z/OS V2R2 Communications Server Technical Update, Parts 1 & 2

SHARE 2015 Winter Technical Conference

Sessions 16739 and 16740

Gus Kassimis
kassimis@us.ibm.com
Sam Reynolds
samr@us.ibm.com

#SHAREorg

SHARE is an independent volunteer-run information technology association that provides education, professional networking and industry influence.

SHARE in Seattle 2015

Trademarks

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

IBM*
IBM Logo*

* Registered trademarks of IBM Corporation

The following are trademarks or registered trademarks of other companies.

Adobe, the Adobe logo, PostScript, and the PostScript logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.
IT Infrastructure Library is a registered trademark of the Central Computer and Telecommunications Agency which is now part of the Office of Government Commerce.
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.
Linux is a registered trademark of Linus Torvalds 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.
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.
UNIX is a registered trademark of The Open Group in the United States and other countries.
Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.
Cell Broadband Engine is a trademark of Sony Computer Entertainment, Inc. in the United States, other countries, or both and is used under license therefrom.
Linear Tape-Open, LTO, the LTO Logo, Ultrium, and the Ultrium logo are trademarks of HP, IBM Corp. and Quantum in the U.S. and other countries.

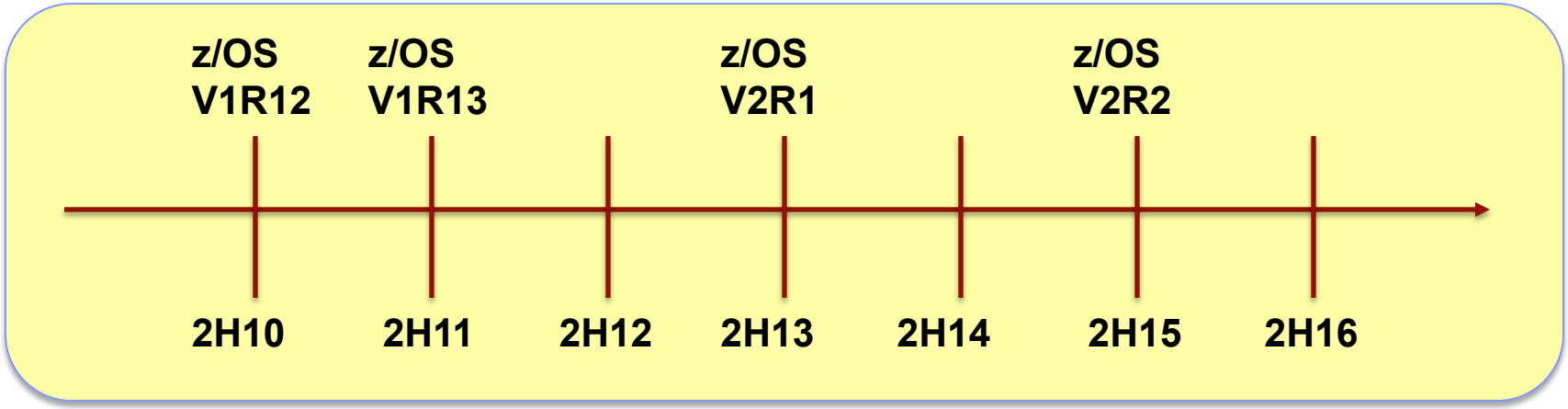
* All other products may be trademarks or registered trademarks of their respective companies.

Notes:
Performance is in Internal Throughput Rate (ITR) ratio based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput that any user will experience will vary depending upon considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve throughput improvements equivalent to the performance ratios stated here.
IBM hardware products are manufactured from new parts, or new and serviceable used parts. Regardless, our warranty terms apply.
All customer examples cited or described in this presentation are presented as illustrations of the manner in which some customers have used IBM products and the results they may have achieved. Actual environmental costs and performance characteristics will vary depending on individual customer configurations and conditions.
This publication was produced in the United States. IBM may not offer the products, services or features discussed in this document in other countries, and the information may be subject to change without notice. Consult your local IBM business contact for information on the product or services available in your area.
All statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only.
Information about non-IBM products is obtained from the manufacturers of those products or their published announcements. IBM has not tested those products and cannot confirm the performance, compatibility, or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.
Prices subject to change without notice. Contact your IBM representative or Business Partner for the most current pricing in your geography.

z/OS V2R2 Communications Server disclaimer

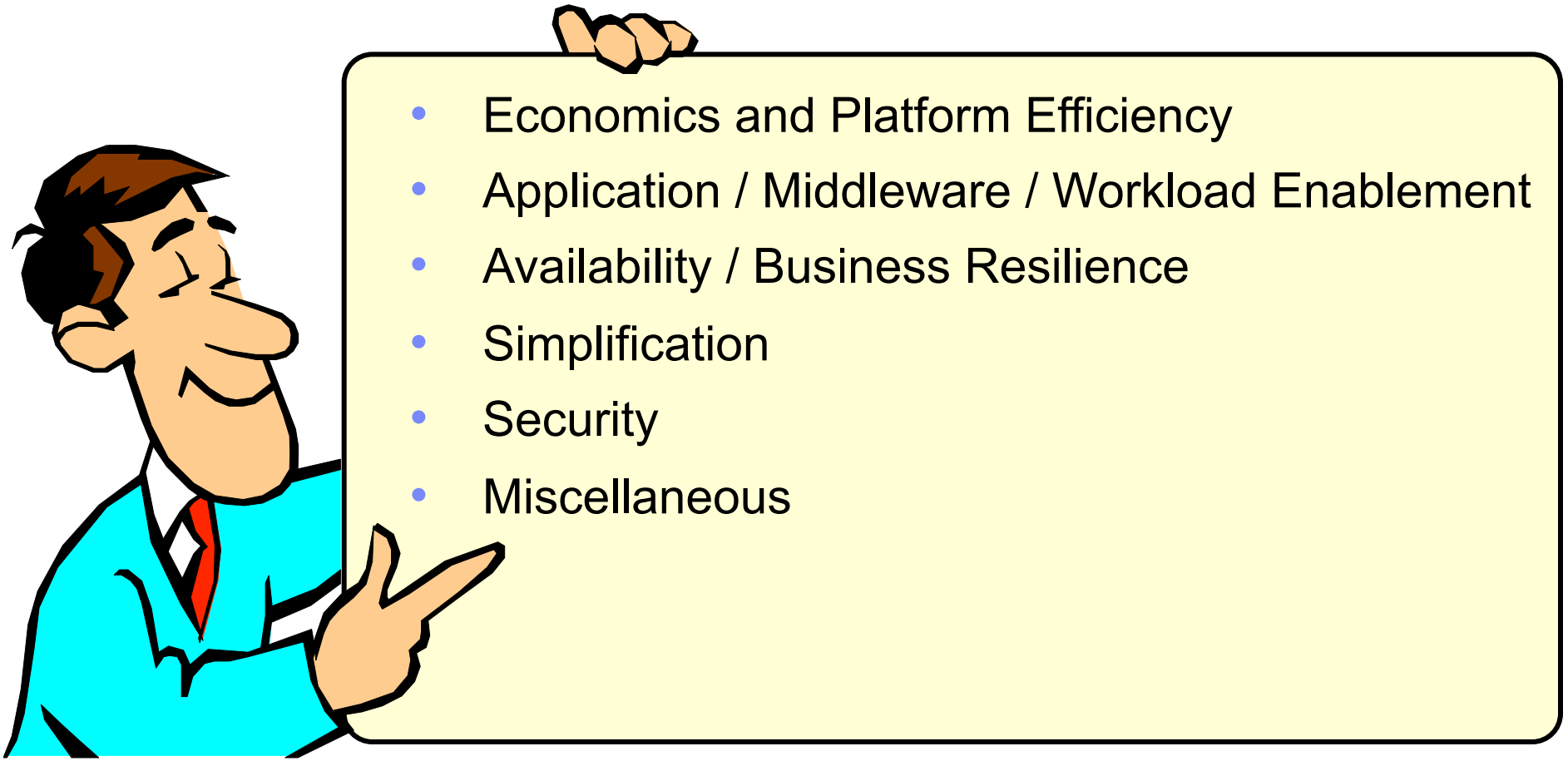
- Plans for the z/OS Communications Server are subject to change prior to general availability.
- Information provided in this presentation may not reflect what is actually shipped by z/OS Communications Server.
- This presentation includes an early overview of selected future z/OS Communications Server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

Plans may change before GA of z/OS V2R2



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

Agenda



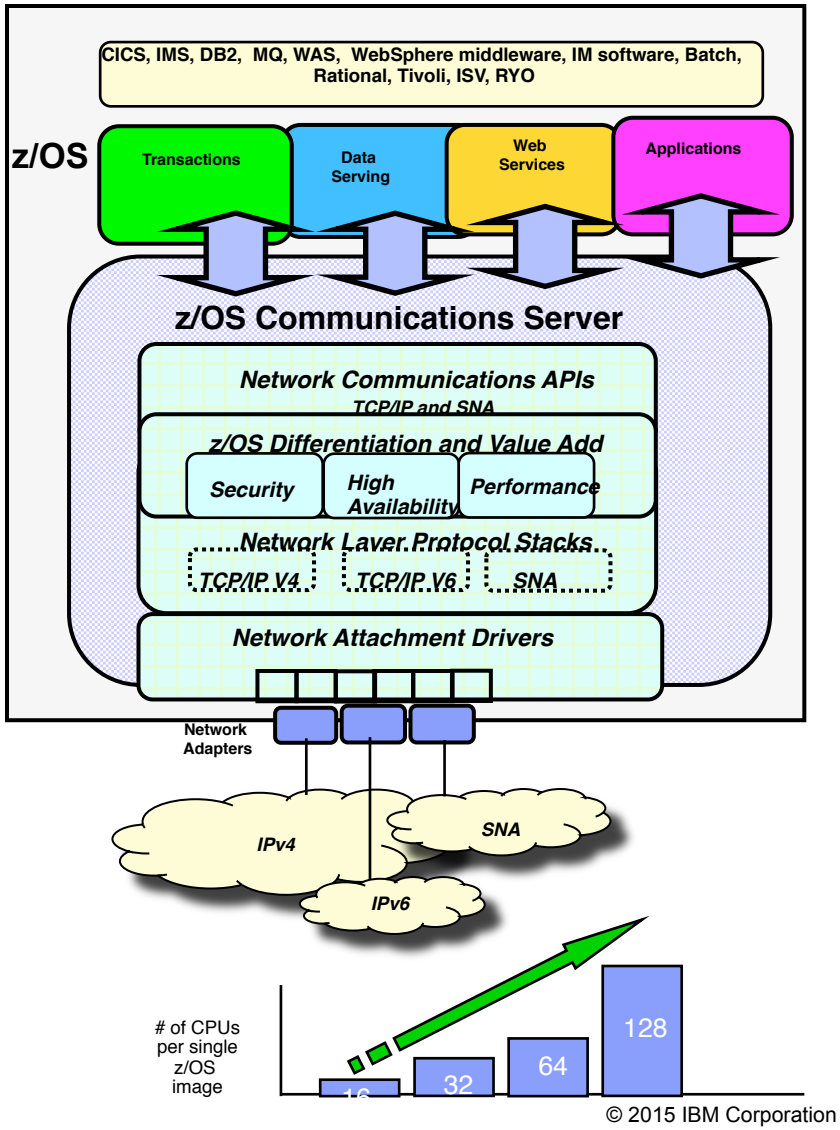
- Economics and Platform Efficiency
- Application / Middleware / Workload Enablement
- Availability / Business Resilience
- Simplification
- Security
- Miscellaneous

Economics / Platform Efficiency

64-Bit enablement of the TCP/IP stack

- TCP/IP has supported 64-bit applications since 64-bit support was introduced on the platform
  - But the mainline path has been 31-bit with extensive use of AR mode

- As systems become more powerful, customers have increased the workloads on the systems which in turn increases the storage demands placed on the systems.
- The storage in 31-bit addressing mode (below the bar) has been of special concern. Over the past several releases work has started to move storage that used to be obtained below the bar to 64-bit addressing mode (above the bar).
- Some of these changes, such as V1R13's move of the CTRACE and VIT above the bar, were visible to customers, while others were just changes in internal "plumbing".
- The next step is a large one: To move most of the remaining storage above the bar without incurring an unacceptable overhead in switching between AMODE(31) and AMODE(64) requires the complete 64-bit enablement of the TCP/IP stack and strategic device drivers (DLCs).



64-bit Enablement of TCP/IP stack (cont)

- Example of memory savings:
  - The table below shows the results of a TN3270 performance run with 128,000 concurrent connections. It shows the change in storage usage in the TCP/IP address space and Common Storage between V2R1 (31 bit TCP/IP stack) and V2R2 (64 bit TCP/IP stack).

Storage Type	V2R1 (KB)	V2R2 (KB)	% change from V2R1
TCP/IP ECSA	9,188.00	6,593.00	-28%
TCP/IP Private	275,338.00	43,332.00	-84%

  - Storage areas and control blocks moved from ECSA and TCP/IP private to HVCOMMON and above the bar TCP/IP private
    - Removing workload growth constraints from below the bar Private and Common storage

Statement of Direction: End of support for TCP/IP legacy device drivers (Issued Feb. 24, 2014)

z/OS V2.1 is planned to be the last z/OS release to provide software support for several TCP/IP device drivers. IBM recommends that customers using any of these devices migrate to more recent device types, such as OSA Express QDIO and Hipersockets. The TCP/IP device drivers planned to be removed are: Asynchronous Transfer Mode (ATM), Common Link Access To Workstation (CLAW), HYPERChannel, Channel Data Link Control (CDLC), SNALINK (both LU0 and LU6.2), and X.25.

Note: Support for SNA device drivers is not affected.

