR) engages after 3 duplicate ACKs are received ▪ duplicate ACKs indicate lost or out of order packets ▪ RFC 2581 defines FRR This parameter defines the number of duplicate ACKs required before FRR is engages for a TCP connection. *Use with care!*	1-2048 duplicate ACKs	3 duplicate ACKs (same as internal setting in previous releases)

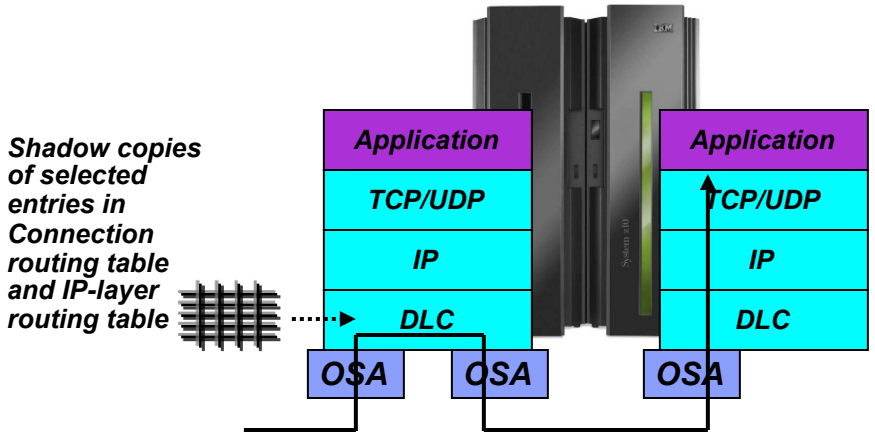
Modified defaults for existing TCPCONFIG parameters in TCPIP profile

TCPCONFIG parameter	*Function*	*New Default*	*Old Default*
SOMAXCONN	Controls how big the TCP backlog queue can be for a listening socket. The backlog value is specified on the listen() socket API. SOMAXCONN controls the maximum value that can be specified. In today's environments and workloads the previous default of 10 was too restrictive and needed to be modified by most customers. The default used to be 10 and is now changed to be 1024. Note: Memory is not pre-allocated based on this setting	1024 connections	10 connections
TCPRCVBUFRSIZE and TCPSENDBUFRSIZE	Control how much data can be stored by local TCP stack on a connection basis (data waiting to be read by application or sent to partner application) Defaults used to be 16K, now changed to 64K (note: memory is not allocated unless needed) These buffer sizes can also be programmatically changed via setsockopt() API on a socket basis.	64K	16K

QDIO acceleration coexistence with IP filtering

- QDIOACCELERATOR
  - Indicates that inbound packets that are to be forwarded should be routed directly between a HiperSockets device and an OSA-Express device in QDIO mode.
  - Function is called “QDIO Accelerator” or “Hipersockets Accelerator”
  - Affected packets are routed without touching the TCP/IP stack
  - Improves performance and reduces processor usage for such workloads
- IPSECURITY
  - Enables IP filtering in the TCP/IP stack
- It has not been possible to specify these two statements together
  - The stack’s IP filters cannot be applied to QDIOACCELERATOR traffic since the routing occurs “below” the stack
  - Specifying both in the same profile results in an error message





QDIO acceleration coexistence with IP filtering …

- However, there are valid cases where it makes sense to specify QDIOACCELERATOR with IPSECURITY - cases where IP filtering is not needed for the QDIO Accelerator traffic:
  - The routed traffic is destined for a target that's doing its own endpoint filtering
  - IPSECURITY is only specified to enable IPSec on the local node

- V2R1 will allow QDIOACCELERATOR to be specified with IPSECURITY in the TCPIP profile under certain conditions:

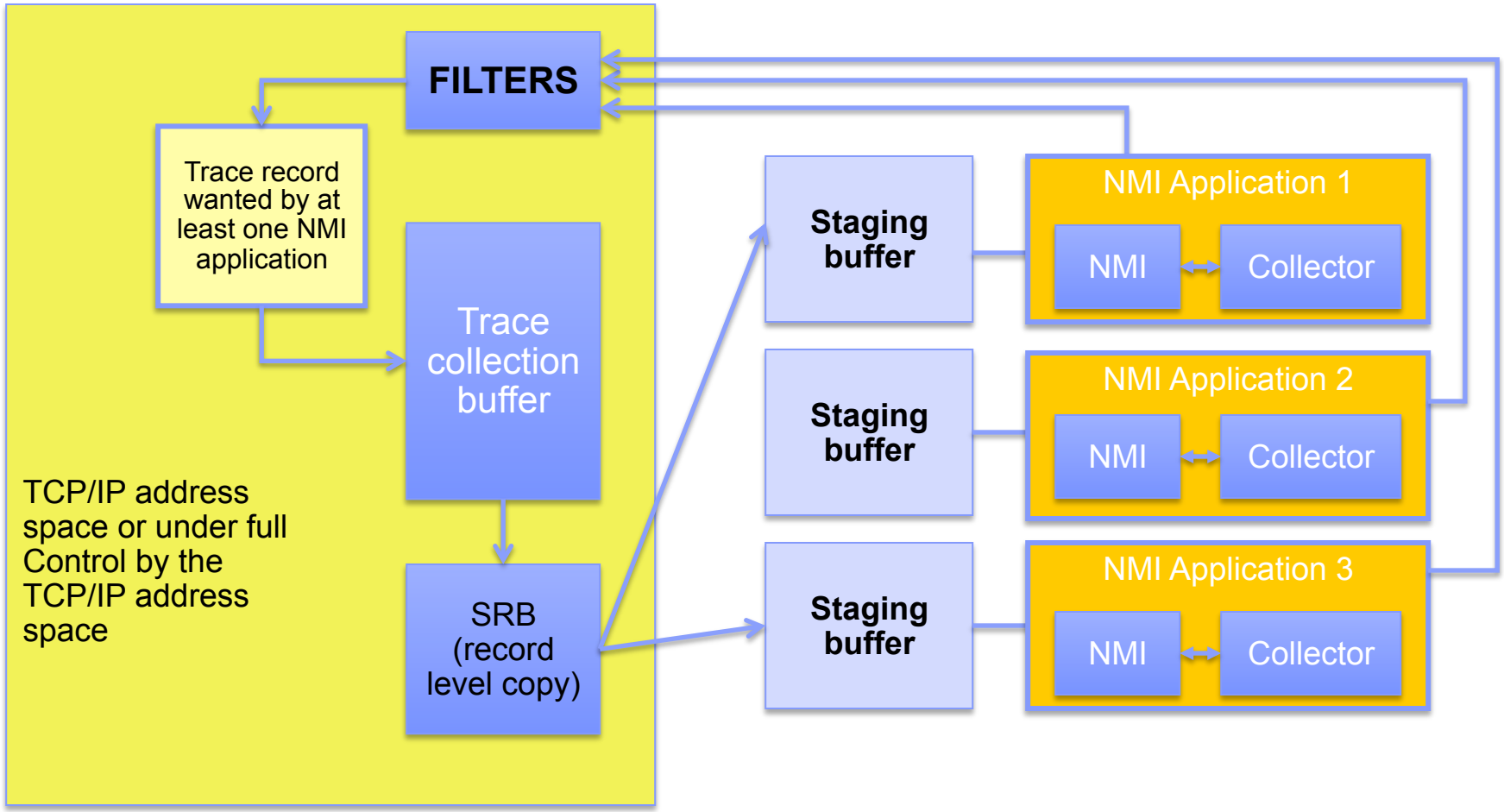
IP filter rules & defensive filter rules permit all routed traffic?	**IP filter rules & defensive filter rules require routed traffic to be logged?**	**QDIO acceleration permitted?**
N	N	**N**
N	Y	**N**
Y	Y	**N**
Y	N	**Y**
Sysplex Distributor traffic always forwarded		

Application / Middleware / Workload enablement

Real-time application-controlled TCP/IP trace NMI

- Current TCP/IP network management interface (NMI) for real-time trace data
  - Allows multiple network management applications to obtain a copy of a packet trace that has been activated via console commands
    - Every application gets the same copy of the trace data
    - No ability to customize what data is collected by each application, a single trace may be active at any time
    - No ability to coordinate multiple distinct trace options
- New NMI that provides support for multiple concurrent and independent trace operations.
  - Applications use granular API-based trace open, filter definition, activation, deactivation, and trace close functions to control the traces they collect
  - Supports multiple independent trace operations running concurrently, each with its own set of trace options
  - Includes packet trace and data trace
  - Operate independently of each other and of the operator controlled trace operations that are started and controlled via MVS console commands
- Allows the inclusion of the following real-time NMI traces in a single trace data stream
  - Packet trace
    - Optionally includes IPSec data in the clear under RACF authorization
  - Data trace
    - Optionally includes AT-TLS data in the clear under RACF authorization
  - New RACF resource profiles for access to trace data

Real-time application-controlled TCP/IP trace NMI …



- Notes about this diagram are on the next chart

Real-time application-controlled TCP/IP trace NMI …

NOTES

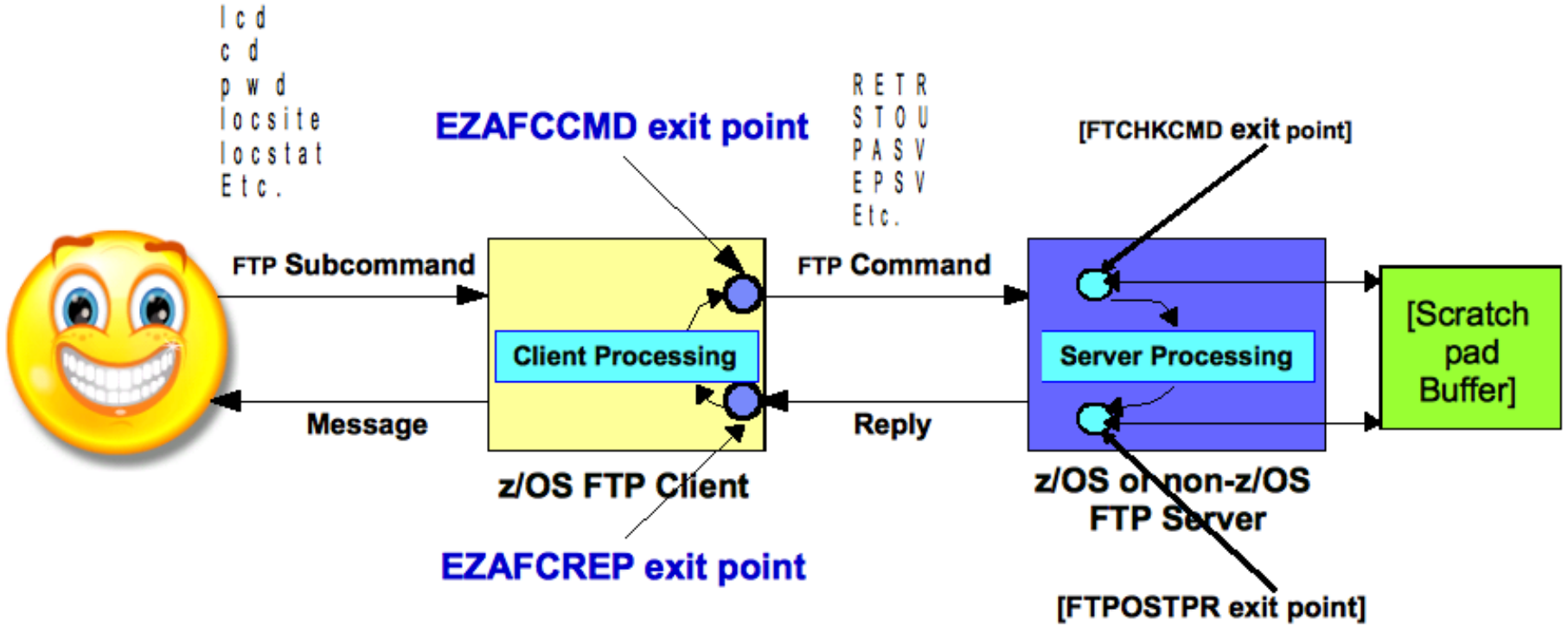
- All trace records are stored in a TCP/IP-controlled collection buffer in TCP/IP 64-bit common storage
- Trace records copied by TCP/IP into a staging buffer for each NMI application
  - Circular set of trace records
  - Uses a 64-bit shared memory object for each staging buffer (NMI application specifies the size of the memory object)
  - One shared object per open trace instance
  - Obtained in application's address space
  - Stack address space gets affinity to shared object
  - Trace records copied to trace instance staging buffers by an SRB running in the TCP/IP address space
  - If records lost (for example because trace wraps before application collects records), application will be notified
- Application collector function can obtain trace data at any point in time
  - Does not have to wait for buffer to become full
  - Invokes an NMI function to obtain the data

FTP client security user exits

- You can use FTP server user exits to limit access to an FTP server
- Currently, a system administrator has no way of controlling FTP client commands or other aspects of the processing done in the z/OS FTP client. Examples of tasks that a system administrator might desire:
  - Preventing a dataset (that the user should have at least limited access to) from being moved from the z/OS host
  - The ability to inspect or modify the dataset names specified by FTP client users for inbound and outbound file transfers
  - The ability to cancel an FTP client address space if that client is in the process of sending an "unauthorized" FTP command
- To address this, V2R1 will implement two FTP client user exit points:
  - FTP command user exit – EZAFCCMD
    - Receives control just before an FTP command is being sent
    - Enables the exit to inspect the command, optionally modify the command arguments, reject the command, or request the FTP client session be terminated
  - FTP reply user exit – EZAFCREP
    - Receives control when an FTP server reply arrives
    - Allows the exit to analyze the results of commands sent to the server, and request the FTP client session be terminated

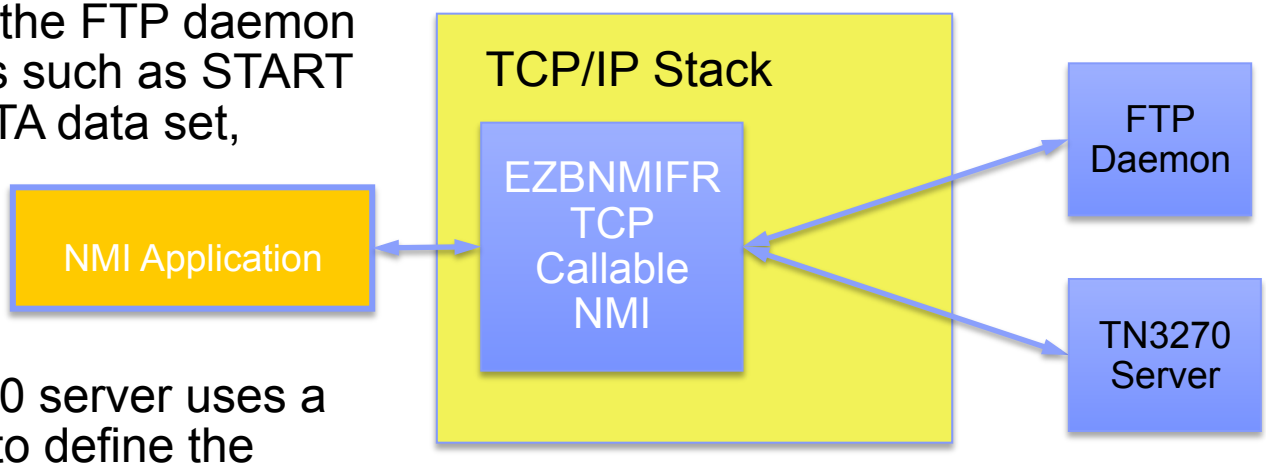
FTP client security user exits …

- EZAFCCMD user exit called whenever the FTP client is about to send an FTP command to the server.
  - The call of this user exit is done before the command is converted to ASCII
- EZAFCREP user exit called whenever the FTP client receives a reply line over the control connection
  - The call of this exit is done after the reply has been converted to EBCDIC



NMI and SMF enhancements for TCP/IP applications

- Users can configure the FTP daemon with various methods such as START parameters, FTP.DATA data set, TCPIP.DATA data set, UNIX environment variables, etc.



- The z/OS CS TN3270 server uses a configuration profile to define the multiple listening ports for Telnet protocols, to define options, and for mapping client sessions to LU names for the interface with VTAM.
- z/OS V2R1 Communications Server provides a way for network management applications to obtain FTP daemon configuration data and TN3270 server configuration data using the existing TCP/IP callable NMI, EZBNMIFR, with new request types.
  - You can also obtain type 119 SMF records for FTP daemon configuration data and TN3270 profile information.

NMI and SMF enhancements for TCP/IP applications …

NOTES

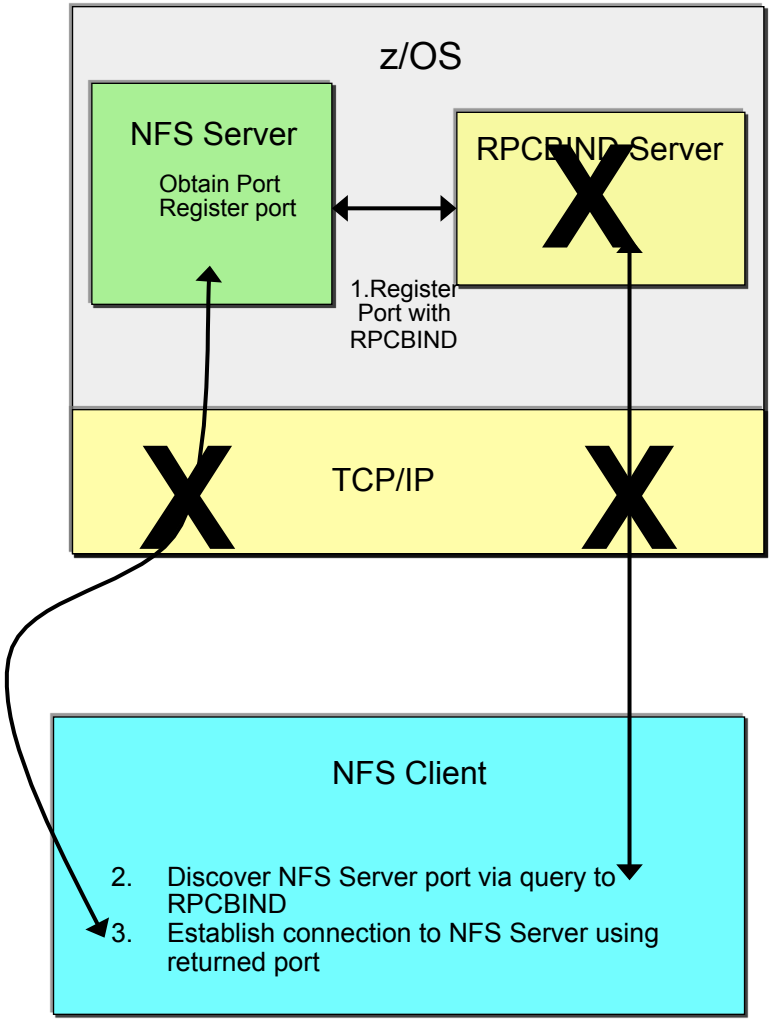
- The new NMI request type GetFTPDaemonConfig allows network management applications to programmatically obtain configuration data of active FTP daemons:
  - New TCP/IP callable NMI request type, GetFTPDaemonConfig - Get FTP Daemon Configuration Data
  - The NMI request requires a single filter which is the ASID of the FTP server whose configuration data is requested.
  - The NMI response includes a single record with all of the configuration data, which includes the FTP.DATA parameters, the START parameters, the UNIX environment parameters, and the TCPIP.DATA and FTP.DATA data set names.
  - Network management applications to call EZBNMCFG to get the configuration data of an FTP daemon can be written using C/C++ or Assembler.
- V2R1 also allows you to obtain a type 119 SMF record for FTP daemon configuration data.:
  - The new type 119, subtype 71 SMF record contains the FTP daemon configuration data and will be written after the FTP daemon finishes initialization and is listening on the listening port
  - Sections in this SMF record will be in same format as the corresponding sessions in the response buffer returned by the GetFTPDaemonConfig NMI request
  - A new server FTP.DATA parameter SMFDCFG is used to control whether to generate the above SMF record
  - As with other TCP/IP SMF records, this SMF record can either be written to an SMF data set, or written to a dataspace and accessed through the Network Management Real Time Interface for SMF Events (SYSTCPSM).