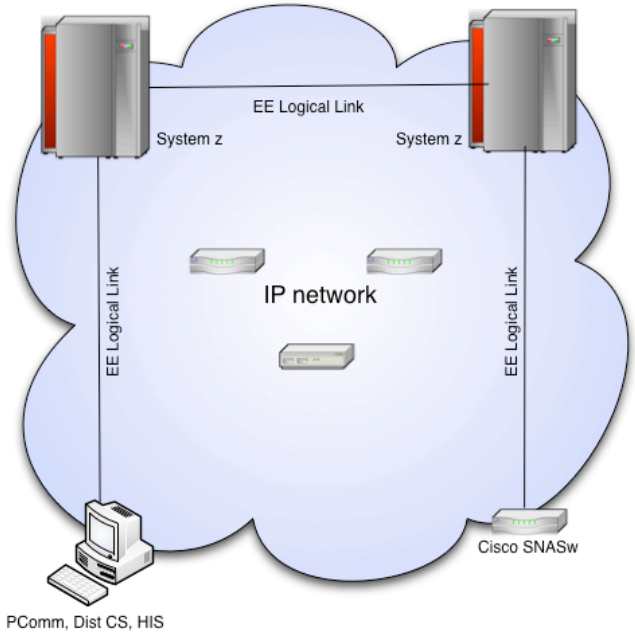
Summary of z/OS CS TCP/IP device drivers

Device driver type	Planned for withdrawal
OSA Express QDIO (OSD, OSX)	no
Hipersockets (iQDIO)	no
Legacy OSA (LCS – OSE)	no
CTC P2P	no
MPC P2P (Multi-path Channel Point-to-Point)	no
XCF (Dynamic XCF)	no
MPC SAMEHOST	no
SNALINK (LU0 and LU6.2)	yes
X.25 SAMEHOST	yes
CLAW (e.g. Cisco CIPs)	yes
Hyperchannel	yes
CDLC (3745/3746 connections)	yes
ATM	yes

Removed support for legacy devices: Migration

- Customers who are still using these device types will need to migrate to one of the supported device types.
- Since many of these customers are already running on unsupported hardware types, this is a migration action that most should be considering already.
- A migration health check is available that will determine if any of these legacy device types are defined in the TCP/IP Profile.
  - This health check is available for V1R13 and V2R1 via APARs PI12977/ OA44669 (V1R13) and PI12981/OA44671 (V2R1).

Enhanced Enterprise Extender scalability



- Enterprise Extender processing uses the UDP protocol for communication between local and remote EE nodes. A local node is defined by at least one local static VIPA and 5 UCB control blocks.
- Each UCB is bound to the static VIPA and one of 5 UDP ports. The ports map to the 4 SNA routing priorities for data traffic, plus one port for LLC commands.
- The z/OS CS EE implementation has scaled very well in most customer implementations. However, we have seen EE deployments with well over 10,000 EE connections into a single z/OS image, and there is a need to improve scalability in those configurations, especially in failover scenarios.
- In V2R2, the internal control block structure in EE's dedicated UDP component is being changed to facilitate significantly-improved scalability in customer configurations with extremely large numbers of EE connections.

Enhanced IKED scalability: Problem

- At least one customer has observed that when multiple thousands of IPsec endpoints attempt to simultaneously establish security associations (SAs) with a single z/OS IKED, the IKED can become bogged down, resulting in an inordinate amount of time to successfully establish all of the SAs.
- Certain factors exacerbate the issue:
  - Heavy CPU utilization across the CEC
    - Even though IKED and its related processes are given high (SYSSTC) priority, IKED's CPU can still be stolen out from under it
  - Peers that retransmit often - Windows IKEv1 clients are the prime example:
    - Expectation that a phase 2 negotiation will always follow a phase 1 main mode when Windows is the responder
    - (Up through Windows 7) does not support aggressive mode or IKEv2
    - Inability to configure the retransmit intervals
    - Failure to reset the refresh timer after a peer-initiated refresh when Windows is the original initiator – result is an extra refresh exchange after completing peer-initiated refresh
    - Automatic deletion of SAs after 5 minutes of inactivity (Microsoft has issued a patch to relieve this one)

Enhanced IKED scalability: Solution

- Current IKED design relies heavily on a single thread to perform the bulk of the work
- Over the years, various measures have been taken to maximize the efficiency and throughput of the current design
- In V2R2, IKED will be modified to handle heavy bursts of negotiations from very large numbers (multiple thousands) of IKE peers
  - A new thread pool is added to parallelize handling of IKE messages from different peers
  - Logic is added to minimize the amount of effort IKED spends processing retransmitted messages from peers
- Externals impact is minimal:
  - Only new configuration value is one new trace level
  - No new command parameters or outputs
  - Syslogd output will appear much more interleaved
- Transparent to the vast majority of current IKED users.
  - Improvement will be most noticeable to users with very large numbers (multiple thousands) of IKE peers
  - Such users may need to tune memory, message queue or UDP queue limits.

Enhanced IKED scalability: Solution (cont)

- Preliminary z/OS V2R2 performance results show significant performance improvements in establishing SAs when a large number of concurrent client requests arrive in a small interval of time
  - IKE V1:
    - *Up to 6.8X improvement in throughput (rate of SA activations) and up to 75% reduction in CPU cost **
  - IKE V2:
    - *Up to 3.8X improvement in throughput (rate of SA activations) and up to 57% reduction in CPU cost **

* *Note:* The performance measurements were collected in IBM internal tests using a dedicated system environment. 4,200 clients simulated using 4 Linux for System z images running under zVM. IKE v1 benchmarks performed with PSK. IKE v2 benchmarks performed with RSA. The results obtained in other configurations or operating system environments may vary significantly depending upon environments used. Therefore, no assurance can be given, and there is no guarantee that an individual user will achieve performance or throughput improvements equivalent to the results stated here.

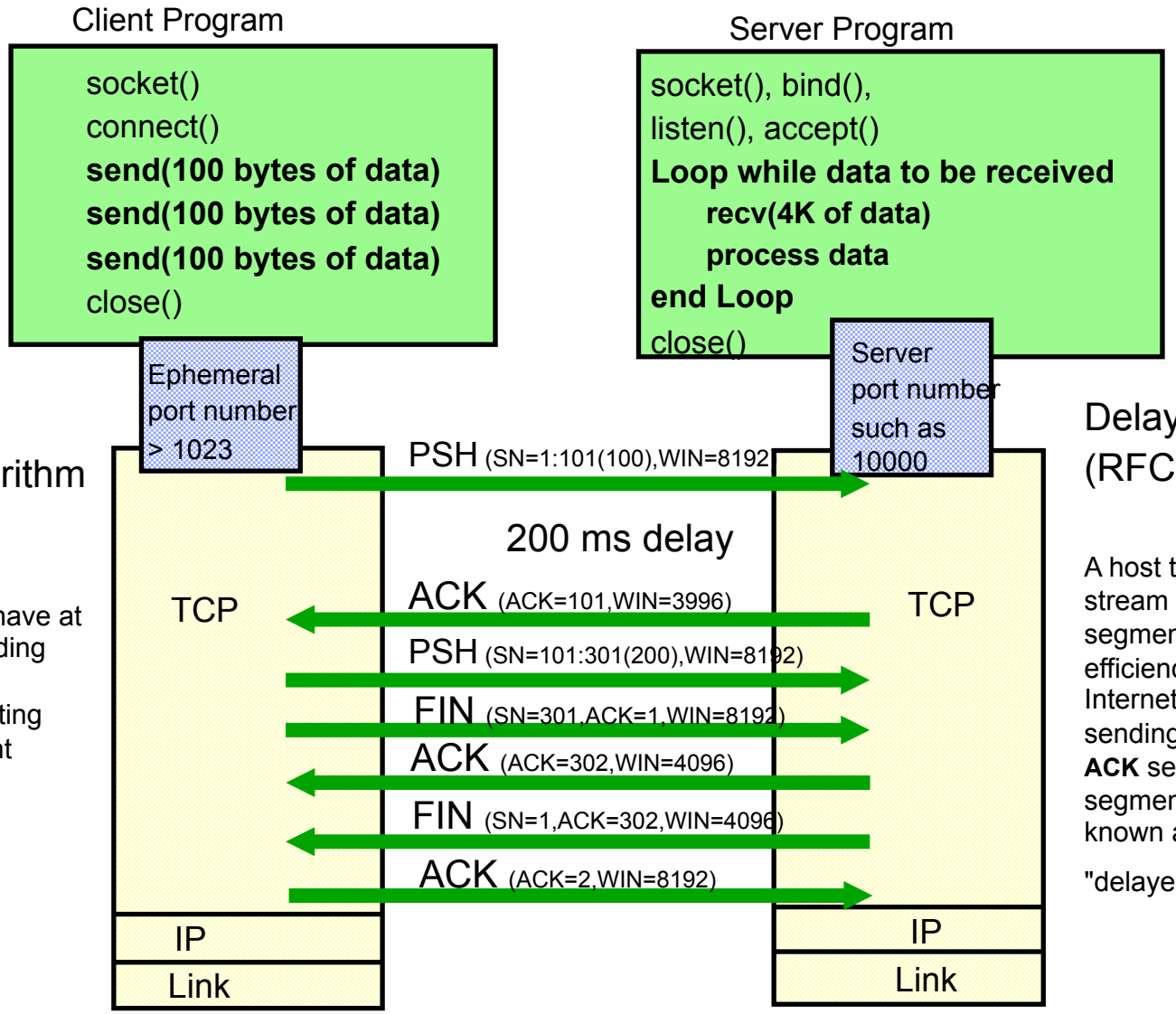
Increase single-stack DVIPA limit

- The current limit on the total number of DVIPAs on a system is 1024. This includes:
  - All DVIPA's defined with VIPADEFINE
  - All DVIPA's defined with VIPABACKUP
  - All deactivated DVIPAs
  - All application instance DVIPAs (defined with VIPARANGE)
    - Popular with many users (used to virtualize network access to single instance of application)
- In V2R2, we will lift the current limit of 1024 DVIPAs to 4096 for application instance DVIPAs.
  - DVIPAs defined with VIPADEFINE and VIPABACKUP and target of a VIPADISTRIBUTE will still be limited to 1024.
  - The current limit of 1024 IPv4 VIPARANGE statements and 1024 IPv6 VIPARANGE statements will also be lifted to 4096 for each type of VIPARANGE statement.

DelayACK processing

- DelayACK processing is the default behavior for the TCP protocol
  - By default, the TCP protocol will delay sending an ACK packet to acknowledge an incoming TCP segment to conserve bandwidth and CPU cycles
    - And it works well most of the time!
    - Significant CPU savings for piggybacking ACK onto response (request/response workloads)
    - And for streaming workloads
      - TCP will ACK every second packet anyway
- Where it's a problem
  - Sender is waiting for ACK before allowing next segment to be sent
    - Nagle's algorithm compounds this problems
      - z/OS has a relaxed Nagle's algorithm that can help when we are the sender only
  - Any time the DELAYACK timer goes off to push an ACK it is probably not a good thing
    - A 200 millisecond delay is injected
      - If the remote peer is waiting for this ACK before pushing out the next segment
    - If these occur multiple times for a the life of a connection this can lead to significant overall delays

TCP Delayack Processing and Nagle's Algorithm "Catch-22"



Nagle's Algorithm (RFC896)

You should only have at most one oustanding small segment ("tinygram") awaiting acknowledgement

Goal: Minimize congestion

Delayed Acks (RFC1122)

A host that is receiving a stream of TCP data segments can increase efficiency in both the Internet and the hosts by sending fewer than one **ACK** segment per data segment received; this is known as a

"delayed ACK"

TCP/IP autonomic tuning enhancements: DelayACK processing

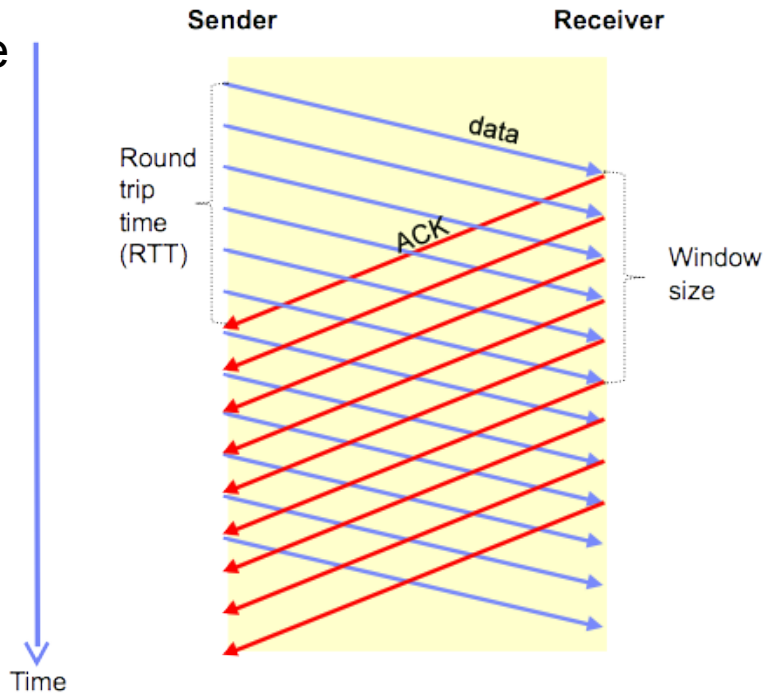
- TCP/IP autonomic tuning of performance sensitive areas
  - New AUTODELAYACK option (in addition to existing DELAYACK and NODELAYACK options)
    - Allows TCP/IP to monitor the impact of delayed ACKs on a
      - TCP connection basis
      - For all connections to a TCP server
    - TCP/IP can then automatically disable/enable Delayed ACKs on a TCP connection basis or a TCP Server basis in order to maximize performance (i.e. response time/throughput) based on the traffic pattern characteristics of the connection
      - Do not delay ACK if it repeatedly prevents the partner from sending more data
      - Do not keep sending ACKs to every packet if the sender is sending its next packet anyway
    - Netstat All report enhanced to display current status for AUTODELAYACK for a TCP connection
  - Does not override an explicit DELAYACK or NODELAYACK specification on PORT|PORTRANGE, ROUTE (BEGINROUTES), or Policy Based Routing (PBR) policy

Determining current Delayack setting for a connection

```
NETSTAT ALL
MVS TCP/IP NETSTAT CS V2R2      TCPIP Name: TCPCS 22:24:30
Client Name: FTPD1                   Client Id: 000000F9
Local Socket: 9.42.104.43..21        Foreign Socket: 9.42.103.165..1035
  BytesIn:             0000000035    BytesOut:             0000000265
  SegmentsIn:          0000000017    SegmentsOut:          0000000014
  StartDate:           01/09/2012    StartTime:            22:04:11
  Last Touched:        22:04:18      State:                Establsh
  RcvNxt:              0214444666    SndNxt:               0216505563
  ClientRcvNxt:        0214443596    ClientSndNxt:         0216504670
  InitRcvSeqNum:       0214443560    InitSndSeqNum:        0216504404
  CongestionWindow:    0000007336    SlowStartThreshold:   0000065535
  IncomingWindowNum:   0214477396    OutgoingWindowNum:    0216538247
  SndWl1:              0214444666    SndWl2:               0216505563
  SndWnd:              0000032684    MaxSndWnd:            0000032768
  SndUna:              0216505563    rtt_seq:              0216505479
  MaximumSegmentSize:  0000000524    DSField:              00
  Round-trip information:
    Smooth trip time:  102.000       SmoothTripVariance:   286.000
  ReXmt:               0000000000    ReXmtCount:           0000000000
  DupACKs:             0000000000    RcvWnd:               0000032730
  SockOpt:             85            TcpTimer:             00
  TcpSig:              84            TcpSel:               60
  TcpDet:              E0            TcpPol:               00
  TcpPrf:              C0            TcpPrf2:              E0
  TcpPrf3:             00
  DelayAck:            AutoYes
  QOSPolicy:           No
...
```

TCP/IP autonomic tuning enhancements: Dynamic Right Sizing Autonomics

- **Provide automatic re-enablement and tuning of Dynamic Rightsizing (DRS) function**
- DRS can have a significant impact on performance of streaming connections over long latency links (i.e. distance)
  - By increasing the receive buffer sizes for an application up to 2MB (“keep the pipe full”)
  - Function is autonomically enabled today but has defensive measures that disable it when maximum benefit is not being obtained
    - For example, when the application is not receiving data fast enough
      - This could be the result of a temporary system constraint (i.e. High CPU utilization, System dumps being taken, etc.)
    - Once disabled today, it cannot be re-enabled again for the life of a connection

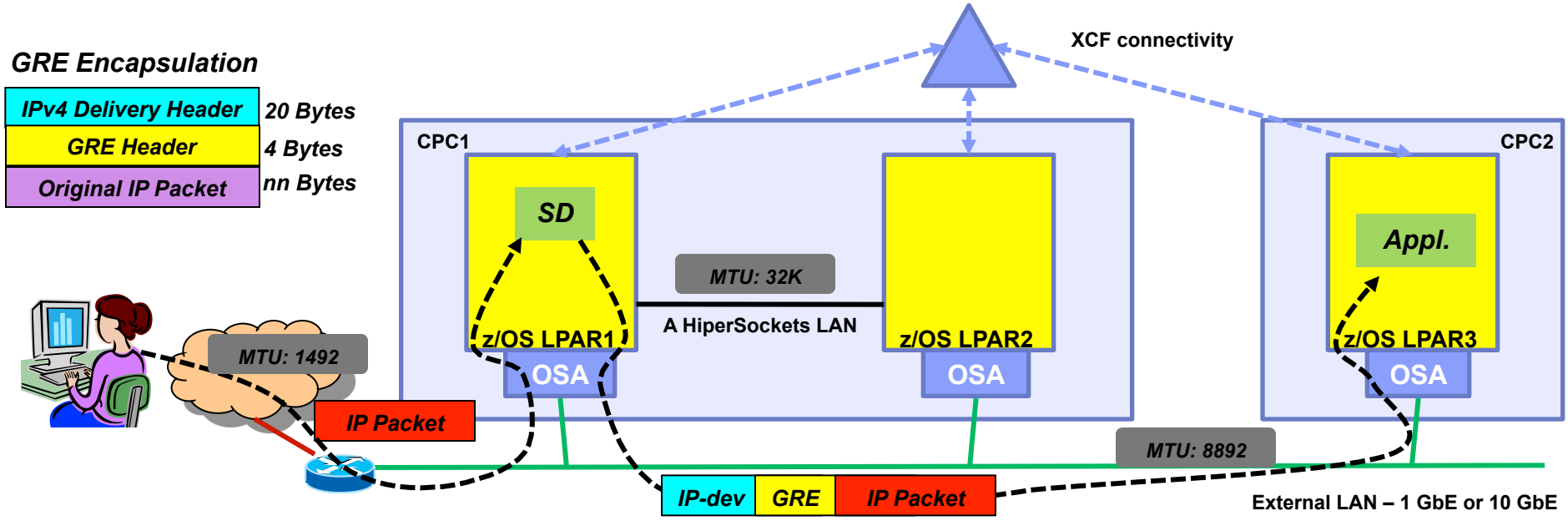


TCP/IP autonomic tuning enhancements: Dynamic Right Sizing Autonomics

- V2R2 will enhance the autonomic logic for disabling/enabling DRS:
  - Allow DRS usage to be restarted on a connection
    - DRS detection can be re-initiated after a certain number of packets are processed
  - When CSM storage is not constrained:
    - Continue using DRS on a connection even if the application falls behind
  - When CSM storage is constrained:
    - If application falls behind, stop DRS on the connection temporarily
    - Do not activate DRS for connection, either initially or during “restart conditions”

VIPAROUTE and MTU size considerations

- When VIPAROUTE is used, the distributing stack adds a GRE header to the original IP packet before forwarding to the target stack
- Two ways to avoid fragmentation between distributing and target stacks:
  - Have clients use path MTU discovery
    - z/OS will factor in the GRE header size (24 bytes) when responding with next-hop MTU size
    - Not always possible to control distributed nodes' settings from the data center
  - Use jumbo-frames on the data center network
    - The access network will typically be limited to Ethernet MTU size (1492 bytes), while the data center network will be able to use jumbo frame MTU size (8892 bytes)
    - Adding the GRE header will not cause fragmentation in this scenario

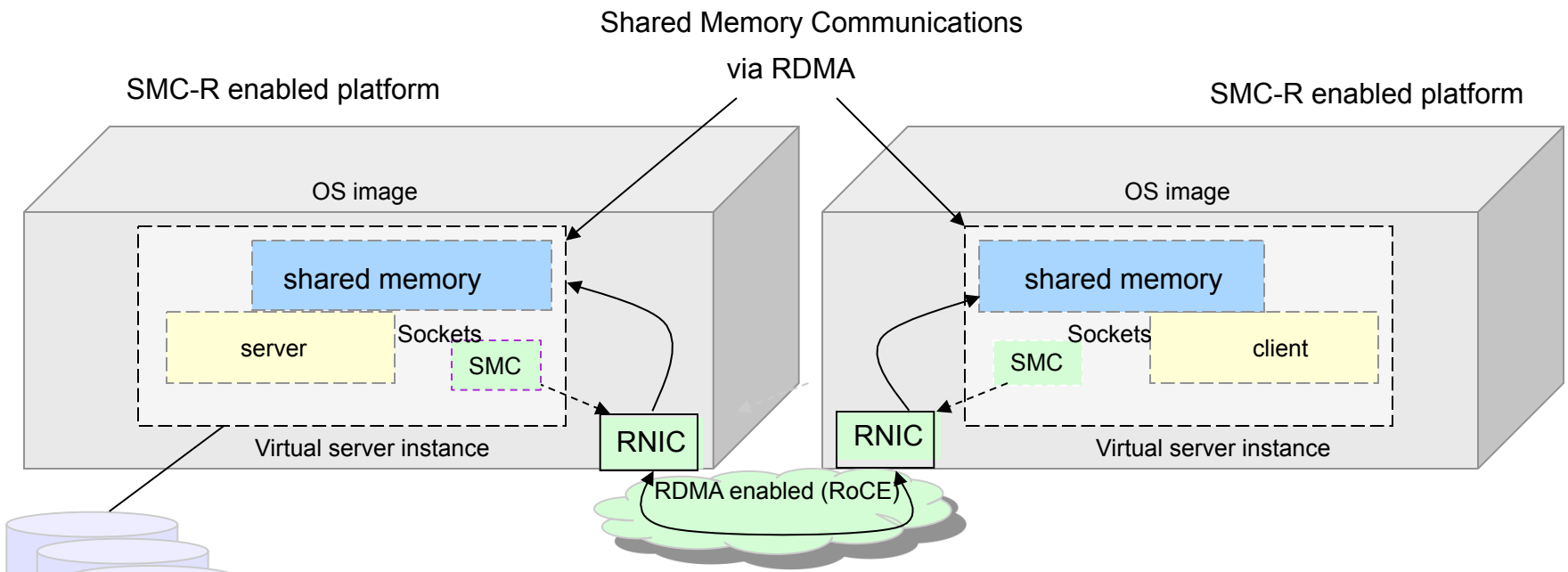


VIPAROUTE fragmentation avoidance

- VIPAROUTE has been used extensively by many users to offload sysplex distributor forwarded traffic from XCF links
  - When used in combination with QDIO Accelerator for SD can result in dramatically reduced overhead for SD forwarding
- Fragmentation is still a concern for several customers
  - Resulting from the extra 24 bytes that are needed for the GRE header
  - Path MTU Discovery helps but doesn't solve the issue in some environments (where ICMP messages cannot flow across FWs)
- V2R2 introduces a new autonomic option that will automatically reduce the MSS (Maximum Segment Size) of a distributed connection by the length of the GRE header.
  - This will allow the client TCP stack to build packets that allow for the 24 bytes of the GRE header to be added without any fragmentation being required.
  - ADJUSTDVIPAMSS on GLOBALCONFIG
    - Defaults to AUTO - Enables adjusted MSS
      - On target TCP/IP stacks when VIPAROUTE is being used
      - On Sysplex Distributor stack if it is also a target and VIPAROUTE is defined
    - If you are already exploiting VIPAROUTE and know that there's no fragmentation possible in your environment you can disable this function (specify NONE option)

"Shared Memory Communications over RDMA" concepts

Clustered Systems



RDMA technology provides the capability to allow hosts to logically share memory. The SMC-R protocol defines a means to exploit the shared memory for communications - *transparent* to the applications!

This solution is referred to as *SMC-R* (Shared Memory Communications over RDMA). SMC-R is an *open* sockets over RDMA protocol that provides transparent exploitation of RDMA (for TCP based applications) while preserving key functions and qualities of service from the TCP/IP ecosystem that enterprise level servers/network depend on!

Optimize server to server networking – transparently

"HiperSockets™-like" capability across systems

Network latency for z/OS TCP/IP based OLTP workloads ***reduced*** by up to ***80%*****

Networking related CPU consumption for z/OS TCP/IP based workloads with streaming data patterns ***reduced*** by up to ***60%*** with a *network throughput* increase of up to ***60%******

Shared Memory Communications (SMC-R):

Exploit RDMA over Converged Ethernet (RoCE) to deliver superior communications performance for TCP based applications

Typical Client Use Cases:

Help to reduce both latency and CPU resource consumption over traditional TCP/IP for communications across z/OS systems

Any z/OS TCP sockets based workload can **seamlessly** use SMC-R without requiring any application changes

NEW **z/OS V2.1 SMC-R**

NEW **z/VM 6.3 support for guests**

NEW **10GbE RoCE Express**

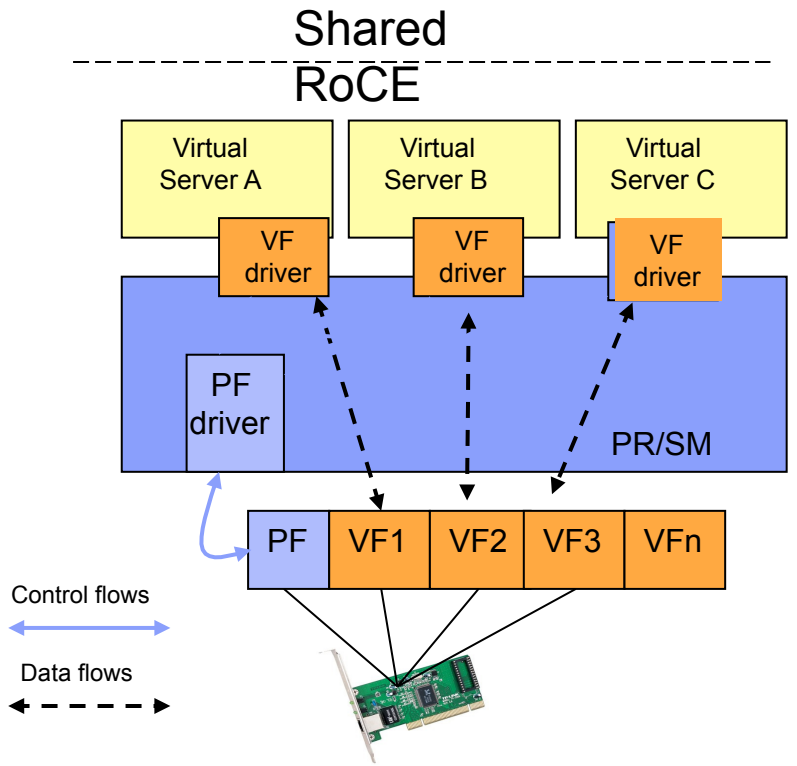
** Based on internal IBM benchmarks in a controlled environment of modeled z/OS TCP sockets-based workloads with request/response traffic patterns using SMC-R (10GbE RoCE Express feature) vs TCP/IP (10GbE OSA Express feature). The actual response times and CPU savings any user will experience will vary.

*** Based on internal IBM benchmarks in a controlled environment of modeled z/OS TCP sockets-based workloads with streaming traffic patterns using SMC-R (10GbE RoCE Express feature) vs TCP/IP (10GbE OSA Express feature). The actual response times and CPU savings any user will experience will vary.