NMI and SMF enhancements for TCP/IP applications …

NOTES

- The new NMI request type GetTnProfile allows network management applications to programmatically obtain complete TN3270 initial profile information.
- V2R1 also includes a new type 119, subtype 24 SMF record which provides the initial TN3270 profile information, as well as information about replacement of the profile caused by VARY TCPIP,Telnet,OBEYFILE processing.
  - As with other TCP/IP SMF records, this SMF record can either be written to an SMF data set, or written to a dataspace and accessed through the Network Management Real Time Interface for SMF Events (SYSTCPSM).
- Network management applications can use a combination of the GetTnProfile request and the new SMF 119 event records that are created during VARY TCPIP,Telnet,OBEYFILE command processing to monitor replacements of the Telnet profile.

RPCBIND recycle notification



- RPCBIND allows RPC services to register their services and ports with it and responds to RPC client requests asking where an RPC service is registered.
  - NFS client/server are one of the main exploiters
- When the RPCBIND server is recycled, all registration information for the RPC services is lost and RPC lookup requests from clients cannot locate the service
  - The RPC service must register again with RPCBIND.
    - In the case of NFS this means recycling the NFS server
      - This causes all existing NFS client connections to be terminated
- In V2R1, the RPCBIND server will now raise an ENF signal when either RPCBIND is started or is stopping allowing RPC servers to dynamically re-register all their RPC ports with RPCBIND

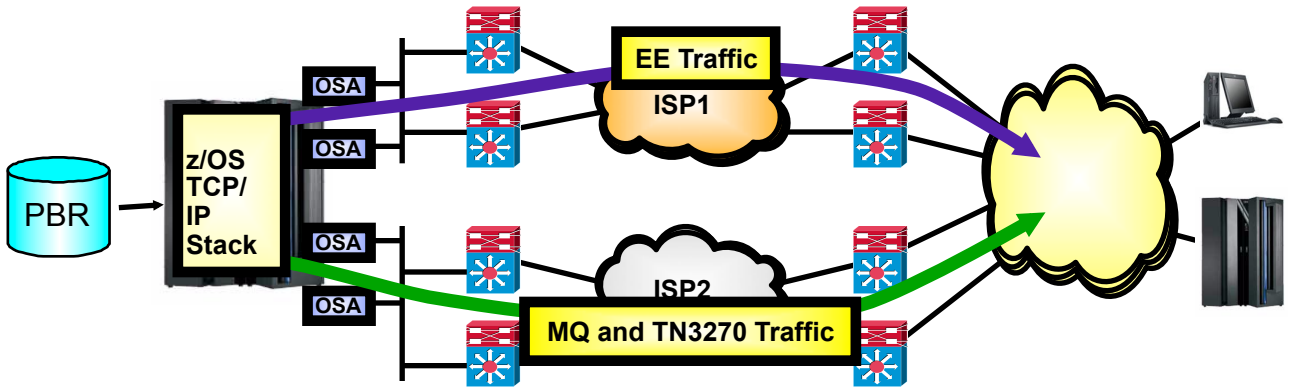
RPCBIND recycle notification (cont)

- In V2R1, the RPCBIND server will now raise an ENF signal when either RPCBIND is started or is stopping.
  - The RPCBIND server will send an ENF signal when it has started and is prepared to accept registrations from RPC applications. RPC applications can monitor this ENF signal and register again with the RPCBIND server should it be stopped and restarted for any reason.
    - The z/OS NFS server will exploit this ENF in z/OS V2R1 to dynamically reregister its ports
      - Without affecting existing connections to the NFS server
  - The RPCBIND server will send an ENF signal when it is stopped or cancelled. RPC applications can monitor this ENF signal and take action when the RPCBIND server is not available to RPC clients.
  - RPCBIND will implement a new ENF signal for event code 80, with qualifier ENF80_RPC_EVENT for indicating when RPCBIND has completed initialization or is terminating.
- ***Note:*** This enhancement is *only* available with the RPCBIND server
  - If you are curently using the PORTMAP server (MVS or z/OS Unix version) you need to migrate to RPCBIND to exploit this enhancement
    - RPCBIND is an enhanced PORTMAP server that includes support for both IPv4 and IPv6

Policy-based outbound routing of traffic that originates on z/OS

- **What does Policy Based Routing (PBR) do?**
  - Choose first hop router, outbound network interface (including VLAN), and MTU
  - Choice can be based on more than the usual destination IP address/subnet
    - With PBR, the choice can be based on source/destination IP addresses, source/destination ports, TCP/UDP, etc.
- **Allows an installation to separate outbound traffic for specific applications to specific network interfaces and first-hop routers:**
  - Security related
  - Choice of network provider
  - Isolation of certain applications
    - EE traffic over one interface
    - TN3270 traffic over another interface
  - PBR policies will identify one or more routes to use
    - If none of the routes are available, options to use any available route or to discard the traffic will be provided

Enabled for IPv6 support in V2R1



PBR technologies are a great companion to VLAN technologies for separation of traffic over different networks or network providers.

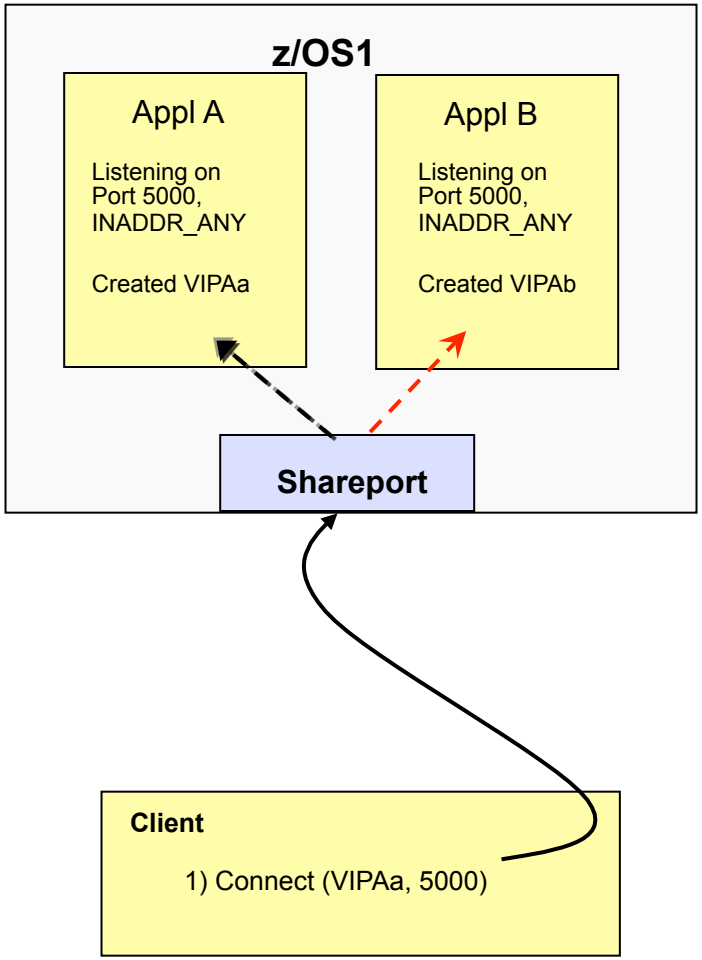
Please fill out your session evaluation

- z/OS Communications Server Technical Update, Part 1
- Session # 12843
- QR Code:

Availability

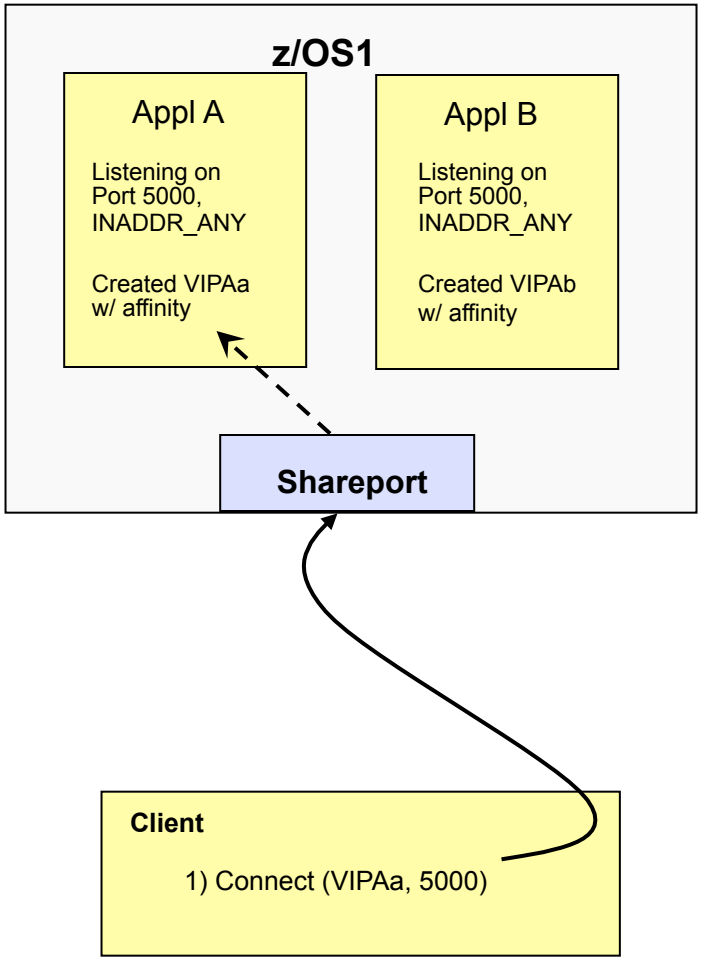
Application Instance DVIPA affinity support

- Application Instance Dynamic VIPAs
  - DVIPAs defined dynamically to represent an instance of a TCP server application running in the sysplex
  - Each application instance has its own DVIPA address
  - Define via VIPARANGE statement – activated
    - Implicitly when application binds listening socket to the DVIPA (explicitly via bind() API or BIND keyword on PORT reservation statement)
    - Explicitly via socket API IOCTL (SIOCSVIPA and SIOCSVIPA6) or MODDVIPA utility program via JCL
- Problem scenario:
  - Multiple instances of an application listening to same port, bound to INADDR_ANY and explicitly defining application instance DVIPAs in the same z/OS image
    - Incoming connections to one of these DVIPAs may be routed to ***any*** application listening on the port via the SHAREPORT
      - This can create problems with client connections ending up on the wrong application server instance



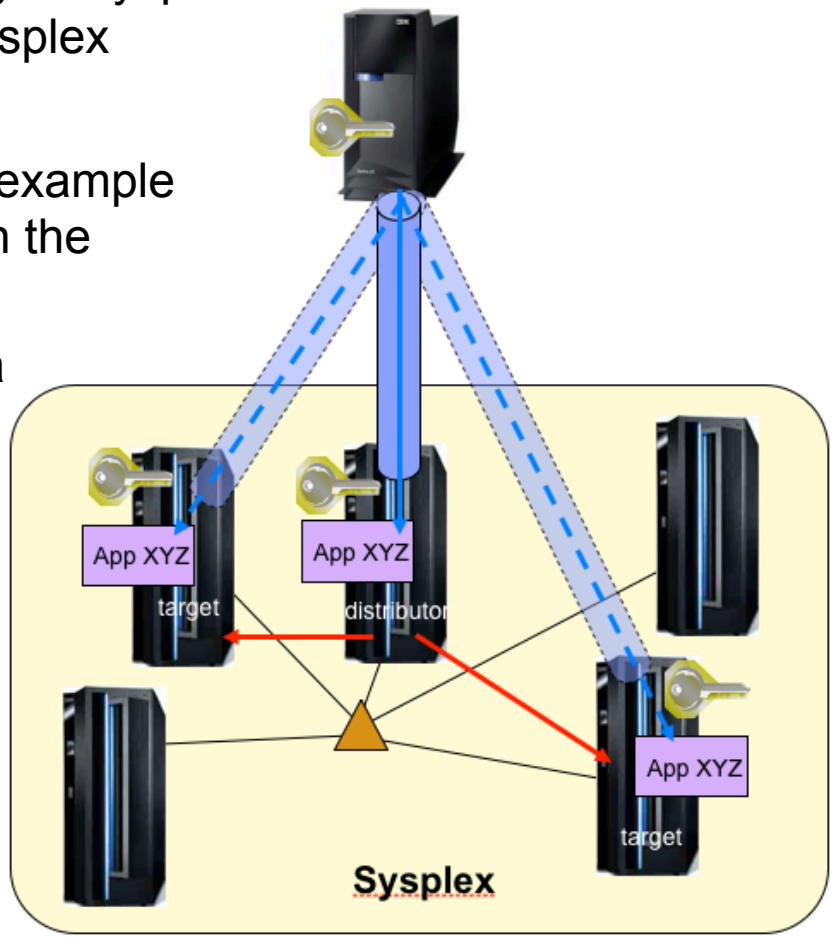
Affinity for application-instance DVIPAs

- Solution:
  - Provide a capability to create an application instance DVIPA with affinity
    - DVIPA affinity determined by creating address space
    - Incoming connections to an “affinity” DVIPA will be handled in a special manner by SHAREPORT processing
      - When multiple listening sockets for the target port are available find the listening socket owned by the address space that created the DVIPA
      - If an address space with affinity is not found with a listening socket on the target port then route the connection to any listening socket that can accept it (i.e. bound to INADDR_ANY
        - Allows the DVIPA to be used by other applications trying to reach that z/OS image
        - Only works while the DVIPA is still active
  - Support to create DVIPA with affinity to be provided on
    - Socket APIs (SIOCSVIPA and SIOCSVIPA6 IOCTLs)
    - MODDVIPA utility program
  - Allows multiple cloned application servers to coexist in the same z/OS image and same well known port, while bound to INADDR_ANY
    - And still achieve isolation via Application Instance DVIPAs



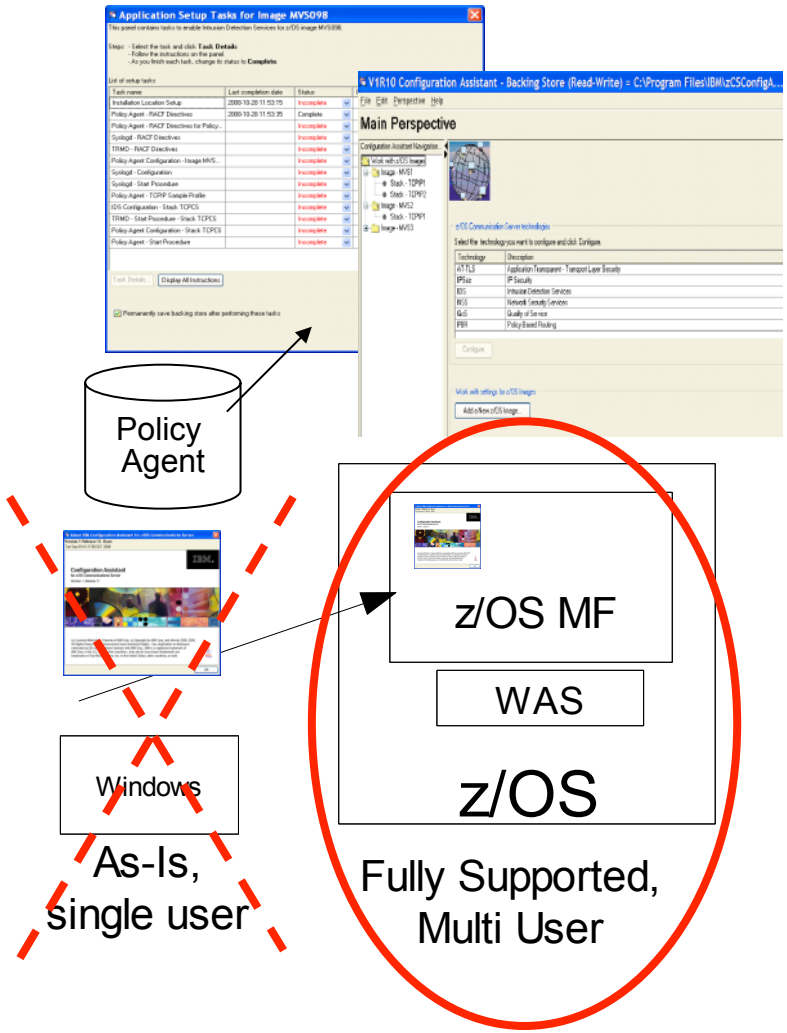
Sysplex-Wide Security Associations for IPv6

- Sysplex-Wide Security Associations (SWSA) allow IPSec-protected traffic to be distributed through a sysplex while maintaining end-to-end security to all sysplex endpoints.
- Security associations and characteristics (for example encryption) are moved and distributed through the sysplex with the target programs
  - VIPA Takeover: Ability of an SA to follow a DVIPA when it moves from one stack to another
  - Distribution: Ability of a target stack to use an IPSec SA that was negotiated on its behalf by the Sysplex distributor stack.
- Currently supported only for IPv4
- V2R1 will complete the SWSA capability by adding IPv6 support