SMC-R and Shared ROCE Support – IBM z13 System

- SMC-R requires a new RDMA capable NIC
  - 10GbE RoCE Express feature introduced in zEC12 GA2 and zBC12
    - Support for up to 16 RoCE Express features per zCPC
    - Cannot be shared across LPARS in initial deliverable (zEC12 GA2 and zBC12)
    - Each LPAR requires 2 RoCE Express features for High Availability
      - z/OS can only exploit a single port from each feature
- Shared RoCE support - *Available exclusively on **new IBM z13 System***
  - Allows concurrent sharing of a RoCE Express feature by multiple virtual servers (OS instances)
    - Efficient sharing for an adapter (getting the Hypervisor out of the data path)
    - Up to 31 virtual servers (LPARs or 2nd level guests under zVM)
    - Will also enable use of both RoCE Express ports by z/OS
  - z/OS support will be available in z/OS V2R2 (base) and on z/OS V2R1 via APAR/PTF
    - z/OS V2R1: APAR OA44576 (PTF UA76424)

10GbE RoCE Express



SMC-R adapter virtualization: Overview

- Multiple PFIDs (PCIe Function IDs) with unique Virtual Function IDs are configured for each physical adapter (PCHID) in HCD (IOCDS)
  - Up to 31 PFIDs (VFs) supported per physical adapter
  - Each z/OS instance (LP or z/VM guest) sharing the adapter consumes a unique (at least one) PFID (each PFID has a corresponding Virtual Function ID / number)
  - Up to 16 physical adapters per CPC (no change)
- Adapter virtualization is transparent to application software

SMC-R adapter virtualization: TCP/IP Configuration

- No (minor) changes in TCP/IP configuration
  - Global Configuration Statement SMCR option continues to define PFID / port
- VFs are almost transparent to CommServer
- TCP/IP requires (consumes) a single PFID (VF) per port
  - All VLANs (if multiple) uses a single PFID (VF)
- Both physical 10GbE ports can be exploited - another PFID (VF) must be configured (per port)
- Multiple TCP/IP stacks (CINET) in a single system exploiting SMC-R:
  - Each TCP/IP stack requires a unique PFID (VF)
  - Potential migration consideration (prior to Shared RoCE support, multiple stacks could use the same PFID for SMC-R – each TCP/IP stack now needs a unique PFID)

SMC-R adapter virtualization: Product externals (VF number / PFIP)

RoCE virtualization is dynamically detected (first RoCE activation) and is fundamentally transparent … minor change in CommServer product externals

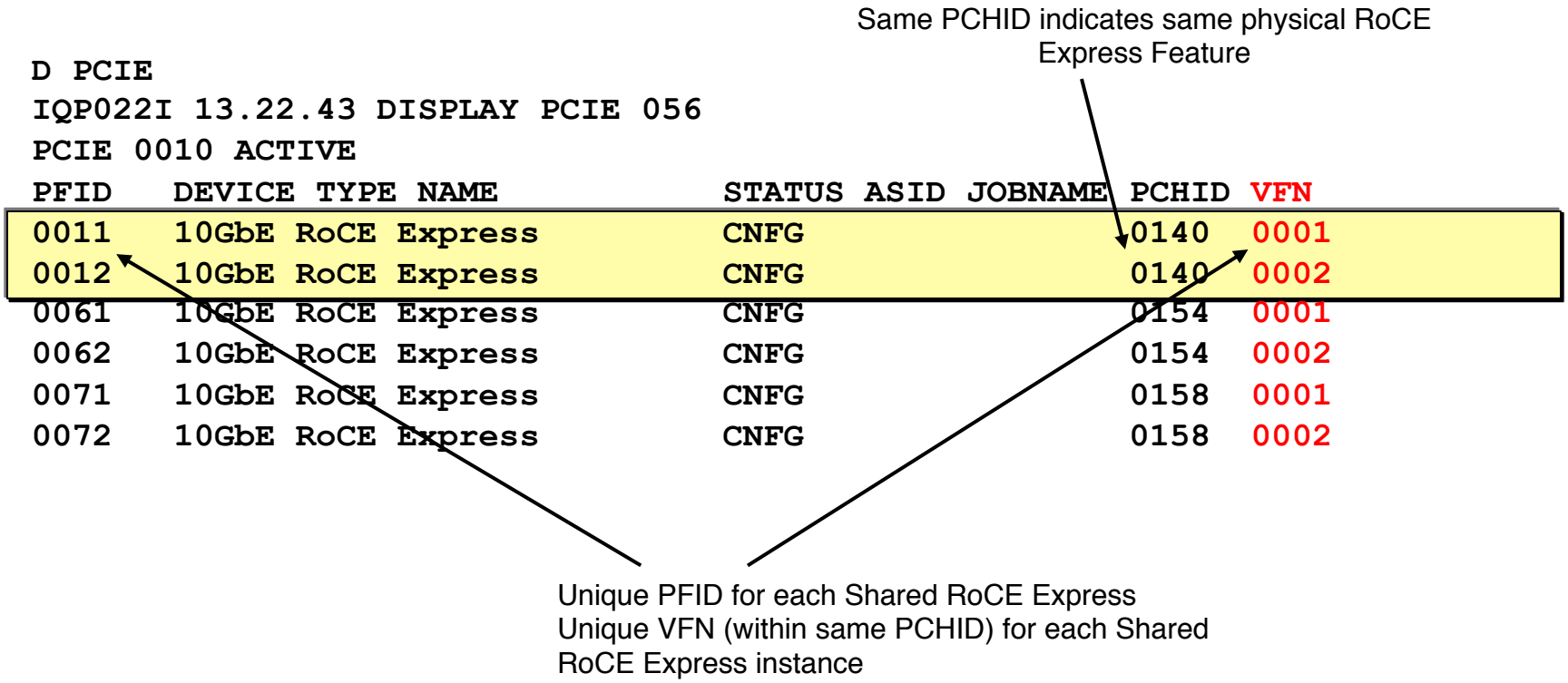
```
D NET,TRL,TRLE=IUT10011
IST097I DISPLAY ACCEPTED
NAME = IUT10011, TYPE = TRLE 341
IST1954I TRL MAJOR NODE = ISTTRL
IST486I STATUS= ACTIV, DESIRED STATE= ACTIV
IST087I TYPE = *NA* , CONTROL = ROCE, HPDT = *NA*
IST2361I SMCR PFID = 0011 PCHID = 0140 PNETID = PNETID1
IST2362I PORTNUM = 1 RNIC CODE LEVEL = **N/A**
IST2389I PFIP = 01000300
IST2417I VF = 0001
IST924I -------------------------------------------
IST1717I ULPID = TCPIP2 ULP INTERFACE = EZARIUT10011
IST1724I I/O TRACE = OFF TRACE LENGTH = *NA*
IST314I END
```

The **PFIP** (PCIE Firmware Internal Path) is displayed for each FID (PCHID). The first byte (value 0 or 1) denotes the path. For best practices (HA) the objective is to provision 2 FIDs (PCHIDs) having unique paths (opposite values 0 and 1). This display (PFIP) can be used to validate your configuration for HA.

When RoCE virtualization support is present VTAM TRLE display for RoCE PFIDs will include new MSG IST2417I displaying the VF number (ID) for this PFID

SMC-R adapter virtualization: z/OS DISPLAY PCIE

- z/OS PCIE display is updated to display the RoCE VF number



Determining SMC-R benefits

- Several customers have expressed interest in SMC-R
  - One of the first questions that is raised is “What benefit will SMC-R provide in my environment?”
    - Some users are well aware of significant traffic patterns that can benefit from SMC-R
    - But others are unsure on how much of their traffic is z/OS to z/OS and how much of that traffic is well suited to SMC-R
  - Reviewing SMF records, using Netstat displays, Ctrace analysis and reports from various Network Management products can provide these insights
    - **But it can be a time consuming activity that requires significant expertise**

SMC Applicability Tool

- A tool that will help customers determine the value of SMC-R in their environment with minimal effort and minimal impact
  - Part of the TCP/IP stack: Gather new statistics that are used to project SMC-R applicability and benefits for the current system
    - Minimal system overhead, no changes in TCP/IP network flows
    - Produces reports on potential benefits of enabling SMC-R
  - Also available ***now*** on existing z/OS releases via the following maintenance:
    - z/OS V2R1 - Apar PI29165, PTFs: UI24762 and UI24763
    - z/OS V1R13 - Apar PI27252 PTF UI24872
    - Does not require SMC-R to be enabled
    - Does not require RoCE Express Features or any specific System z processor
    - Can be used for determining potential benefits prior to moving to latest software and hardware levels

SMC Applicability Tool …

- Activated by Operator command - ***Vary TCPIP,,SMCAT,dsn(smcatconfig)*** – Input dataset contains:
  - Interval Duration, list of IP addresses or IP subnets of peer z/OS systems ((i.e. systems that we can use SMC-R for)
    - If subnets are used, the entire subnet must be comprised of z/OS systems that are SMC-R eligible
    - It is important that all the IP addresses used for establishing TCP connections are specified (including DVIPAs, etc.)
  - At the end of the interval a report is generated that includes:
    1. % of TCP traffic that is eligible for SMC-R (SMC-R Eligible Traffic)
       – All traffic that matches configured IP addresses
    2. % of SMC-R Eligible Traffic that is well suited to SMC-R (excludes workloads with very short lived TCP connections that have trivial payloads)
       – Includes break out of TCP traffic send sizes (i.e. how large is the payload of each send request)
       – Helps users quantify SMC-R benefit (reduced latency vs reduced CPU cost)

SMC Applicability Tool …

- Report includes 2 sections
  - Configured TCP traffic that could not use SMC-R without changes (does not meet direct route connectivity requirements)
    - This represents the opportunity of re-configuring routing topology to enable SMC-R
  - Configured TCP traffic that can use SMC-R as is immediately (meets SMC-R direct route connectivity requirements)
    - Detected by the tool automatically (non-routed traffic)

16744: z/OS CS: New Shared Memory Communications over RDMA (SMC-R), Part 2 of 2
Tuesday, March 3, 2015: 3:15 PM-4:15 PM
Issaquah A (Sheraton Seattle)
Speaker: Dave Herr (IBM Corporation)

SMC Applicability Tool …

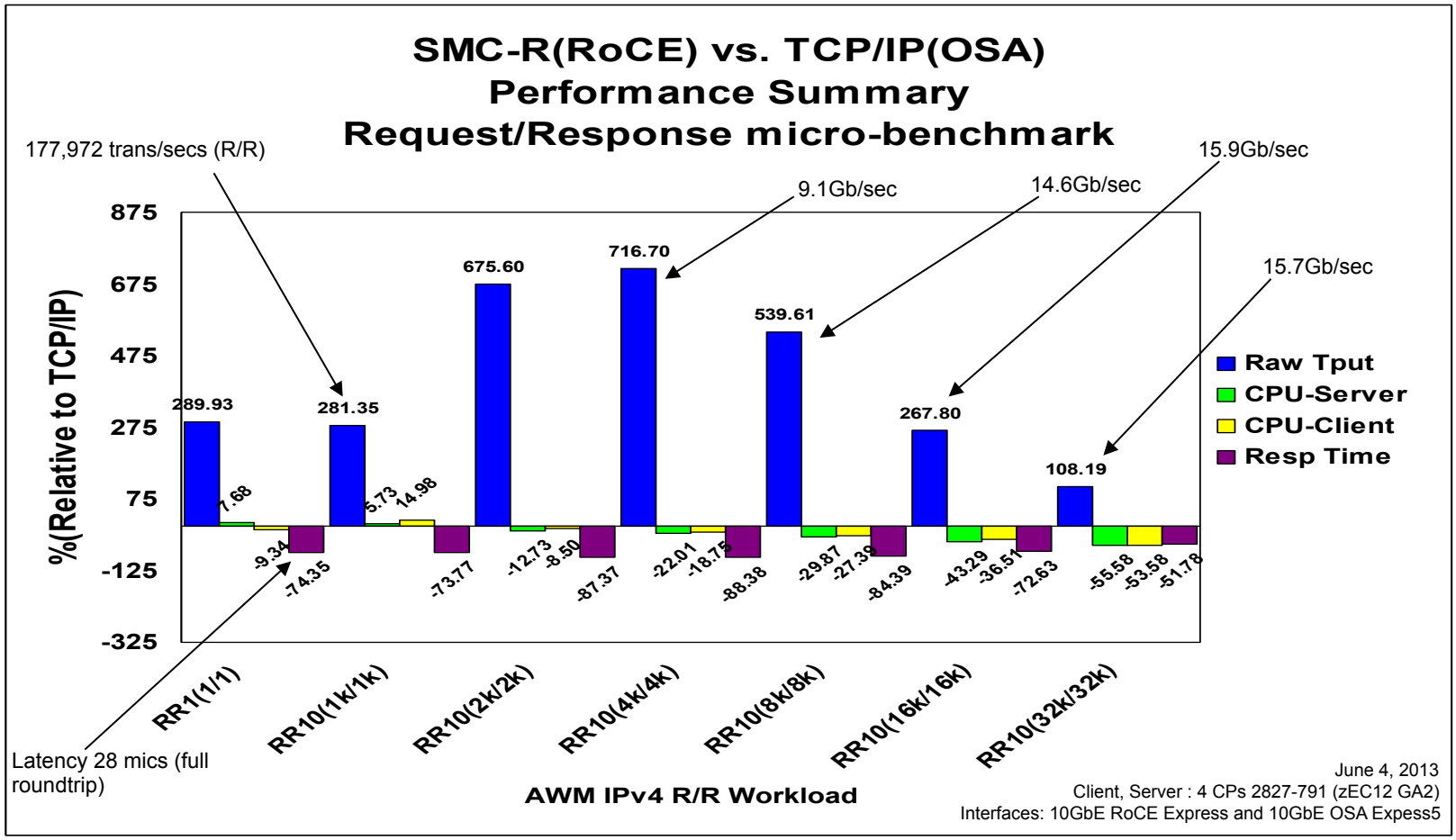
```
TCP SMC-R traffic analysis for matching direct connections
-----------------------------------------------------------
  Connections meeting direct connectivity requirements

     50% of connections can use SMC-R (eligible)
       67% of eligible connections are well-suited for SMC-R
     79% of total traffic (segments) is well-suited for SMC-R
       81% of outbound traffic (segments) is well-suited for SMC-R
       75% of inbound traffic (segments) is well-suited for SMC-R

  Interval Details:
    Total TCP Connections:                                      6
    Total SMC-R eligible connections:                           3
          Total SMC-R well-suited connections:                  2
    Total outbound traffic (in segments)                      274
          SMC-R well-suited outbound traffic (in segments)    222
    Total inbound traffic (in segments)                       211
          SMC-R well-suited inbound traffic (in segments)     159

    Application send sizes used for well-suited connections:
      Size                                  # sends     Percentage
      ----                                  -------     ----------
      1500 (<=1500):                              1      20%
      4K (>1500 and <=4k):                        1      20%
      8K (>4k and <= 8k):                         0       0%
      16K (>8k and <= 16k):                       0       0%
      32K (>16k and <= 32k):                      0       0%
      64K (>32k and <= 64k):                      1      20%
      256K (>64K and <= 256K):                    2      40%
      >256K:                                      0       0%

  End of report
```

How much of my TCP workload can benefit from SMC-R?