Simplification

Review: IBM Configuration Assistant for z/OS Communications Server



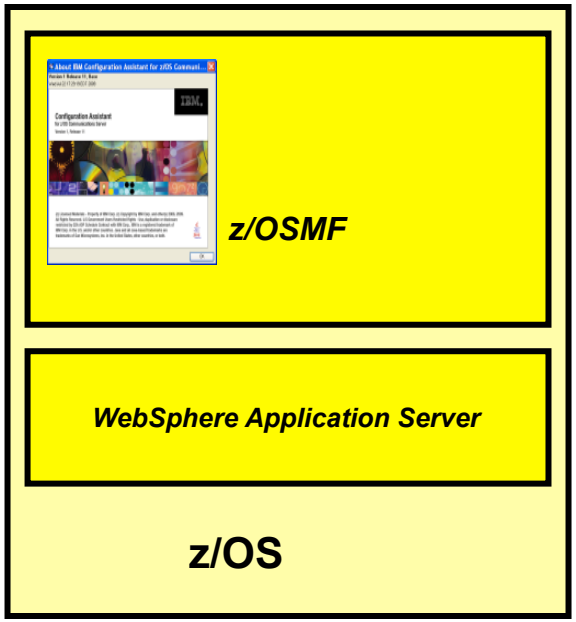
- As of z/OS V1R11, IBM Configuration Assistant for z/OS Communications Server is integrated with z/OS Management Facility (z/OSMF)
  - z/OSMF version is integrated into the product and runs on z/OS.
  - z/OSMF version is officially supported.
  - Supports policy definitions for IPSECURITY (IP filters, IPSec), AT-TLS, IDS, Network QoS, etc.
- The standalone Windows version is still available, but is made available as-is, without any official support:
  - http://www.ibm.com/support/docview.wss?rs=852&context=SSSN3L&dc=D400&uid=swg24013160&loc=en_US&cs=UTF-8&lang=en&rss=ct852other

    or
  - http://tinyurl.com/cgoqsa

Statement of Direction: IBM Configuration Assistant for z/OS Communications Server

z/OS V1.13 is planned to be the final release for which the IBM Configuration Assistant for z/OS Communications Server tool that runs on Microsoft Windows will be provided by IBM. This tool is currently available as an as-is, nonwarranted web download. Customers who currently use Windows-based IBM Configuration Assistant for z/OS Communications Server tool should migrate to the z/OS Management Facility (z/OSMF) Configuration Assistant application. The IBM Configuration Assistant for z/OS Communications Server that runs within z/OSMF is part of a supported IBM product and contains all functions supported with the Windows tool.

Configuration Assistant for z/OS Communications Server



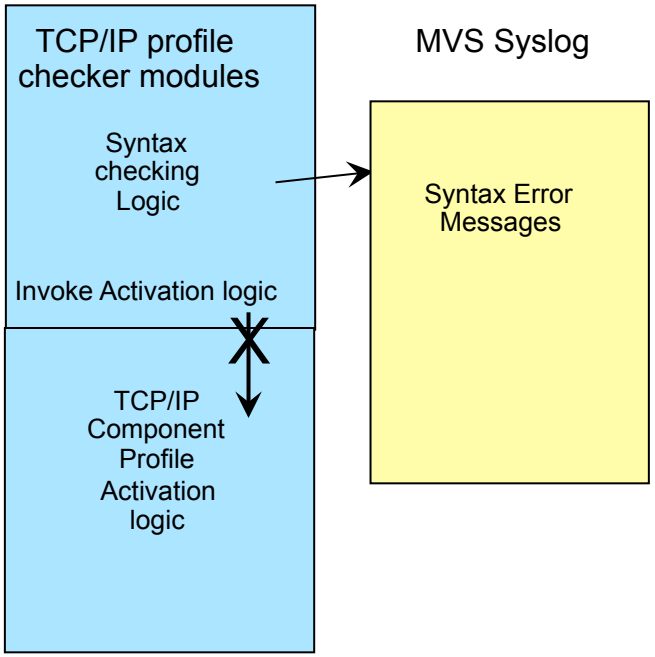
Interface for Comm Server policy based definition, installation and activation

- Re-writing from AUIML (Abstract User Interface Markup Language) to Javascript using Dojo toolkit
  - More consistent look and feel with z/OSMF
  - Better performance
    - More work done on the client's browser
    - Less processing on the z/OSMF host
  - Uses Web 2.0 model based on RESTful services defined between the client side browser and the Config Assistant running on the z/OSMF server
    - May be able to leverage and externalize a subset of these services for additional use cases in the future

Check TCP/IP profile syntax without applying configuration changes

- There is currently no easy way to validate TCP/IP profile syntax
  - Original TCP/IP profile during stack activation
  - V TCPIP,,OBEY processing
- Syntax errors can lead to undesirable results:
  - Partial profile put into effect - could lead to unintended outage
  - Changes caused by earlier statements might need to be undone before processing a repaired profile
- New command will trigger the reuse of the existing TCP/IP profile parser to check the syntax:

V TCPIP,,SYNTaxcheck,dsname




```
EZZ0065I VARY SYNTAXCHECK COMMAND BEGINNING
 ...
<error messages as syntax errors encountered>
 ...
EZZ0064I VARY SYNTAXCHECK FOUND ERRORS: SEE PREVIOUS MESSAGE
EZZ0065I VARY SYNTAXCHECK COMMAND COMPLETE
```

Check TCP/IP profile syntax without applying configuration changes …

- The VARY SYNTAXCHECK command requires an active TCP/IP stack at the same level as the intended target system
  - Can be a development or test system, does not need to be issued on the target system
    - Note: If system symbolics are being used they will be resolved based on the system symbolic configuration of the system the command is issued on
- Can limit access to the command via SAF profiles
- Processes all INCLUDE files specified (even nested ones, just like OBEYFILE)
- Will only flag syntax errors, not semantic (configuration) errors
  - Does not validate TCP/IP profile statements against the current active configuration (e.g., issuing a DELETE PORT for a port that is not currently defined will not be flagged)
- No updates are applied to the active configuration

IPv4 INTERFACE statement for HiperSockets and static VIPAs

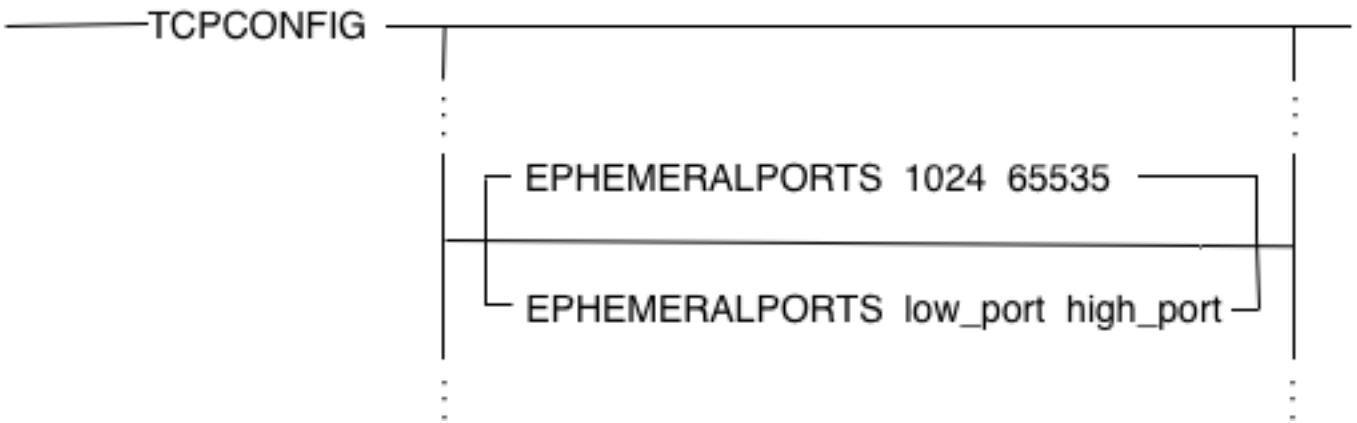
- In z/OS V1R4, a new TCP/IP PROFILE statement to define IPv4 network interfaces was introduced – the INTERFACE statement
  - One statement for all interface-related attributes for IPv6 network interfaces
  - IPv4 faces continued to require use of DEVICE, LINK, HOME, and optionally BSDROUTINGPARMS statements to define all the attributes of an IPv4 network interface
- z/OS V1R10 extended the use of the INTERFACE statement to IPv4 QDIO network interfaces
  - Non-QDIO IPv4 network interfaces continued to require the old configuration syntax
- z/OS V2R1 allows the INTERFACE statement to be used to configure IPv4 HiperSockets and static VIPAs
  - Provides a more straightforward way of configuring the source VIPA for these IPv4 interfaces.
  - Allows for configuration of multiple VLANs from the same TCP/IP stack for a single HiperSockets CHPID for both IPv4 and IPv6.

```
INTERFACE HSINTF1
    DEFINE   IPAQENET
    IPADDR   200.16.1.1/24
    CHPID    FE
    SOURCEVIPAINTERFACE VIPA4811L
```

  - Consider migrating to INTERFACE statements where supported
    - DEVICE/LINK statements should only be required for legacy DLC's (LCS, Claw, etc.)

User control of ephemeral port ranges

- Increased security requirements and port controls on firewalls, etc. are requiring endpoints to use specified ranges for ephemeral ports.
- A new parameter on the TCPCONFIG and UDPCONFIG configuration statements will allow users to specify the range of ephemeral ports assigned at bind time.
- Any ports within the EPHEMERALPORTS range that are already reserved (via other configuration statements) will be skipped during assignment of ephemeral ports from the range



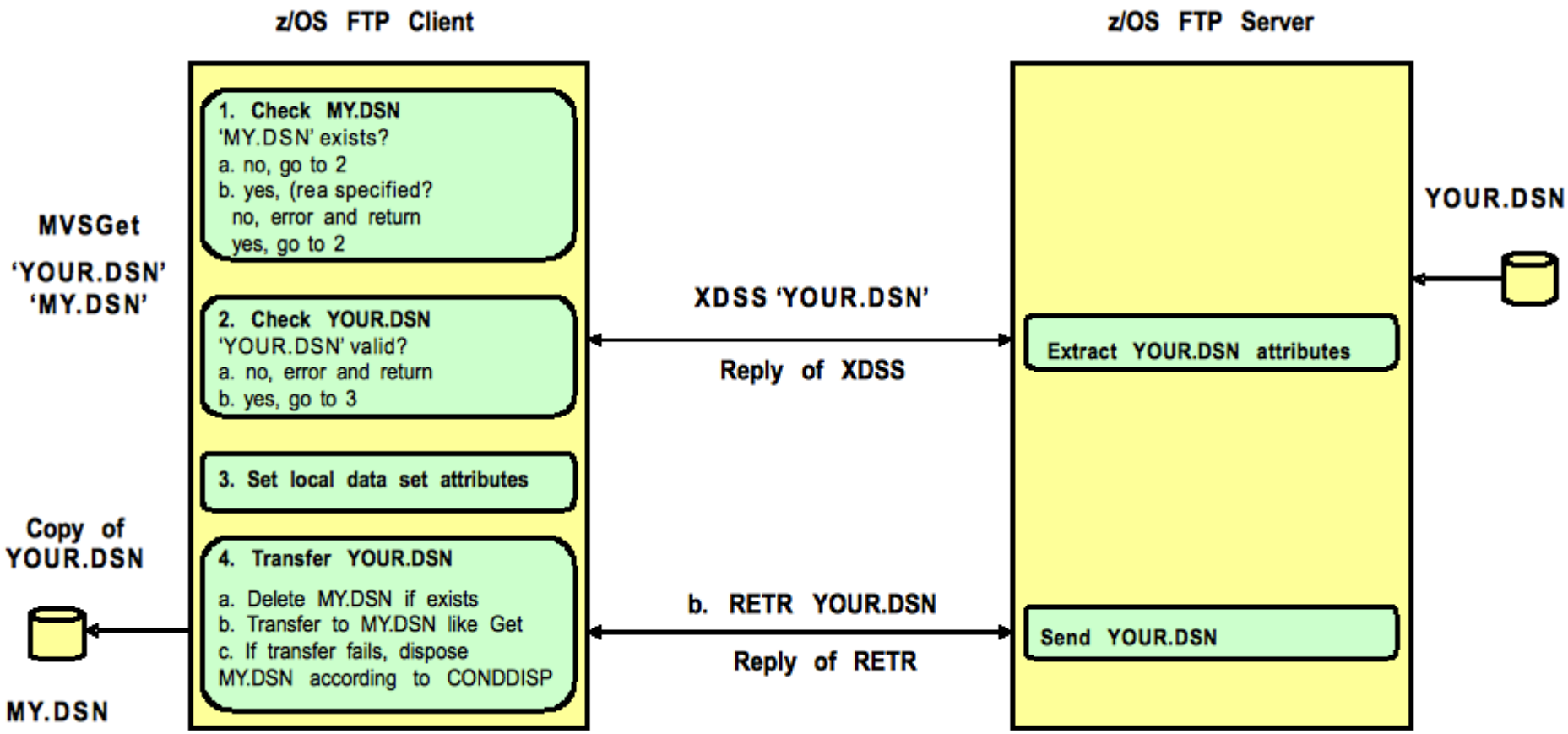
Simplify FTP transfer of datasets between z/OS systems

- When you log into the z/OS FTP server with the z/OS FTP client to get an MVS data set from the server, you have to know a lot about the source data set. You also have to follow many steps to complete the transfer
  - Determine attributes of the source data set:
    - Record format information (LRECL, RECFM, BLKSIZE)
    - SPACE information (allocation units, primary and secondary allocation)
    - Data set type (sequential, PDS or library – and if PDS: number of directory blocks)
  - Issue one or more LOCSITE subcommands to set these attributes on the client side
  - For sequential data sets: issue a GET subcommand
  - For PDS or library data sets: issue an lmkdir subcommand, followed by an MGET * subcommand
  - The transfer might result in the source and target data sets having mismatched attributes if the target data set already exists. This increases the likelihood of data loss due to wrapping, truncation, space constraints, and so on
- The same problem exists when you want to transfer MVS data sets from the z/OS FTP client to the z/OS FTP server.

Simplify FTP transfer of datasets between z/OS systems …

- V2R1 provides new commands to put the complex interactions under-the-covers and simplify the FTP data transfer between z/OS systems
- New FTP command XDSS
  - XDSS is used internally by FTP to get the attributes of the remote MVS data set and send them back to the z/OS FTP client in a 200 reply
- Two new FTP subcommands MVSGet and MVSPut
  - The complex interactions for transferring an MVS data set between z/OS systems are encapsulated in the new FTP subcommands MVSGet and MVSPut
  - With MVSGet or MVSPut, FTP will automatically extract the data set attributes of the source data set and then apply them to the target system before allocating the target data set
  - With MVSGet and MVSPut, you are able to reallocate the existing target data set instead of replacing it. This enhanced the reliability of data set transfers as the source and target data set attributes match
  - With MVSGet and MVSPut, you are able to transfer a PDS or library as a whole. z/OS FTP did not support this before V2R1.

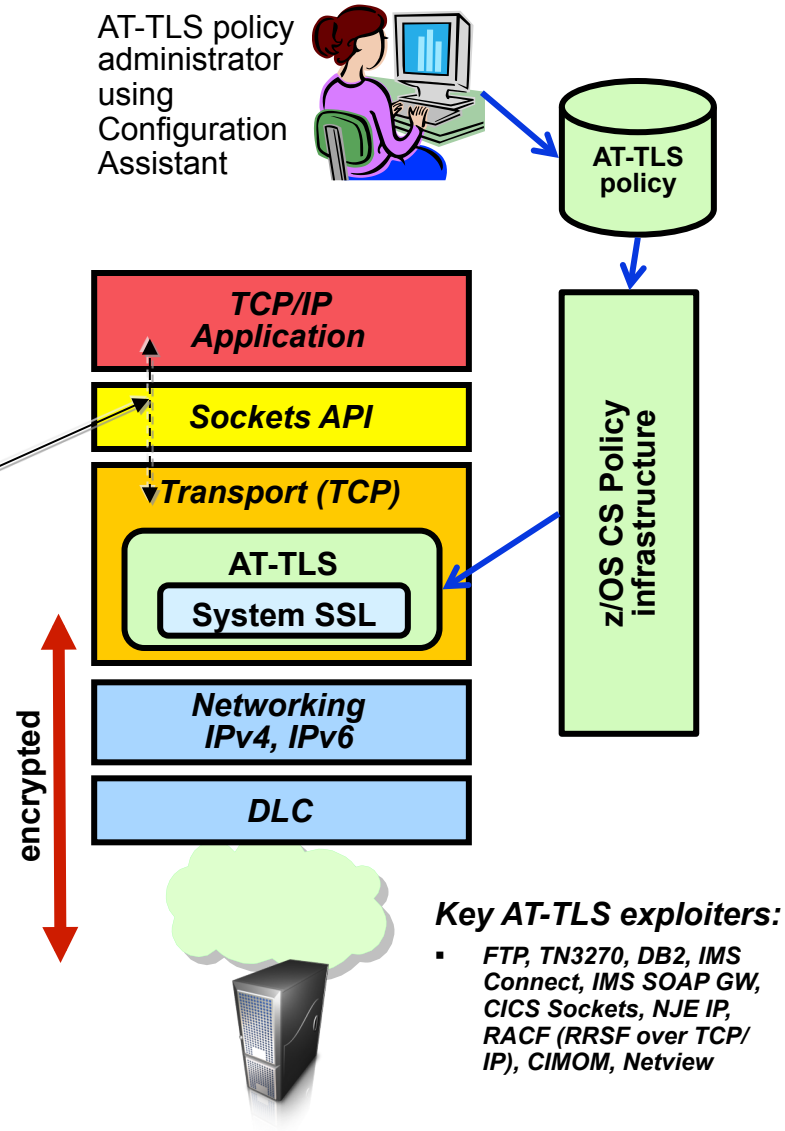
Simplify FTP transfer of datasets between z/OS systems …



Security

z/OS Application Transparent TLS overview

- **Stack-based TLS**
  - TLS process performed in TCP layer (via System SSL) without requiring any application change (transparent)
  - AT-TLS policy specifies which TCP traffic is to be TLS protected based on a variety of criteria
    - Local address, port
    - Remote address, port
    - Connection direction
    - z/OS userid, jobname
    - Time, day, week, month
- **Application transparency**
  - Can be fully transparent to application
  - An optional API allows applications to inspect or control certain aspects of AT-TLS processing – "application-aware" and "application-controlled" AT-TLS, respectively
- **Available to TCP applications**
  - Includes CICS Sockets
  - Supports all programming languages except PASCAL
- **Supports standard configurations**
  - z/OS as a client or as a server
  - Server authentication (server identifies self to client)
  - Client authentication (both ends identify selves to other)
- **Uses System SSL for TLS protocol processing**
  - Remote endpoint sees an RFC-compliant implementation
  - interoperates with other compliant implementations