What kind of CPU savings can I expect from SMC-R?

SMC-R enhancements

- SMC-R Autonomics
  - Automatically cache SMC-R negative set-up attempts
    - Avoid future attempts to negotiate SMC-R with the specific peer
  - Automatically determine whether SMC-R is suitable for a given z/OS TCP Server
    - Workloads with very short lived connections and very small payloads may see no benefit from SMC-R
    - Automatically disables SMC-R negotiations for that server port
- Support 4K MTU for RoCE
  - In addition to existing 1K and 2K MTU
- Enhancements in reporting of SMC-R connection local and remote buffer sizes
  - Provided on Network Management Interfaces (NMI) and TCP/IP SMF records
    - NMI GetConnectionDetail API
    - SMF Record (Type 119)



Significant Latency reduction across all data sizes (52-88%)
Reduced CPU cost as payload increases (up to 56% CPU savings)
Impressive throughput gains across all data sizes (Up to +717%)

Note: vs typical OSA customer configuration
MTU (1500), Large Send disabled
RoCE MTU: 1K

Additional sessions related to SMC-R at Winter SHARE

16743: z/OS CS: New Shared Memory Communications over RDMA (SMC-R), Part 1 of 2
Tuesday, March 3, 2015: 1:45 PM-2:45 PM
Issaquah A (Sheraton Seattle)
Speaker: Gus Kassimis (IBM Corporation)

16744: z/OS CS: New Shared Memory Communications over RDMA (SMC-R), Part 2 of 2
Tuesday, March 3, 2015: 3:15 PM-4:15 PM
Issaquah A (Sheraton Seattle)
Speaker: Dave Herr (IBM Corporation)

16746: z/OS Communications Server Performance Update
Wednesday, March 4, 2015: 8:30 AM-9:30 AM
Issaquah B (Sheraton Seattle)
Speaker: Dave Herr (IBM Corporation)

Additional Information on SMC-R

- For additional information on SMC-R, including presentations, white papers, etc., please check out our web page:

"Shared Memory Communications over RDMA Reference Information" at

http://www-01.ibm.com/software/network/commserver/SMCR/

Applications /

Middleware /

Workload Enablement

CICS transaction tracking – Multiple ports of origin

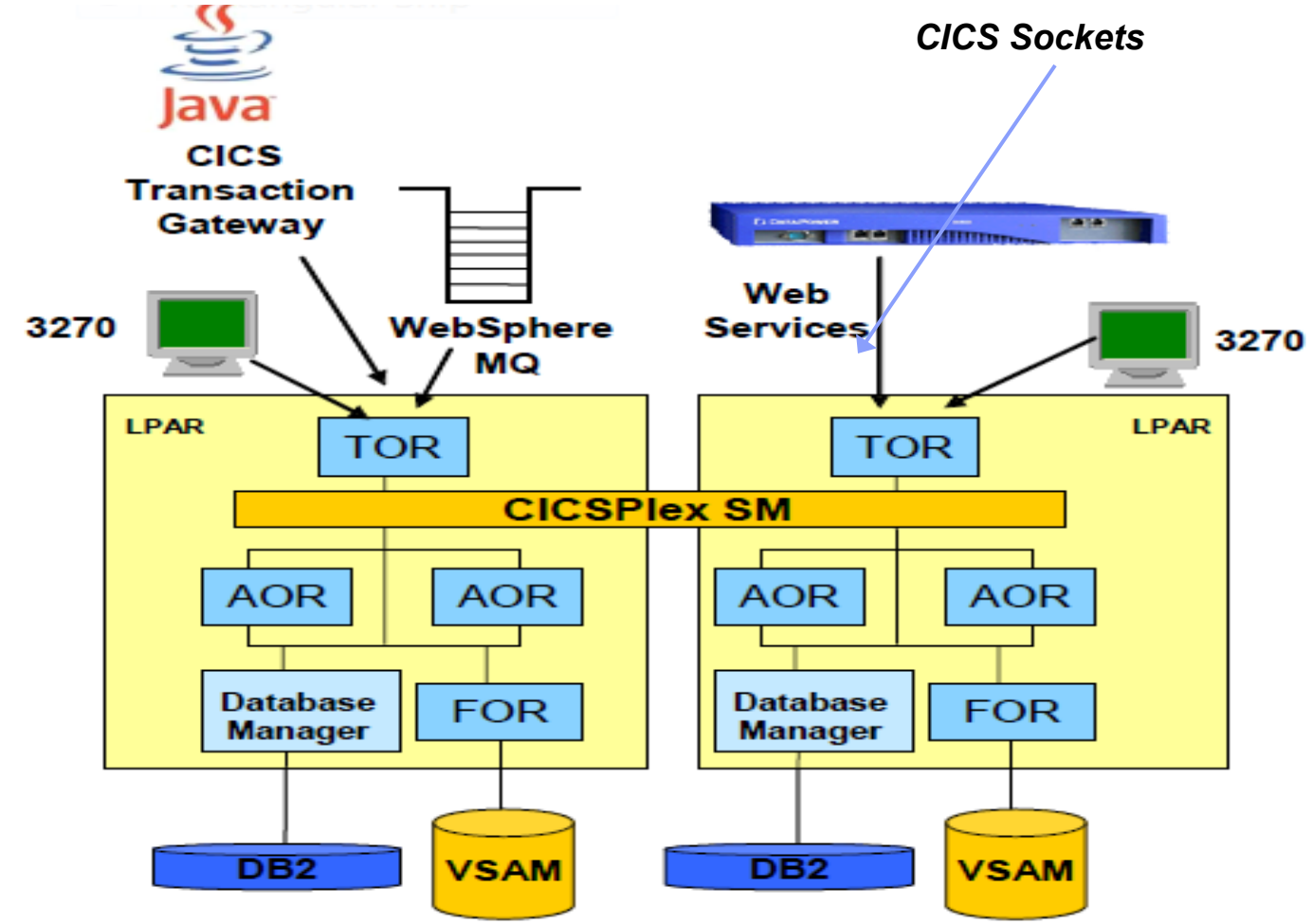
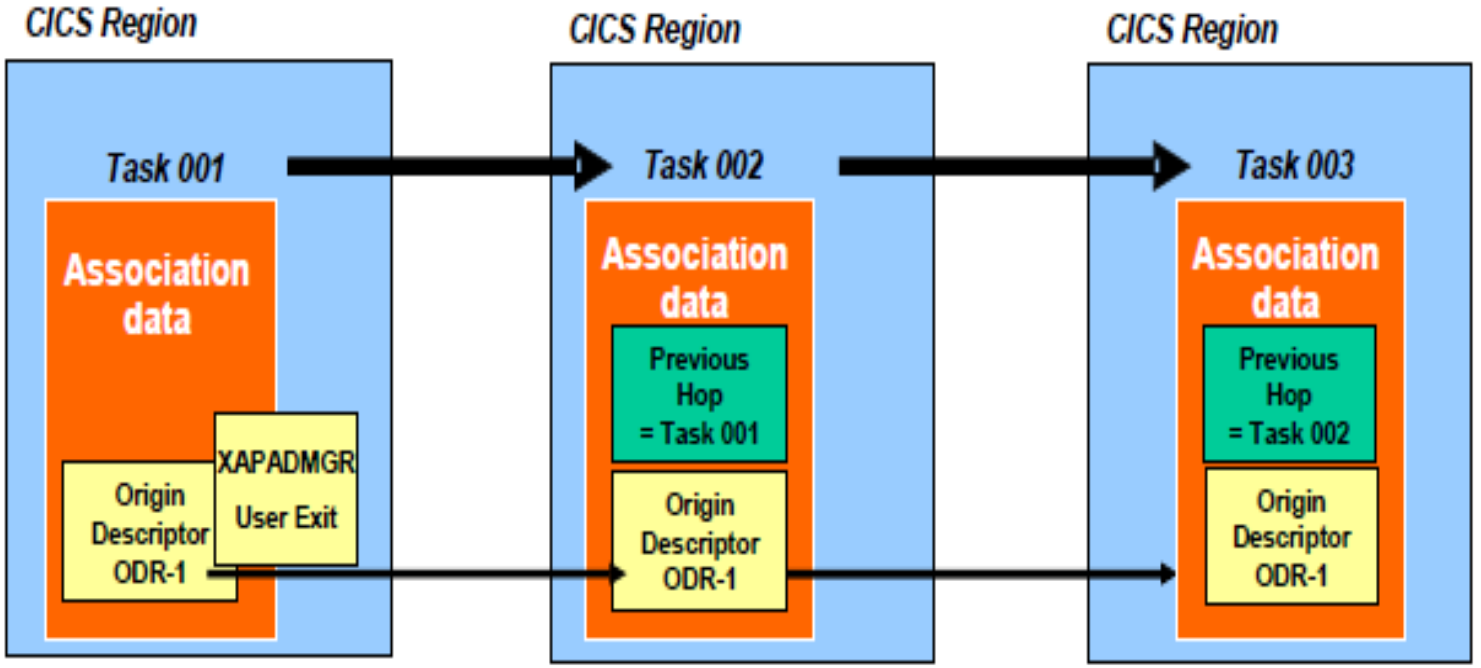


Figure 1. CICSplex with multiple front ends

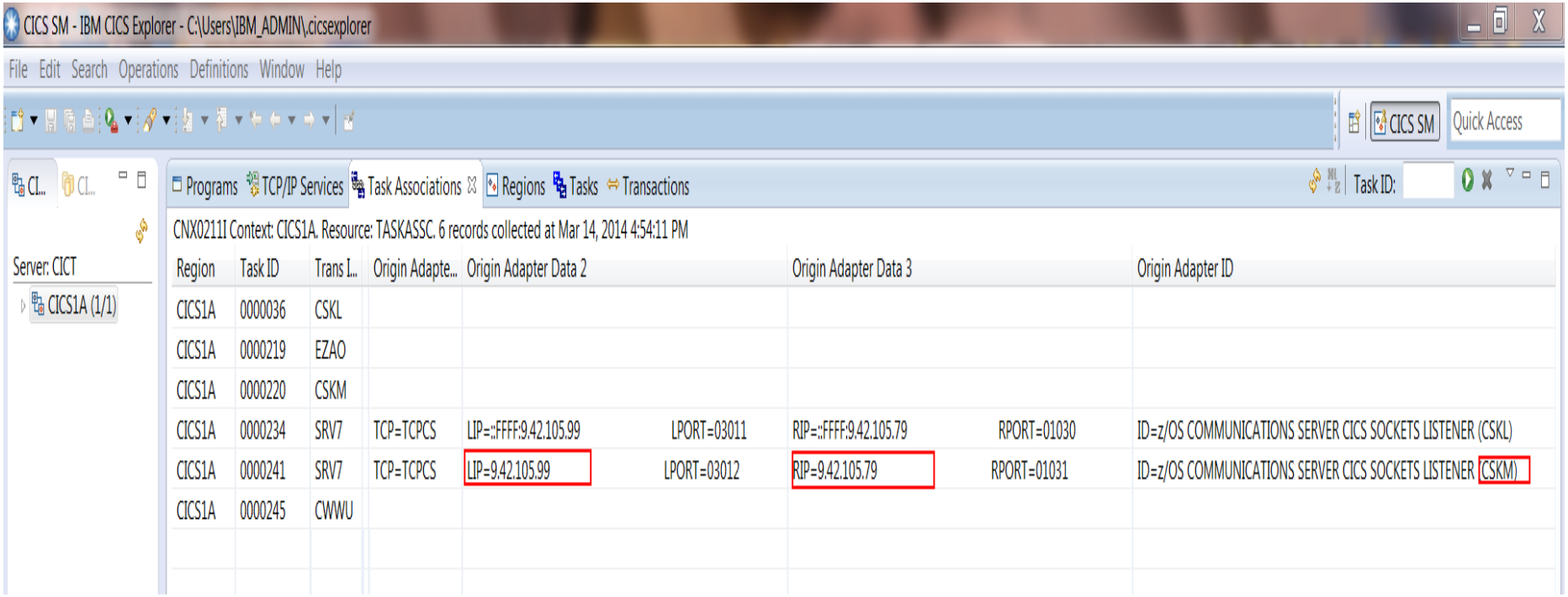
CICS transaction tracking – Propagating tracking info across CICS tasks/ transactions (CICS TS 4.2)



- CICS Transaction tracking enables you to locate a transaction in CICS based on knowledge of the entry point, such as an IP address, queue name, or SNA logical unit (LU) name. With this information, it is possible to use new search functions in the CICS Explorer® to search the CICSplex to locate other active tasks that have been initiated from the originating task, and to build a picture of the relationships between the associated tasks.

CICS transaction tracking support for CICS TCP/IP IBM Listener

- The z/OS CS CICS Sockets Listener (CSKL) is used extensively as a means to launch CICS transactions on behalf of CICS Sockets clients running on a variety of platforms.
  - CSKL does not support CICS transaction tracking prior to V2R2.
- In V2R2, the CICS Sockets Listener will provide to CICS the IP addresses and port numbers of the local and remote session partners for use by the CICS Explorer or Session Monitor.