Key AT-TLS exploiters:

- ***FTP, TN3270, DB2, IMS Connect, IMS SOAP GW, CICS Sockets, NJE IP, RACF (RRSF over TCP/IP), CIMOM, Netview***

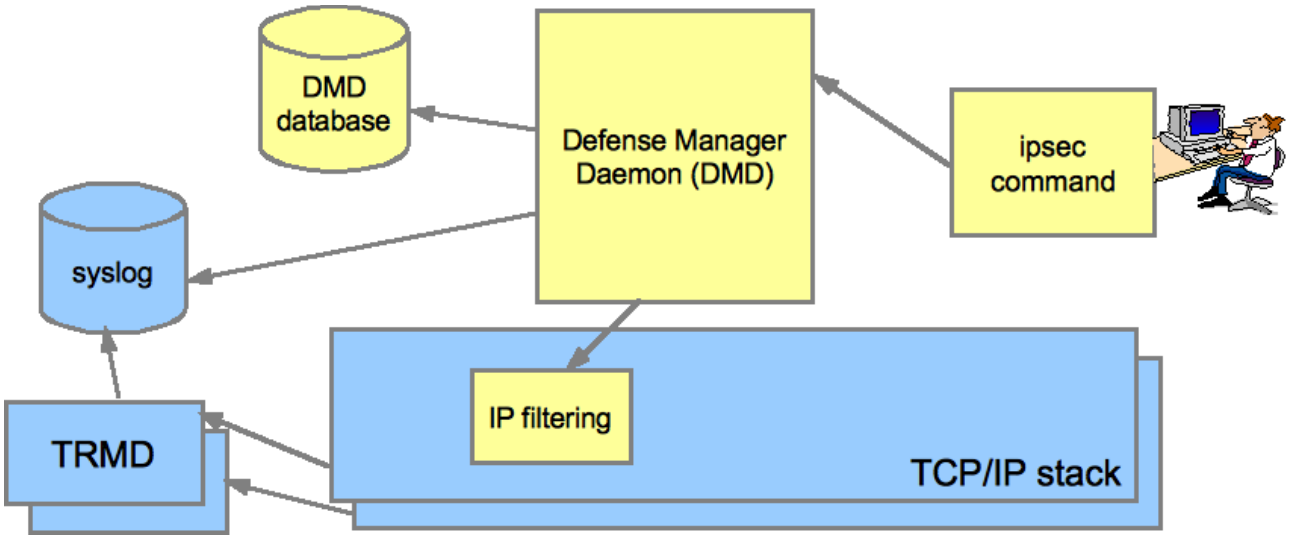
AT-TLS support for TLS v1.2 and related features

- Transport Layer Security (TLS) Renegotiation Extension (RFC 5746):
  - Provides a mechanism to protect peers that permit re-handshakes
  - When supported, it enables both peers to validate that the re-handshake is truly a continuation of the previous handshake
- Support Elliptic Curve Cryptography (ECC)
  - Twenty new ECC cipher suites
    - ECC cipher suites for TLS (RFC 4492)
- TLS Protocol Version 1.2 (RFC 5246):
  - Twenty-one new cipher suites
    - 11 new HMAC-SHA256 cipher suites
    - 10 new AES-GCM cipher suites
  - Requires new System SSL support
- Support for Suite B cipher suites
  - TLS is required
  - ECC 128-bit or 192-bit cipher suites are required

- TLS v1.2 support is also available on V1R13 via APARs OA39422 and PM62905

Limit defensive filter logging

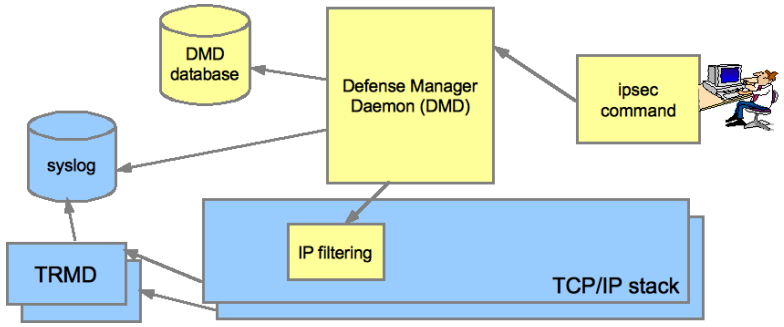
- Defensive filtering provides a mechanism to install temporary defensive filters into a TCP/IP stack to block a specific attack or pattern of attacks



- Two modes of defensive filters:
  - Blocking mode – denies packets that match the defensive filter
  - Simulate mode – for a packet that matches the defensive filter, a log record is written to syslogd to indicate that the packet would have been denied if this were a blocking filter, but filtering of packet continues
    - Often used initially by customers to gauge the effect of implementing defensive filtering before enabling blocking mode
    - "EZD1722I Packet would have been denied by defensive filter" logged for each packet that matches this filter

Limit defensive filter logging…

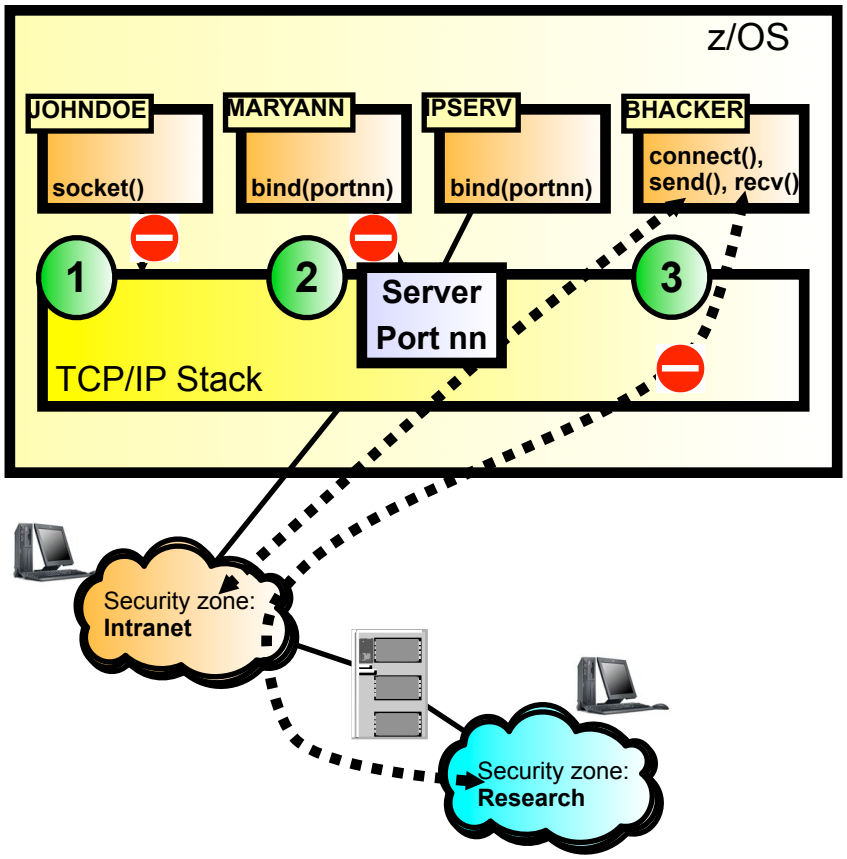
- Per-packet logging can flood syslogd during an attack
  - In blocking mode, logging can be turned on or off
  - In simulate mode, logging is always on, with a log record generated for each packet that matches the filter
  - No mechanism to limit syslogd output



- V2R1 adds a user-specified limit for the number of defensive filter messages written to syslogd over a 5-minute interval
  - Limit is per defensive filter (not across all defensive filters)
  - Limits number of EZD1721I and EZD1722I defensive filter messages
  - Value from 0 – 9999
    - 0 indicates “no limit” - message written to syslogd for each packet that matches the defensive filter
    - 1 – 9999 – indicates limit to be applied for defensive filter
  - Number of suppressed messages reported per defensive filter every five minutes (and when a defensive filter expires)

Improved auditing of NetAccess rules

- NetAccess provides the ability to control z/OS user access to certain security zones (networks, subnetworks, and hosts)
- Via NETACCESS statement In TCP/IP
- Access to security zone allowed if user permitted to SAF resource (SERVAUTH class: NETACCESS)
- Results from SAF calls to check if a user can access a security zone are cached
  - Only first access check for a user to a security zone results in a SAF call
    - No SAF call for subsequent access checks for user for same IP address
    - No SAF call for subsequent access checks for user to different IP addresses in same zone



- NetAccess provides a log string on all calls made to SAF to check a user's access to a SAF resource profile, and SAF includes that log string in its audit records (for example, RACF SMF record)

Improved auditing of NetAccess rules …

- Customers have asked for more control of caching to provide more audit details
  - SAF audit records are only written when a SAF call is made
  - Security auditors need audit records that are inhibited by caching
- In V2R1, a new parameter will be added to NetAccess to control caching:
- **CACHEALL (default)**
  - Results from all NetAccess SAF calls are cached, both when the user is permitted access and when the user is denied access to the zone
    - Audit record written for only the first access check made for a user to each security zone
- **CACHEPERMIT**
  - Results from NetAccess SAF checks are cached when the user is permitted access, but not when the user is denied access to the zone
    - Audit record written for only the first access check made for a user to each security zone to which user is permitted
    - Audit record written for all access checks made for a user to each security zone to which user is denied
- **CACHESAME**
  - Same as CACHEPERMIT, but the cache is used by a socket only as long as the user and the IP address being accessed remain unchanged
    - Audit records written as for CACHEPERMIT, plus audit record written for next access check after socket user or remote IP address being used by the socket changes

Improved auditing of NetAccess rules …

- SAF audit record contains user ID and resource profile name, but not the IP address that is being accessed
  - The security zone associated with the resource profile can contain multiple IP addresses
  - No record of which IP addresses within the security zones are being accessed
- Security auditors need the IP address information
  - Especially important when access is denied
- In V2R1, all SAF calls made for NetAccess will include the IP address that triggered the call

ICMP outbound flood prevention

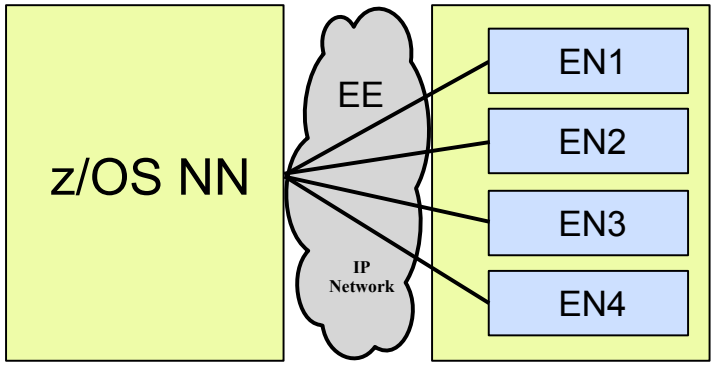
- Communications Server paces TCP and EE outbound traffic
  - One benefit of this is that it prevents excessively long queue buildups that could tie up large amounts of CSM storage.
  - This type of pacing doesn't apply to ICMP, RAW, or non-EE UDP traffic.
- Communications Server also limits the number of inbound packets on a single QDIO interface at a given time.
- V2R1 will add similar protection for outbound ICMP, RAW, and UDP traffic by dropping outbound QDIO packets when approaching CSM constrained or critical conditions, and with the outbound QDIO queues in a congested state.

Improved FIPS 140 diagnostics

- z/OS Communications Server components that offer a FIPS-140 operational mode:
  - IKED, NSSD, and AT-TLS (configured in FIPS-140 mode)
- In FIPS-140 mode, components must call z/OS cryptographic modules for all cryptographic operations
  - Must call ICSF and System SSL (configured in FIPS-140 mode)
  - Internal routines disabled
  - Direct hardware calls disabled
- Starting in V2R1, System SSL in FIPS-140 mode must call ICSF
  - Change made to satisfy FIPS-140 requirements, eliminate redundancy, and improve efficiency
  - Applies to FIPS-140 mode only
- As a result, components calling System SSL in FIPS-140 mode now require ICSF
  - IKED and NSSD will fail to initialize if ICSF is not active, and AT-TLS policy groups will be installed but inactive if ICSF is not active
  - In V2R1, z/OS CS adds new messages to indicate ICSF status during IKED and NSSD initialization, and during the installation of AT-TLS policy groups

Enterprise Extender / SNA / Miscellaneous

EE/SNA



- V1R11 introduced Progressive Mode ARB flow control to address the tendency of HPR's Responsive Mode ARB to be very sensitive to minor variations in packet round-trip time or unpredictability in response time from the RTP partner node. If the partner suddenly becomes CPU constrained, even for a short period, throughput and response time can be degraded. For example:
  - Partner node has a shortage of CPU availability, memory, or network bandwidth
  - Partners in a virtual server environment on a single hardware platform cannot guarantee consistent response time
- Progressive mode ARB implemented several small changes to the flow control rules to improve responsiveness in a CPU-constrained environment
  - Both partners must agree to use progressive mode ARB
  - Limited to single-hop pipes over an EE connection (including two-virtual-hop connection network paths)
  - HPREEARB = PROGRESS can be specified on an EE PU in a switched major node, on a connection network GROUP in the EE XCA major node, or on the EE model PU in a model major node
- **V2R1 adds the HPREEARB parameter to the GROUP statement for pre-defined EE connections**
  - Provide the ability to specify the HPREEARB parameter on the switched major node GROUP statement for pre-defined EE connections

EE/SNA …

- IPv6 support for EE's IPADDR operand
  - Prior to V2R1, EE's IPADDR parameter is IPv4-only, with the assumption that IPv6 addresses will be provided via HOSTNAME resolution. V2R1 will add IPv6 support for the IPADDR start option, XCA, and switched PU parameters

```
VTAM Start Option List
…
IPADDR=2000::67:1:2
…
```

- PSRETRY Enhancements
  - Provide a PSRETRY option to look for a new route after a local topological change. This is sort of the inverse of link-inop-driven path switch, and should provide better function and performance than the current timer-only mechanism.
- Display EE command enhancements
  - A new CPNAME filter will be added to DISPLAY EE to request all active EE connections to a given partner CP name
- Enhanced TRS traces
  - Provides additional internal traces to be used by z/OS CS service in diagnosing APPN routing problems
- The TSO DNET command will be removed from z/OS CS

Miscellaneous items

- The Communications Server system resolver will now start even if errors are detected with statements in the resolver setup file.
  - The resolver will also start if the resolver setup file does not exist or cannot be accessed by the resolver.
  - This allows your TCP/IP stacks and other applications dependent on resolver processing to continue their initialization despite any resolver setup file errors.
- OMPROUTE change to turn on HELLO_HI by default
  - Enable processing of OSPF hello packets at a high priority by default. Adds a new global configuration parameter Enable_Hello_Hi with YES and NO values to the GLOBAL_OPTIONS statement in the OMPROUTE configuration file.
- Netstat socket creation time
  - The NETSTAT ALL output is updated to include a start date and time for a socket connection
- TSO/VTAM will provide the ability to translate Extended English characters for the TPUT EDIT macro instruction
- Support is added for cross-memory TPUT messages from TSO/VTAM to z/OSMF ISPF address spaces.

Miscellaneous items …

- Trace event exit regardless of CFS trace option
  - The Coupling Facility Services (CFS) component will always trace connection related activities and other important information in the mini-trace table for the Coupling Facility structures: ISTGENERIC, EZBDVIPA and EZBEPORT
- Additional diagnostic data associated with OMPROUTE heartbeats will be captured to aid in the analysis of OMPROUTE responsiveness problems
- OMPROUTE messages reporting failures in adding, changing, or deleting a route will contain more information on the failing route
- The EZZ4310I messages that report device failures will be supplemented by additional messages that further describe the failure
- The FTP client has been enhanced with trace messages to assist with the diagnosis of problems opening files. In addition to the already existing EZA2564W messages documenting a failure, these trace messages will provide additional information about the root cause of the failure.

Miscellaneous items …

- In V2R1, z/OS Communications Server provides a mechanism that allows an application to issue a synchronous or asynchronous receive socket API call that completes only when a TCP connection is terminated.
- z/OS CS will provide an API for determining the SyslogD configuration file location and name.
- Enhanced IDS IP fragment detection
  - V2R1 will change the fragmentation attack probe to no longer consider fragment length as a criteria. Checks will be based purely on whether overlays occur and whether they change the packet content.

For more information

URL	Content
http://www.twitter.com/IBM_Commserver	IBM z/OS Communications Server Twitter Feed
http://www.facebook.com/IBMCommserver	IBM z/OS Communications Server Facebook Page
https://www.ibm.com/developerworks/mydeveloperworks/blogs/IBMCommserver/?lang=en	IBM z/OS Communications Server Blog
http://www.ibm.com/systems/z/	IBM System z in general
http://www.ibm.com/systems/z/hardware/networking/	IBM Mainframe System z networking
http://www.ibm.com/software/network/commserver/	IBM Software Communications Server products
http://www.ibm.com/software/network/commserver/zos/	IBM z/OS Communications Server
http://www.redbooks.ibm.com	ITSO Redbooks
http://www.ibm.com/software/network/commserver/zos/support/	IBM z/OS Communications Server technical Support – including TechNotes from service
http://www.ibm.com/support/techdocs/atsmastr.nsf/Web/TechDocs	Technical support documentation from Washington Systems Center (techdocs, flashes, presentations, white papers, etc.)
http://www.rfc-editor.org/rfcsearch.html	Request For Comments (RFC)
http://www.ibm.com/systems/z/os/zos/bkserv/	IBM z/OS Internet library – PDF files of all z/OS manuals including Communications Server
http://www.ibm.com/developerworks/rfe/?PROD_ID=498	RFE Community for z/OS Communications Server
https://www.ibm.com/developerworks/rfe/execute?use_case=tutorials	RFE Community Tutorials

For pleasant reading ….

Please complete your session evaluation

- z/OS Communications Server Technical Update, Part 2
- Session # 12844
- QR Code:

IBM Comm Server

Find us on Facebook at
http://www.facebook.com/IBMCommserver

Follow us on Twitter at
http://www.twitter.com/IBM_Commserver

Read the z/OS Communications Server blog at
http://tinyurl.com/zoscsblog

Visit the z/OS CS YouTube channel at
http://www.youtube.com/user/zOSCommServer