CICS transaction tracking support for CICS TCP/IP IBM Listener …

- Parameters to be provided by the EZACIC01 TRUE:

Parameters	Value
ODAPTRID	ID=z/OS COMMUNICATIONS SERVER CICS SOCKETS LISTENER (CSKL)
ODAPTRDATA1	TCP=tcpip_name
ODAPTRDATA2	LIP=local IP address LPORT=local port number
ODAPTRDATA3	RIP=remote IP address RPORT=remote port number

Statement of Direction: z/OS Communications Server Internet mail applications: Sendmail and SMTPD (Issued Feb. 24, 2014)

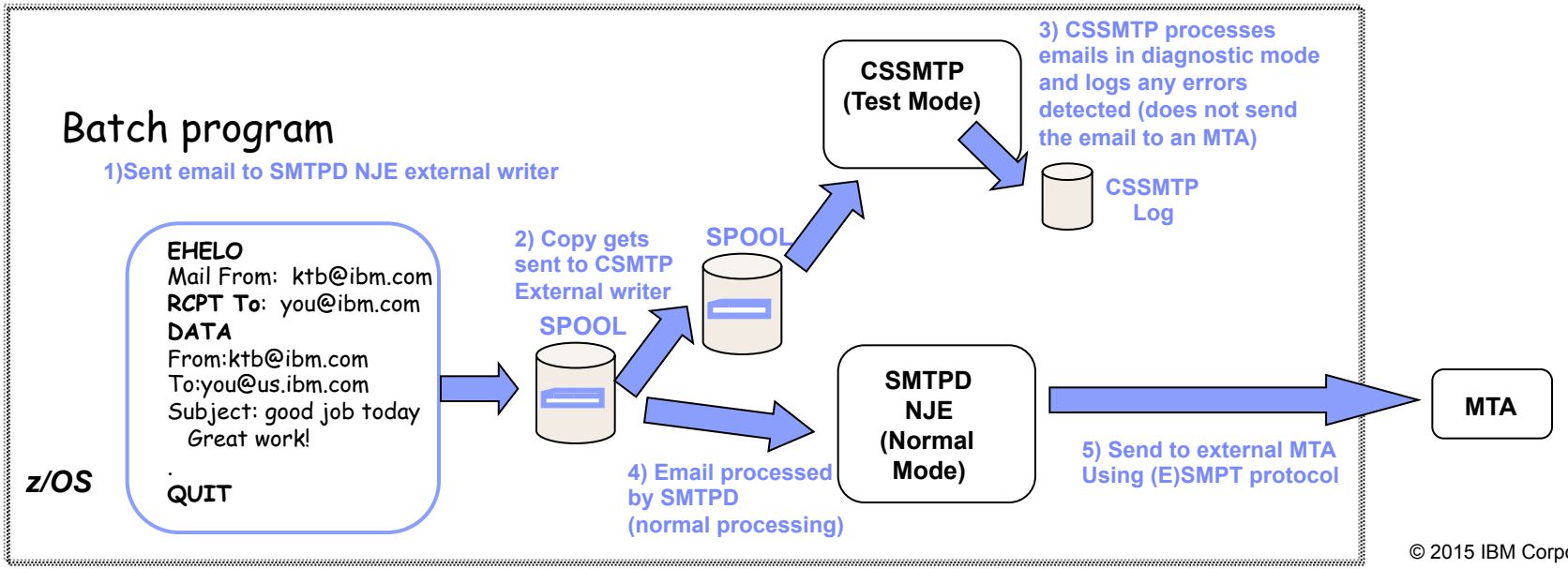
IBM intends to remove the Simple Mail Transport Protocol Network Job Entry (SMTPD NJE) Mail Gateway and Sendmail mail transports from z/OS Communications Server in the future. If you use the SMTPD NJE Gateway to send mail, IBM recommends you use the existing CSSMTP SMTP NJE Mail Gateway instead. CSSMTP provides significant functional and performance improvements.

The Sendmail client program can also be used to send mail messages; IBM plans to provide a replacement function using CSSMTP as the SMTP transport, which will be designed so that it does not require application programming changes.

No replacement function is planned in z/OS Communications Server to support using SMTPD or Sendmail as a (SMTP) server for receiving mail for delivery to local TSO/E or z/OS UNIX System Services user mailboxes, or for forwarding mail to other destinations.

CSSMTP migration enablement

- Key Concern: “How to ensure my **production** mail workload will be processed successfully with CCSMTP”?
- Migration aid being considered for z/OS V2R2:
  - Allow for running SMTPD and CSSMTP (side by side) and processing the same sysout datasets with CSSMTP in “**Test Mode**”
    - SMTPD sends the mail as it does normally
    - CSSMTP processes the mail messages and produces a report indicating
      - “success and would have sent or would have failed”
- Migration Health Checks for SMTPD and Sendmail to alert users to the removal of these applications
  - Plan to roll these back to V2R1



Availability / Business Resilience

Activate resolver trace without restarting applications

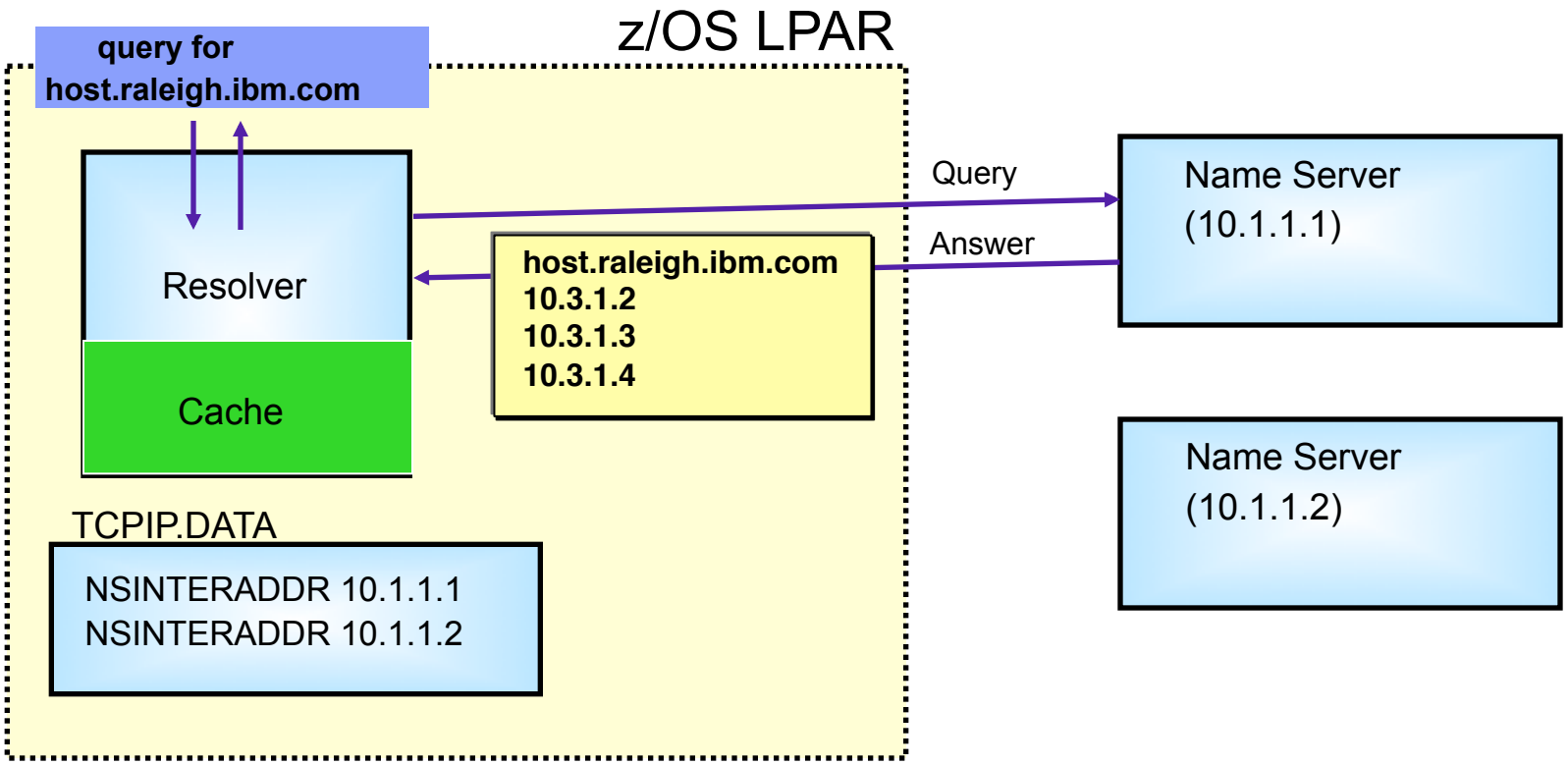
- Trace Resolver output can be used by application programmers and networking system programmers to diagnose problems in resolving hostnames to IP addresses or IP addresses to hostnames
- z/OS V2R2 Communications Server provides a Resolver CTRACE option which can be used to collect Trace Resolver information as Resolver CTRACE records
  - Allows users to dynamically enable or disable collection of Trace Resolver information for one or more applications without having to first stop and restart the application
  - New option on SYSTCPRE Component Trace (TRACERES option)
  - Also allows for more efficient trace data collection when a large number of trace resolver events need to be traced
- Users view the formatted component trace data from a dump, or from an external trace writer, using IPCS CTRACE subcommand processing.
  - The format of the trace is very similar to the existing Trace Resolver information
    - With some additional CTRACE record headers

Sample Resolver CTRACE Output under IPCS

```
==========================================================================00000094
MVSTST TRACERES 000A0002 13:12:18.900090 Formatted Trace Resolver
ASID.... 0024 TCB.... 007F83F0 JOBN.... USER23 CID.... 00000004
Resolver Trace Initialization Complete -> 2014/10/08 13:12:18.869790
res_init Resolver values:
Setup file warning messages = No
CTRACE TRACERES option = Yes
Global Tcp/Ip Dataset = None
Default Tcp/Ip Dataset = None
Local Tcp/Ip Dataset = SYS1.TCPPARMS(TCPDATA)
...
(*) Options NDots = 1
(*) SockNoTestStor
(*) AlwaysWto = NO (*) MessageCase = MIXED
(L) LookUp = DNS
(L) NoCache
res_init Succeeded
==========================================================================000000AD
MVSTST TRACERES 000A0002 13:12:18.929192 Formatted Trace Resolver
ASID.... 0024 TCB.... 007F83F0 JOBN.... USER23 CID.... 00000004
res_init Started: 2014/10/08 13:12:18.929172
res_init Ended: 2014/10/08 13:12:18.929188
**************************************************************************
```

z/OS V1R11 introduced resolver caching

- Provide maximum performance with minimal configuration
- Resolver cache queried for each request
- Communication with name server only if cache information not available



Reordering of cached resolver results

- System Resolver Caching allows for the system-wide caching of Domain Name System (DNS) responses. The primary advantage of caching is the improved performance that is obtained by the elimination of repetitive queries to the name servers.
- DNS implementations can reorder the list of IP addresses returned for a given host name through the query in a round robin fashion. This allows some basic load balancing of IP addresses used by clients.
- Although Resolver APIs (getaddrinfo(), gethostbyname()) return a list of IP addresses, most applications only use the first IP address in the list.
- The System Resolver Cache function saves the results from DNS for a given hostname and always returns them in the same order as cached data. This defeats any round-robin processing performed by the name server.

Reordering of cached resolver results …

- z/OS V2R2 Communications Server enhances the cache support for the system resolver. The new enhancement allows for the system-wide round-robin reordering of the list of IP addresses that are associated with a cached hostname.
- A new CACHEREORDER statement in the resolver setup file enables the system-wide resolver reordering of cached data.
  - When a hostname has several IP addresses the results will be reordered for every query
  - Note: If getaddrinfo() is invoked (for IPv4 or IPv6) the reordering occurs within the context of Default Address Selection rules
- To disable system-wide resolver reordering, specify the NOCACHEREORDER statement in the resolver setup file. (This is the default.)
- To disable resolver reordering for an application, specify the NOCACHEREORDER statement in the TCPIP.DATA file.
- The Resolver NMI (GetResolverConfig) will be updated to return the new Resolver setup statements.

Reordering of cached resolver results …

- Resolver reorders the cached information on a resolution query basis
  - Reordering is independent of which application issues the query
  - Reordering is independent of which type of query (Gethostbyname or Getaddrinfo) is issued
- Resolver reorders IPv4 and IPv6 resource information separately
  - Resolver reorders the list before performing any sorting
    - Gethostbyname results sorted based on SORTLIST configuration statement
    - Getaddrinfo results sorted based on default destination address selection algorithm

Application A issues gethostbyname for host.raleigh.ibm.com

host.raleigh.ibm.com
10.3.1.2
10.3.1.3
10.3.1.4

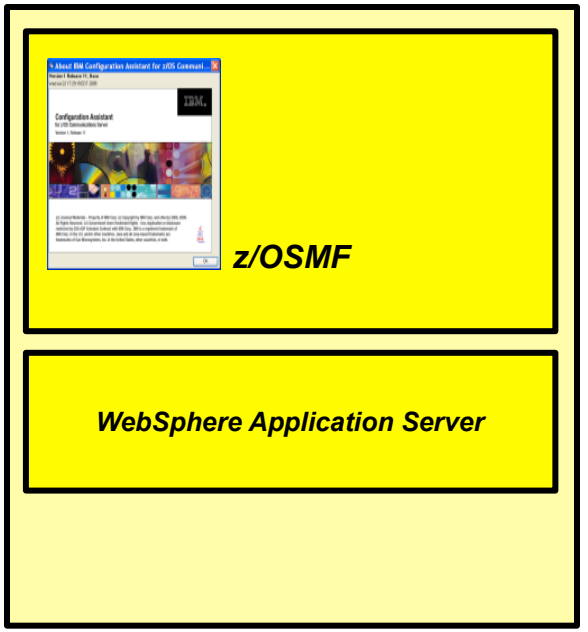
Application B issues gethostbyname for host.raleigh.ibm.com

host.raleigh.ibm.com
10.3.1.3
10.3.1.4
10.3.1.2

Simplification

Configuration Assistant: TCP/IP stack configuration

- Skilled z/OS system programmers and administrators are an aging skillset, leading to concerns about future skill shortages.
- Configuration Assistant (CA) only supports configuration of z/OS CS policy-based networking functions, such as IPSec, AT-TLS, and IDS.
- While TCP/IP configuration is not that complex, some aspects are not intuitive.
- User must look through a lot of documentation.
- Some statements are not easy to configure.

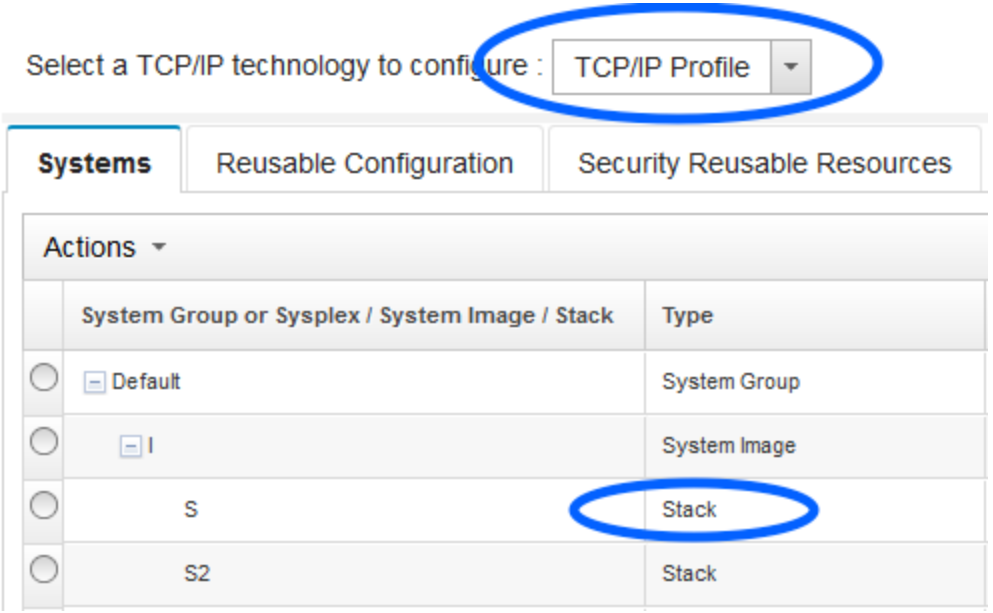


V2R1 Configuration Assistant: Interface for Communications Server policy based definition, installation and activation

16946 & 16947: z/OSMF Configuration Assistant for z/OS CS 2.1 Hands-on Lab, Parts 1 & 2
Tuesday, March 3, 2015: 11:15 AM-12:15 PM & 12:30 PM-1:30 PM
Redwood (Sheraton Seattle)
Speaker: Todd Valler and Linda Harrison (IBM Corporation)

Configuration Assistant: TCP/IP stack configuration …

- V2R2 will provide a new “TCP/IP” configuration perspective in the CA
- Support will be provided for both novice and more experienced users.
- CA will include integrated health checks for best practices configuration, including migration health checks for deprecated functions.



  - The configuration model will support “levels of configuration” which include a sysplex level, image level, and a stack level with the goal to allow for configuration to be applied for grouping of stacks that require related configuration.
  - CA will support the ability to generate modifications for an active TCP/IP profile (VARY OBEY support).
  - CA will assist with “install” of the generated configuration files as it does with policy configuration.

Configuration Assistant: TCP/IP stack configuration …

Configuration Assistant (Home) ▸ TCP/IP Profile ▸ TCP/IP Profile: Default.I.S

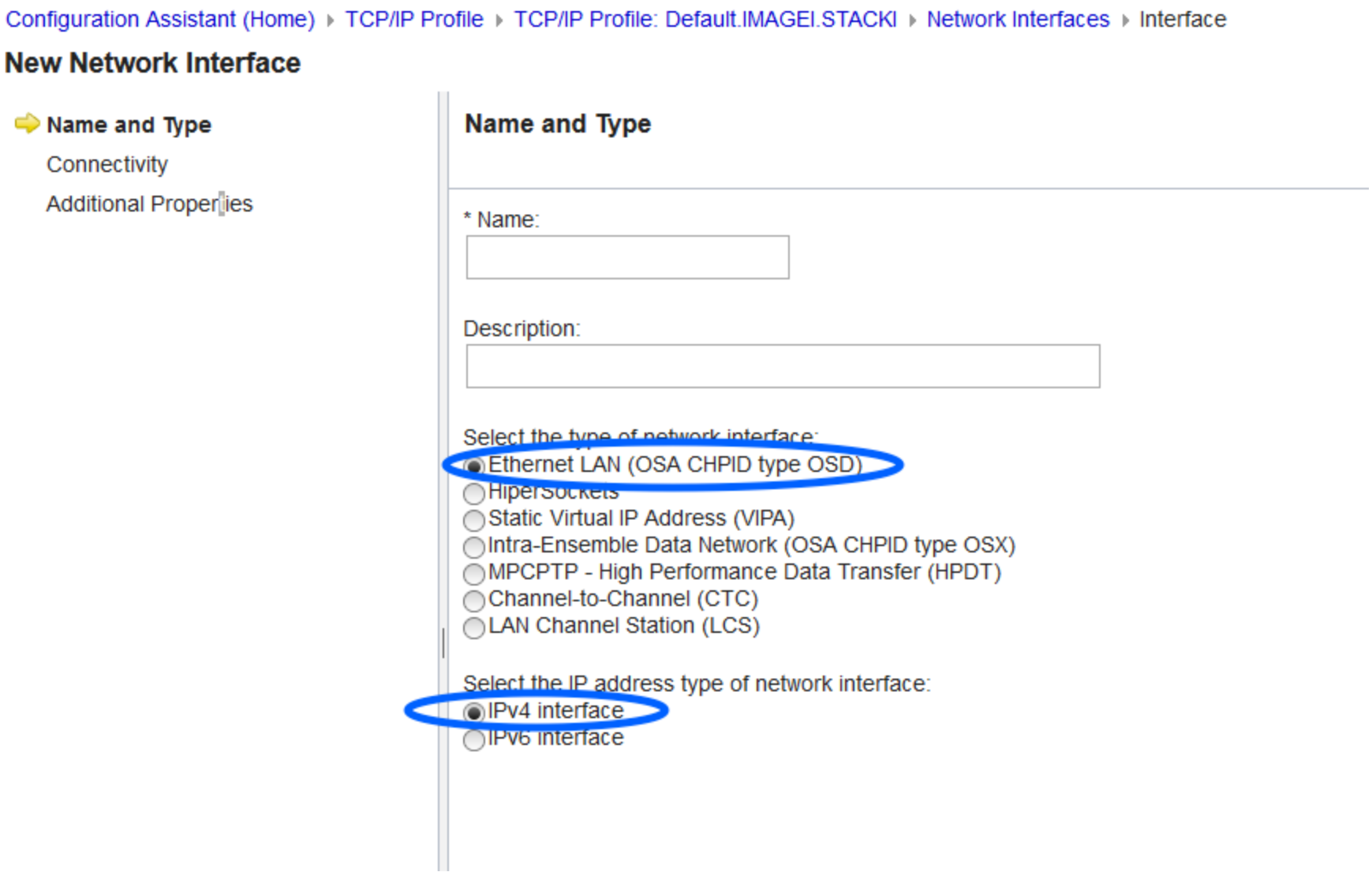
TCP/IP Profile for Group Default, System Image I, Stack S

Configure | Manage Reusable Configuration

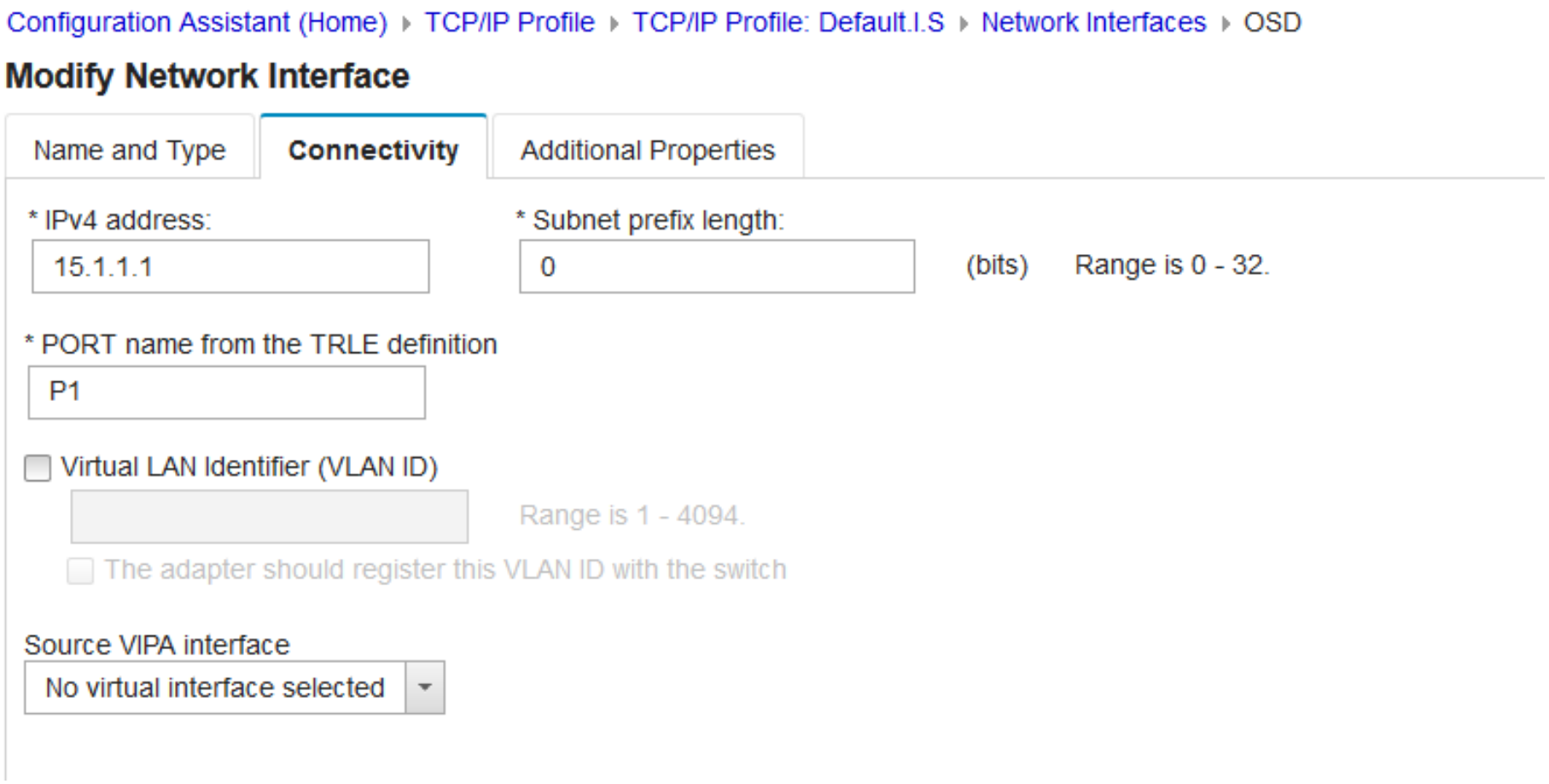
Use the following links to create and modify TCP/IP resources to define this stack's profile configuration.

TCP/IP Stack Resources	Status
Interfaces: Attach to networks	**Incomplete**
Routes: Connect to other systems	Configured
Ports: Reserve ports for TCP/IP applications	**Not configured**
Security: Control network access to and from the System	Configured
Source IP Addressing: Control outbound connection source IP addressing	Configured
Performance and Protocol: Tune your TCP/IP stack	**Not configured**
Management and Traces: Enable TCP/IP stack systems management and diagnosis	**Not configured**

Configuration Assistant: TCP/IP stack configuration …



Configuration Assistant: TCP/IP stack configuration …



Removed support for the GATEWAY statement in the TCP/IP profile

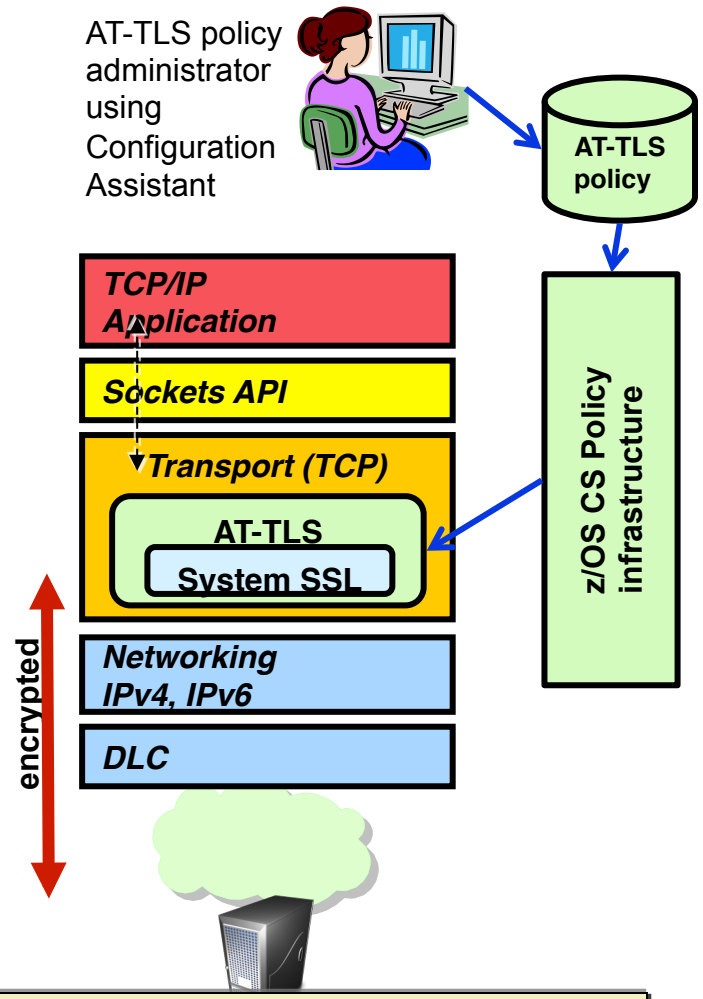
- Per a statement of direction included with the V2R1 preview announcement issued in February of 2013, z/OS V2R1 is the last release of z/OS to support the GATEWAY statement.
  - A health check was provided to warn that a GATEWAY statement was in use.
- The GATEWAY statement cannot be configured in V2R2. The BEGINROUTES block should be used instead.

```
BEGINRoutes
;
;    Destination Subnet Mask   First Hop  Link Name Packet Size
ROUTE 130.50.75.0 255.255.255.0 =        TR1       MTU 2000
ROUTE 193.5.2.0/24              =        ETH1      MTU 1500
ROUTE 9.67.43.0   255.255.255.0 =        FDDI1     MTU 4000
ROUTE 193.7.2.2   HOST          =        SNA1      MTU 2000
ENDRoutes
```

Security

TLS session reuse support for FTP and AT-TLS applications

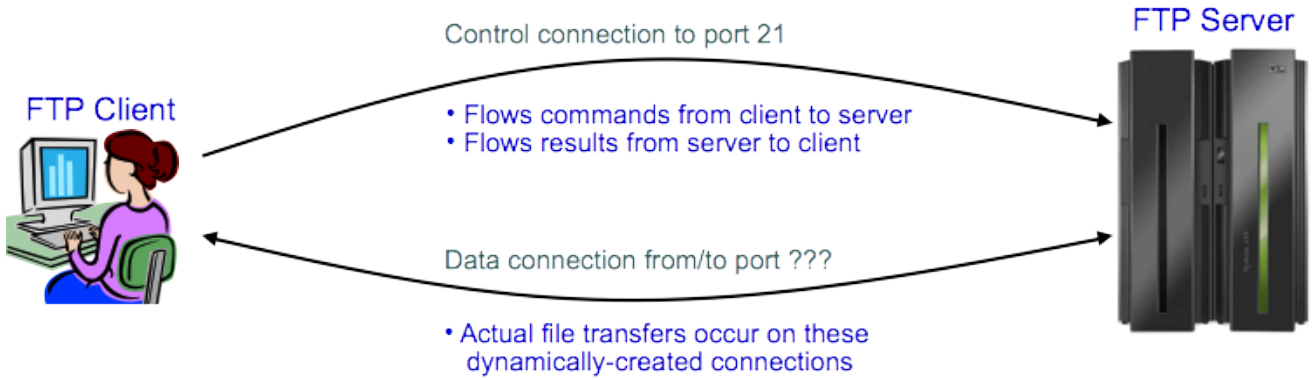
- Stack-based TLS
  - TLS process performed in TCP layer (via System SSL) without requiring any application change (transparent)
  - AT-TLS policy specifies which TCP traffic is to be TLS protected based on a variety of criteria
- Application transparency
  - Can be fully transparent to application
  - An optional API allows applications to inspect or control certain aspects of AT-TLS processing – "application-aware" and "application-controlled" AT-TLS, respectively
- Uses System SSL for TLS protocol processing
  - Remote endpoint sees an RFC-compliant implementation
  - Interoperates with other compliant implementations



16943: Leveraging z/OS CS AT-TLS for a Lower Cost and More Rapid TLS Deployment
Monday, March 2, 2015: 3:15 PM-4:15 PM
Issaquah B (Sheraton Seattle)
Speaker: Linwood Overby (IBM Corporation)

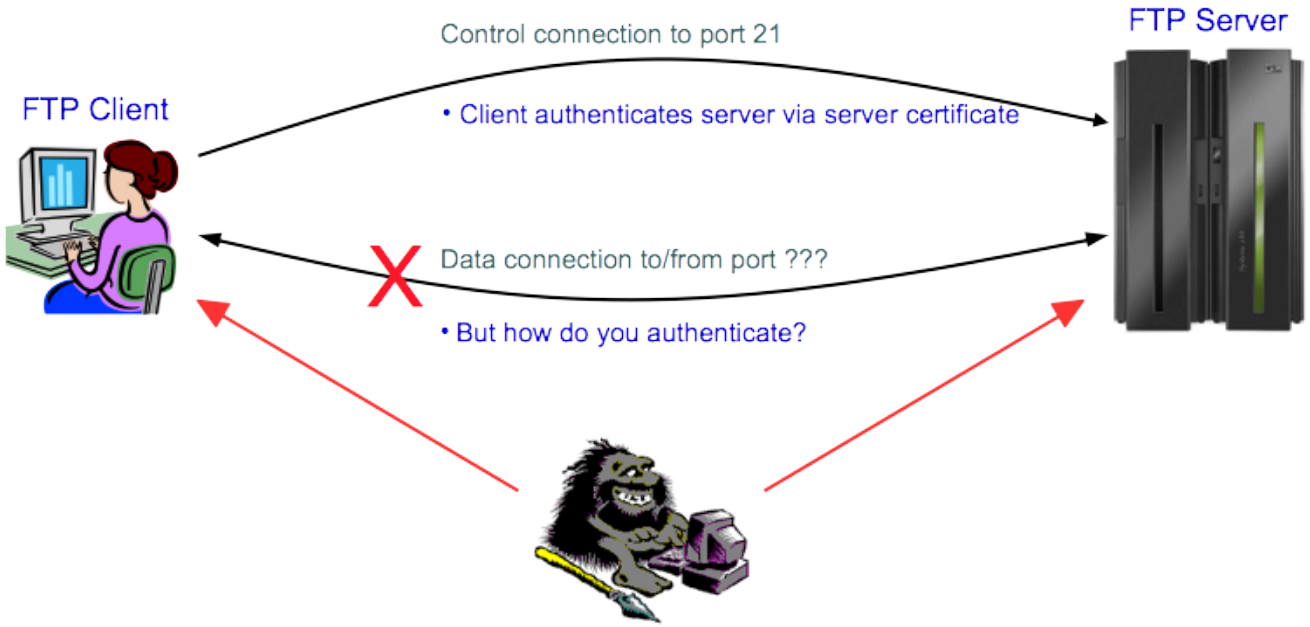
TLS session reuse support for FTP and AT-TLS applications

- FTP is a TCP-based protocol (default is port 21)
- Client initiates session (a “control connection”) to FTP daemon on server
- FTP daemon spawns a new FTP server process to handle the client’s session
- Client sends commands to server and receives replies on control connection. Examples include LIST, RETR, STOR, etc.
- RETR, STOR and other commands cause a separate “data connection” to be established on a different set of ports between the server process and the client:
  - Active mode: server initiates data connection to the client
  - Passive mode: client initiates data connection to server
  - Regardless of active/passive mode, FTP client ALWAYS initiates SSL sessions.



TLS session reuse support for FTP and AT-TLS applications …

- When server authentication is used (common for FTP), the FTP server does not receive the client's certificate and therefore cannot authenticate the client. So there's no way to check the identity associated with the control and data connections, and ensure that the control and data connections are from the same source.



TLS session reuse support for FTP and AT-TLS applications …

- RFC 4217 (Securing FTP with TLS) specifies that it is reasonable for the server to insist that the data connection uses a TLS cached session.
- System SSL sessions are bound to specific TCP ports, so a session created for a control connection cannot be reused on an ephemeral data port.
  - The result is that when using the z/OS FTP client, you must disable session reuse enforcement on the server if it was enabled.
- In V2R2, z/OS System SSL will allow session reuse without port binding.
  - The z/OS FTP client and server will exploit this capability of System SSL to support reusing the session ID (SID) of the control connection or a previous data connection on the following data connections within an FTP session.
  - This support will be provided for both FTP's native SSL support and for AT-TLS.

TLS session reuse support for FTP and AT-TLS applications …

- A new FTP.DATA statement SECURE_SESSION_REUSE is added for both the FTP client and server. This statement is used to specify whether session reuse is required when SSL/TLS is being used to protect the connections.
- Client: Specifies whether the client requires session reuse when SSL/TLS is being used to protect the connections.

```
        +--SECURE_SESSION_REUSE NONE----------+
>>------+-------------------------------------+-----><
        +--SECURE_SESSION_REUSE--+--NONE------+
                                 +--ALLOWED---+
                                 +--REQUIRED--+
```

- Server: Specifies whether the server requires session reuse when SSL/TLS is being used to protect the connections.

```
           +--SECURE_SESSION_REUSE ALLOWED----------+
>>---------+---------------------------------------+-----><
           +--SECURE_SESSION_REUSE-----+--ALLOWED---+
                                       +--REQUIRED--+
```

TLS session reuse support for FTP and AT-TLS applications …

- The SECURE_SESSION_REUSE value is made available via:
  - The FTP client user exit (EZAFCCMD)
  - The GetFTPDaemonConfig request type of the TCP callable NMI (EZBNMIFR)
  - The FTP daemon general configuration section in the type 119 subtype 71 SMF record

TLS session reuse support for FTP and AT-TLS applications …

- AT-TLS controlling applications can use the SIOCTTLS IOCTL:
  - To retrieve the session ID for the secure socket.
  - To request that a session be is reused on a socket by retrieving and setting the session token.
- The session ID and the required yes/no field for a session secured with AT-TLS will be available via:
  - The TCP connection termination SMF record (type 119, subtype 2).
  - The FTP security in SMF 119 records (subtypes 3,70,72, and 100-104)
  - The GetConnectionDetail (NWMTcpConnType) NMI.
  - The SNMP ibmMvsTcpConnectionTtls_xxxx MIB object (to be defined)
  - The Netstat TTLS/-x output

```
MVS TCP/IP NETSTAT CS V2R1 TCPIP Name: TCPCS 19:51:22
     ConnID: 000000B8
       …
       SecLevel: TLS Version 1.2
       …
       FIPS140: Off
       Session ID: abcdeffeaffe132450559ddddd000200220
     TTLSRule: ftp_serv_21
```

Simplified access permissions to ICSF cryptographic functions for IPSec

- The TCP/IP stack's IPSec support calls ICSF callable services to perform a variety of cryptographic operations.
- The TCP/IP stack often runs under the SAF credentials (ACEE) of the calling application, so IPSec operations also run under the caller's credentials.
  - A consequence of this is that anyone who uses IPSec and specifies CHECKAUTH(YES) in their ICSF options dataset must have access to the SAF resources that protect the ICSF services that IPSec uses.
  - Customers who specify CHECKAUTH(YES) in their ICSF options dataset have complained about having to authorize so many users to the relevant ICSF SAF resources. If the TCP/IP stack used its own credentials when calling ICSF instead of the caller's, only the TCP/IP userid would require access to the SAF resources.
- In V2R2, ICSF is providing a new CSFACEE function that will allow TCP/IP to invoke the necessary ICSF functions under its own credentials instead of the calling application’s credentials.

AT-TLS certificate processing enhancements

- In V2R2, System SSL will add the following new functions:
  - Ability to retrieve Certificate Revocation Lists (CRLs) via HTTP
  - Caching enhancements for CRLs retrieved via LDAP
  - Support for RFC5280 compliant certificate validation
  - Support for Online Certificate Status Protocol (OCSP)
    - See next chart
- AT-TLS will expose these new capabilities. Doing so requires:
  - Policy changes
  - Configuration Assistant changes
  - Enhance the NETSTAT TTLS command and pasearch -t display for new option values
  - IPCS formatter updates to format new policy
  - No IOCTL, NMI, or SMF changes since this level of configuration is not reported via these mechanisms

Certificate Revocation mechanisms: A comparison

Certificate Revocation Lists (CRLs)

- CRLs are lists of X.509 certificates that have been revoked by the CA
- Signed by the CA
- Requires a fair amount of work on the certificate users's part:
  - Download the CRL from an LDAP or HTTP server
  - Parse the document and (typically) store it in a cache
  - Search the list for a matching certificate serial number
  - CRLs can be very large, increasing network bandwidth and processing burden
- Design introduces (almost guarantees) the potential for CRLs being out of date

Online Certificate Status Protocol (OCSP)

- HTTP-based request/response protocol
- OCSP client sends request to OCSP responder (the server) with an identifier of the particular certificate to check
- OCSP responder replies with a "good", "revoked" or "unknown" indication. Responses are signed (like CRLs)
- OCSP responders typically operated by the Certificate Authority so that the responder has realtime access to the CA's certificate status database
- Eliminates much of the client-side burden involved with CRLs and provides realtime status, minimizing or eliminating the potential for out of date revocation information

NIST compliance for DCAS, SNMP, Sendmail, and centralized policy agent

- A recent NIST mandate (SP800-131a) stated that by the end of 2013, U.S. Government systems had to support TLSv1.1, TLSv1.2, SHA-2 hashes and encryption key strengths of 112 bits or more.
- This required updates to the following z/OS CS functions:
  - Centralized policy agent
  - Sendmail
  - SNMP
  - Digital Certificate Access Server (DCAS)
- In V2R2, the centralized policy agent is updated to support the TLSv1.1 protocol and TLSv1.2 protocol with its 2-byte cipher suites.
  - You can now define these new 2-byte cipher suites on the ServerConnection statement / ServerSSLV3CipherSuites option.
- V2R2 adds support to the Sendmail client and server to now support TLSv1.1 and TLSv1.2 with a new set of ciphers.
- The z/OS CS SNMP Agent, the z/OS UNIX SNMP command, and the SNMP manager API are enhanced to support the Advanced Encryption Standard (AES) 128-bit cipher algorithm as an SNMPv3 privacy protocol for encryption.
  - The z/OS Integrated Cryptographic Services Facility (ICSF) is required for AES 128-bit cipher encryption.
- DCAS is enhanced to support TLSv1.1 or TLSv1.2 with the new set of TLSv1.2 2-byte ciphers.
  - To do this, DCAS is enhanced to use AT-TLS for the TLS/SSL client connection.
  - A new TLSMECHANISM keyword can be used to specify whether to use AT-TLS policies or IBM System SSL to connect to the client.

NIST compliance for DCAS, SNMP, Sendmail, and centralized policy agent (cont)

- To comply with the NIST SP800-131a mandated 2013 availability, these security enhancements to DCAS, SNMP, Sendmail, and the centralized policy agent have been made available on z/OS V2R1 via the following APARs/PTFs:

Function	APAR Number	PTF Number
Policy agent	PM96891	UI13120
Sendmail	PM96896	UI13138
DCAS	PM96898	UI13139
SNMP	PM96901	UI13140

Recommended White Papers and References

- "IBM 3270 emulation: security considerations":
http://www-01.ibm.com/common/ssi/cgi-bin/ssialias?subtype=WH&infotype=SA&appname=STGE_ZS_ZS_USEN&htmlfid=ZSW03276USEN&attachment=ZSW03276USEN.PDF
- "Shared Memory Communications over RDMA (SMC-R) Security Considerations:"
https://w3-03.sso.ibm.com/sales/support/ShowDoc.wss?docid=ZSW03255USEN

Miscellaneous

End of support for CCL, XOT, NCP and SSP (Issued Feb. 3, 2015)

IBM has announced end of support, effective on March 31, 2016, for IBM Communication Controller for Linux (CCL), IBM X.25 over TCP/IP for Communication Controller for Linux (XOT), IBM Advanced Communication Function/Network Control Program (ACF/NCP), and IBM Advanced Communication Function/System Support Program (ACF/SSP).

Since neither CCL nor NCP will be supported after March 31, 2016, CCL and NCP customers will need to investigate alternate technologies, and migrate to the selected alternatives by that date. The alternate technologies to consider are dependent on the NCP-related functions and features currently in use.

16745: Enterprise Extender on z/OS CS: SNA Hints and Tips
Thursday, March 5, 2015: 11:15 AM-12:15 PM
Issaquah A (Sheraton Seattle)
Speaker: Sam Reynolds (IBM Corporation)

Please complete your session evaluation

- z/OS V2R2 Communications Server Technical Update, Part 1
- Session # 16739
- QR Code:

Please complete your session evaluation

- z/OS V2R2 Communications Server Technical Update, Part 2
- Session # 16740
- QR Code: