z/OS Network update

Comm Server 1.8

Large Systems Update 2006

olle.zetterlund@se.ibm.com

AGENDA

1. Sysplex Enhancements
2. Application Enhancements
3. Enterprise Extender and SNA Enhancements
4. IPv6 on z/OS Communications Server
5. Security

Before we get to comm server..

Not Comm Server – but still Communication news…..

NJE over TCP/IP

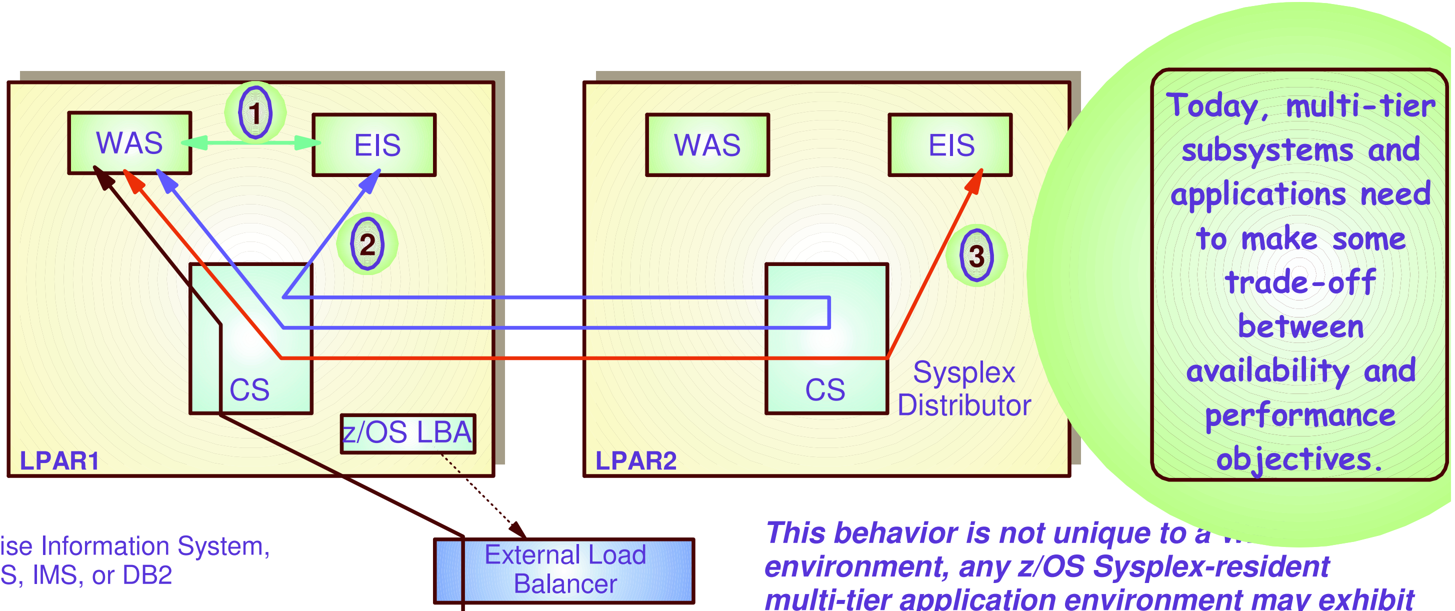
- JES2/JES3 supports NJE over SNA and BSC networks
- Could prevent migration off 3745
- VM(RSCS), iSeries and VSE/POWER all have supported NJE over TCP/IP for years
- So...
    - JES2 z/OS 1.7 supports established TCP/NJE protocol
        - Enabled by APAR OA12364 avail 1Q06
    - JES3 support z/OS 1.8
    - NJE Improvements:
        - Support for SSL/TLS (using AT/TLS)
        - Stronger authentication

1. Sysplex Enhancements
2. Application Enhancements
3. Enterprise Extender and SNA Enhancements
4. IPv6 on z/OS Communications Server
5. Security

Sysplex Enhancements

- **Focus areas:**
  - Extending the autonomic behavior in error scenarios for the IP Sysplex
  - Improved quality in workload distribution decisions made by Sysplex Distributor
  - Improved operator control

Local vs. remote connector support in today's z/OS environment



Today, multi-tier subsystems and applications need to make some trade-off between availability and performance objectives.

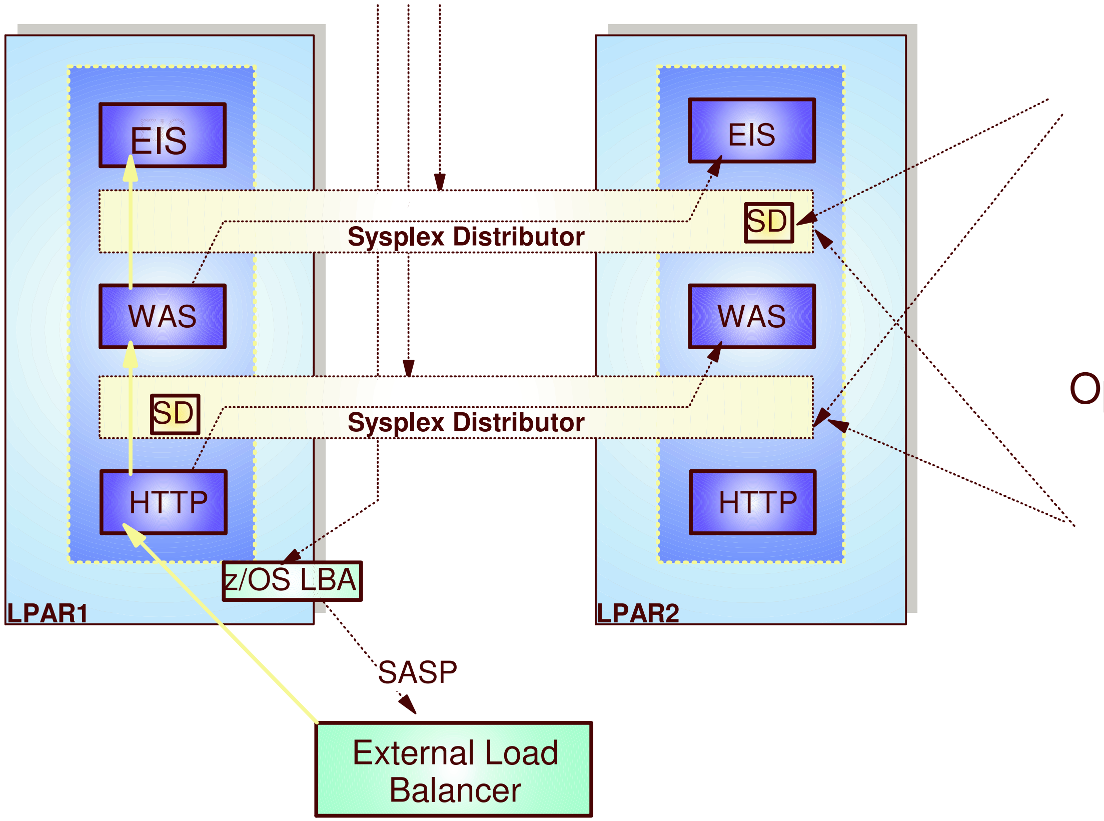
EIS: Enterprise Information System, such as CICS, IMS, or DB2

This behavior is not unique to a [illegible] environment, any z/OS Sysplex-resident multi-tier application environment may exhibit similar behavior and have similar issues.

- **Local connectors (1)**
  - Optimized high-speed path (based on local services, such as cross-memory services and RRS)
  - Concern - what happens if local target is not available
    - No automatic switch to alternate target on another LPAR
    - WAS transactions may complete fast causing WLM to prefer that LPAR for increased workload (storm-drain issue)
- **Remote connectors (2 and 3)**
  - Uses TCP/IP for communication
  - Sysplex Distributor (or other load balancer) selects a target among any available targets in the Sysplex
  - If target is local and Sysplex Distributor is remote, communication path is not efficient (2)
  - It is not today possible to favor a local target even if one exists and has capacity

Improved multi-tier application support by Sysplex Distributor

1. WLM LPAR and server-specific performance weights
2. TCP/IP stack server-specific health weights



Level of local favoritism can be configured

- Always choose local target if target is available and healthy
- Control level of WLM weight impact on target selection

Optimized traffic flow:

- "Distributed" Sysplex Distributor logic in each stack avoids cross-LPAR flows for connection setup when local target is chosen.
- Avoids traffic routing via SD-owning LPAR for local targets

Improved multi-tier application support by Sysplex Distributor - optimized for local performance without losing availability

Application endpoint awareness via enhanced Sysplex sockets API processing

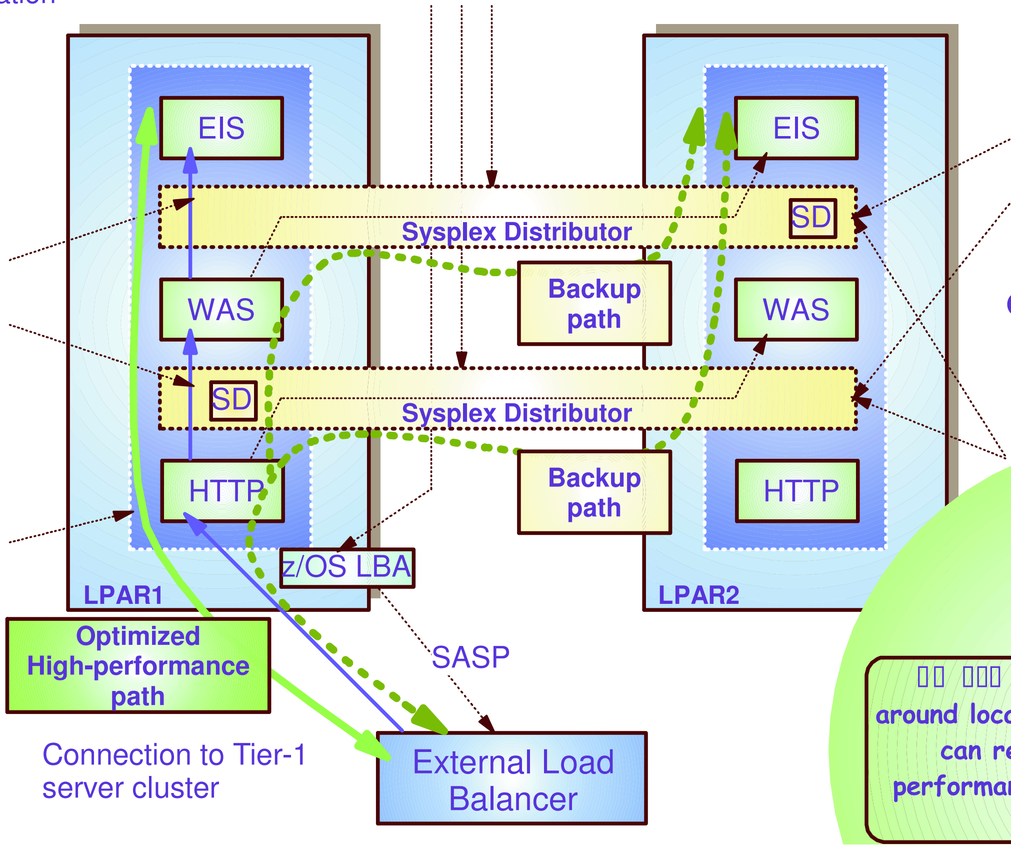
- Avoid authentication overhead
- Avoid data conversions

1. WLM LPAR and server-specific performance weights
2. TCP/IP stack server-specific health weights

Level of **local favoritism** can be configured

- Always choose local target if target is available and healthy
- Control level of WLM weight impact on target selection

Fast direct local sockets path inside the same "tower" (inside the same TCP/IP stack)



Server instances within same "tower" are **preferred targets** for Sysplex Distributor

Connection to Tier-1 server cluster

Optimized traffic flow:

- "Distributed" logic in target stack avoids cross-LPAR flows to SD for connection setup when local target is chosen - **configured**
- Avoids traffic routing via SD-owning LPAR to local targets - **automatic**

□□ □□□ □ cannot solve all t□e issues around local vs. remote connectors, but it can reduce t□e network-related performance penalty w□en usin□ remote connectors.

Usage and Invocation

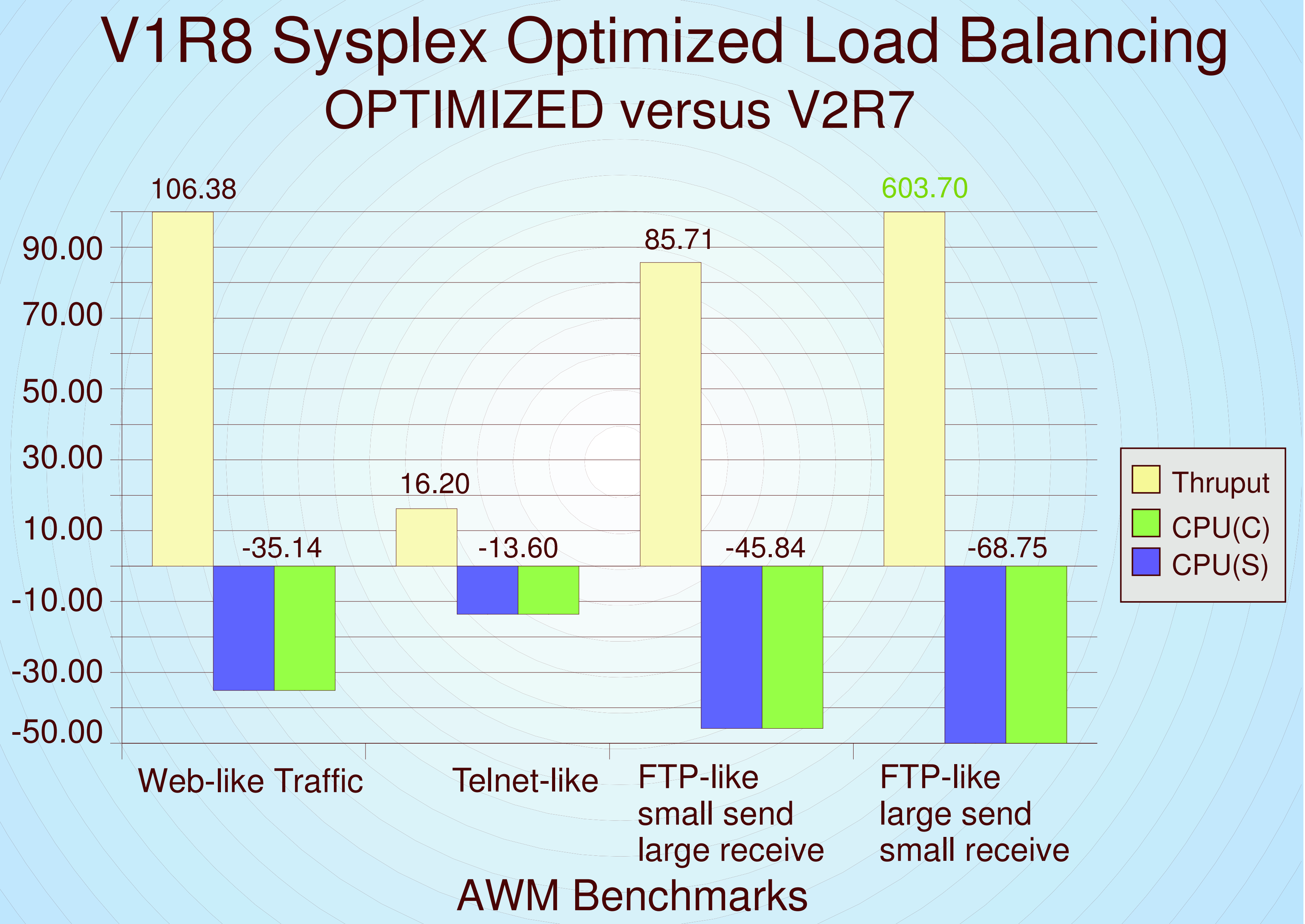
This function is controlled by a new keyword, OPTLOCAL, on the VIPADISTRIBUTE configuration statement.

- **OPTLOCAL 0**: the connection should always remain local.
- **OPTLOCAL 1**: the connection should remain local unless the server's WLM weight is zero.
- **OPTLOCAL values 2-16** are used as multipliers to increase the local server's WLM weight
  to favor the local stack.

Regardless of the value specified, the connection will always be sent to the distributor if any of the following are true:

- No server application is available on the local stack
- Server Efficiency Fraction (SEF) value on the local stack is less than 75
- The health indicator for the local stack is less than 75
- The abnormal transactions count for the local stack is greater than 250

Benchmark Comparison

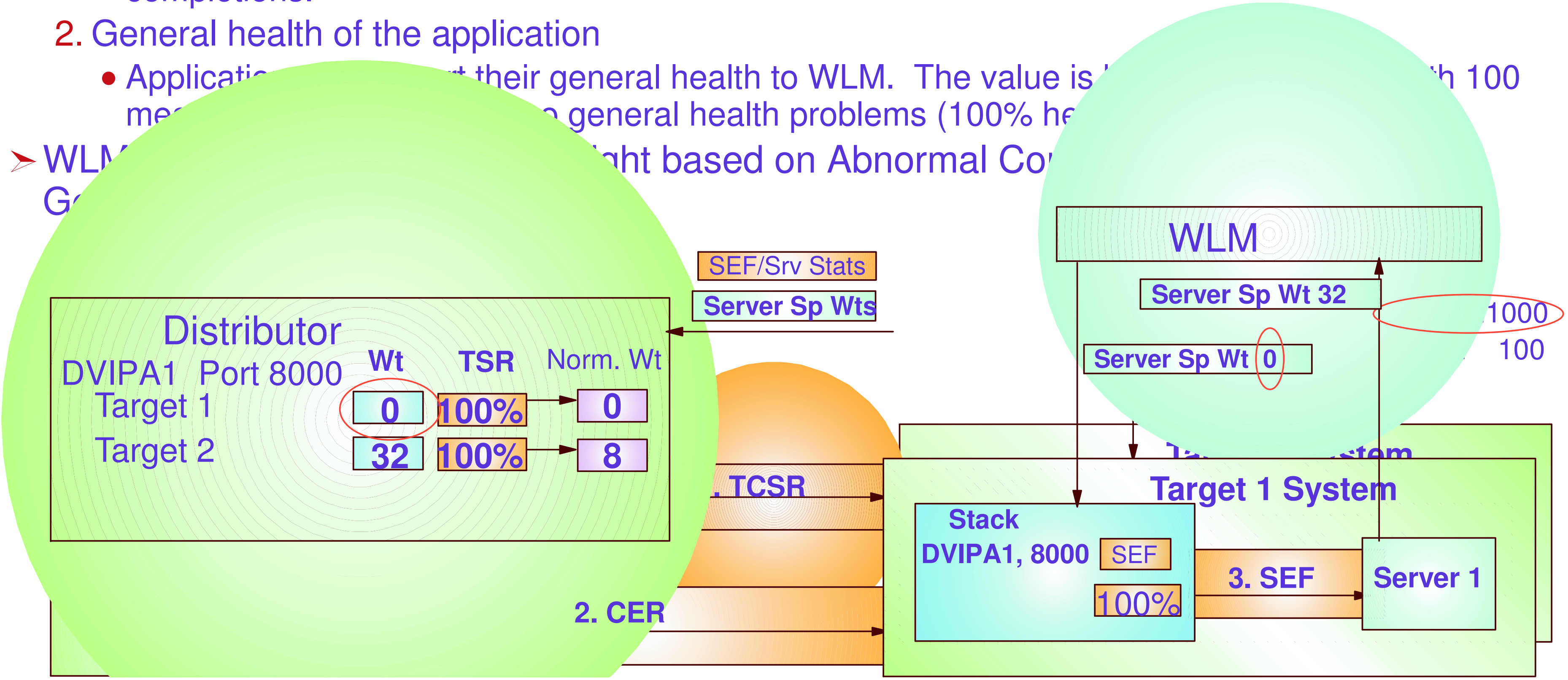


Storm Drain Problem

- WLM is not aware of all problems experienced by load balancing targets:
  - The server application needs a resource such as a database, but the resource is unavailable
  - The server application is failing most of the transactions routed to it because of internal processing problems
  - The server application acts as a transaction router for other back-end applications on other system(s), but the path to the back-end application is unavailable
- In each of these scenarios, the server appears to be completing the transactions quickly (using little CPU capacity) when they are actually being failed
- This is the Storm Drain Problem
  - The server is favored by WLM since it is using very little CPU capacity
  - As workloads increase, the server is favored more and more over other servers
  - All this work goes "down the drain"

Solution

- WLM provides an interface which allows a server to pass additional information about its overall health:
  1. Abnormal transaction completion Rate
     - Applications such as the CICS Transaction Server for z/OS, that act as Subsystem Work Managers, can report an abnormal transaction completion rate to WLM (abnormal completions per 1000 transactions). The value is between 0 and 1000 with 0 meaning no abnormal completions.
  2. General health of the application
     - Applicati t their general health to WLM. The value is h 100 me o general health problems (100% he
- WL ight based on Abnormal Co G



WLM
Target Application Awareness Improvements

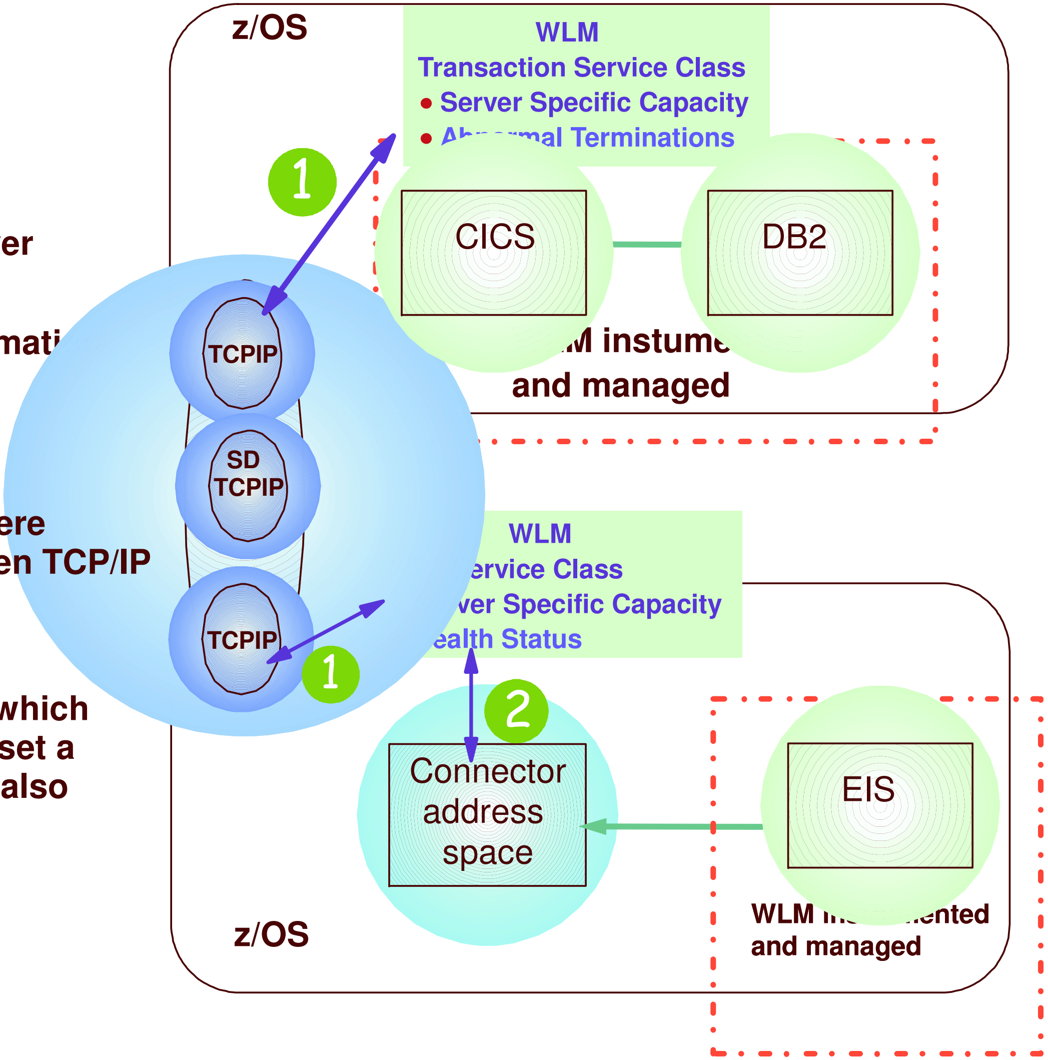
Server Scenarios

1 **IWM4SRSC**

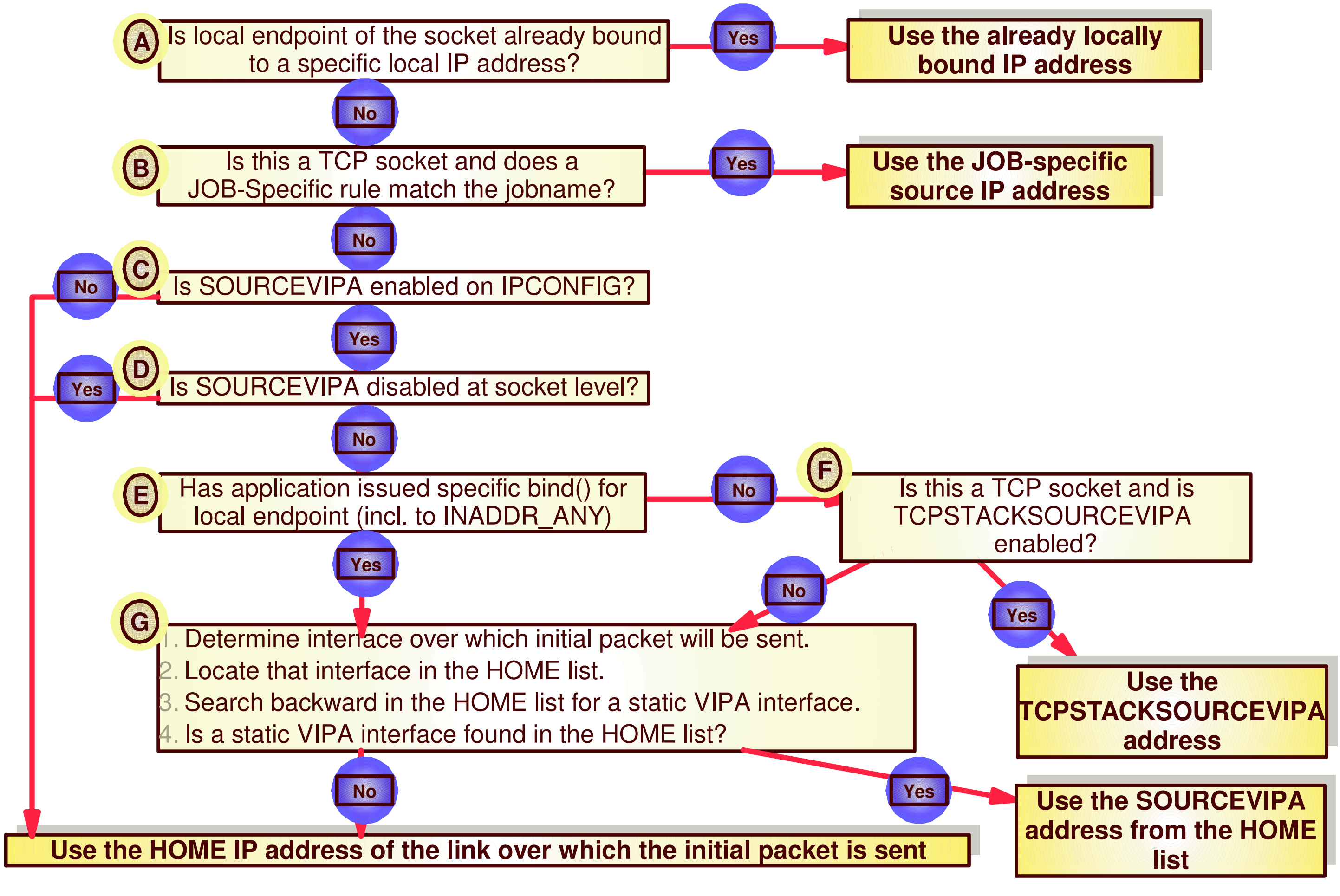
- Used by SD to retrieve Server Specific Information
- Abnormal Termination informati solves the case where the registered server is not the transaction consumer
- Does not solve the case where another connector is between TCP/IP and the consumer:

2 **IWM4HLTH**

Allows address spaces which are not instrumented to set a health status which is also returned by IWM4SRSC



Selecting source IP address for outbound IPv4 connections or associations in CS z/OS V1R6



A Is local endpoint of the socket already bound to a specific local IP address?

Yes → Use the already locally bound IP address

No →

B Is this a TCP socket and does a JOB-Specific rule match the jobname?

Yes → Use the JOB-specific source IP address

No →

C Is SOURCEVIPA enabled on IPCONFIG?

No → Use the HOME IP address of the link over which the initial packet is sent

Yes →

D Is SOURCEVIPA disabled at socket level?

Yes → Use the HOME IP address of the link over which the initial packet is sent

No →

E Has application issued specific bind() for local endpoint (incl. to INADDR_ANY)

No → F Is this a TCP socket and is TCPSTACKSOURCEVIPA enabled?

- Yes → Use the TCPSTACKSOURCEVIPA address
- No → G

Yes → G

G
1. Determine interface over which initial packet will be sent.
2. Locate that interface in the HOME list.
3. Search backward in the HOME list for a static VIPA interface.
4. Is a static VIPA interface found in the HOME list?

Yes → Use the SOURCEVIPA address from the HOME list

No → Use the HOME IP address of the link over which the initial packet is sent

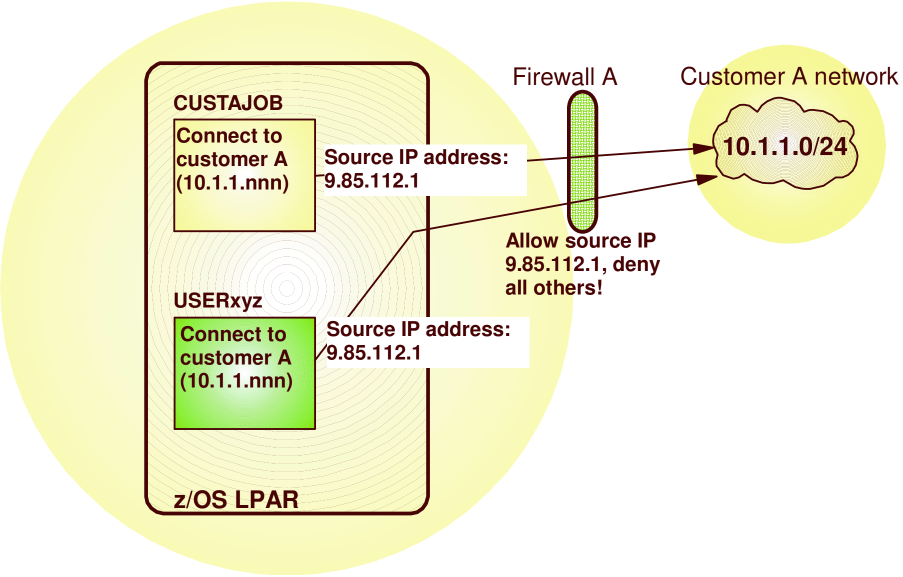
Control over Source IP address for outbound connections from z/OS

Extending configuration control over which local IP address to use for outbound connections from z/OS

✓ Communications Server Introduced Job-specific Source IP Addressing in z/OS V1R6

- A new TCPIP Profile statement SRCIP/ENDSRCIP allows the selection of a source IP address for outbound TCP connections by job name
- Overrides TCPSTACKSOURCEVIPA and SOURCEVIPA specifications
- Helps in distributed DVIPA scenario

Destination-based source IP address selection



Firewall A
Customer A network
CUSTAJOB
Connect to customer A (10.1.1.nnn)
Source IP address: 9.85.112.1
10.1.1.0/24
Allow source IP 9.85.112.1, deny all others!
USERxyz
Connect to customer A (10.1.1.nnn)
Source IP address: 9.85.112.1
z/OS LPAR

z/OS V1R8 introduces Destination-based source IP address selection

- Extends the SRCIP/ENDSRCIP block with destination IP address-based rules
- The source IP address used by a DESTIP rule cannot be a distributed DVIPA
- Useful if jobnames are unpredictable or if the same jobname establishes connections to multiple partner companies

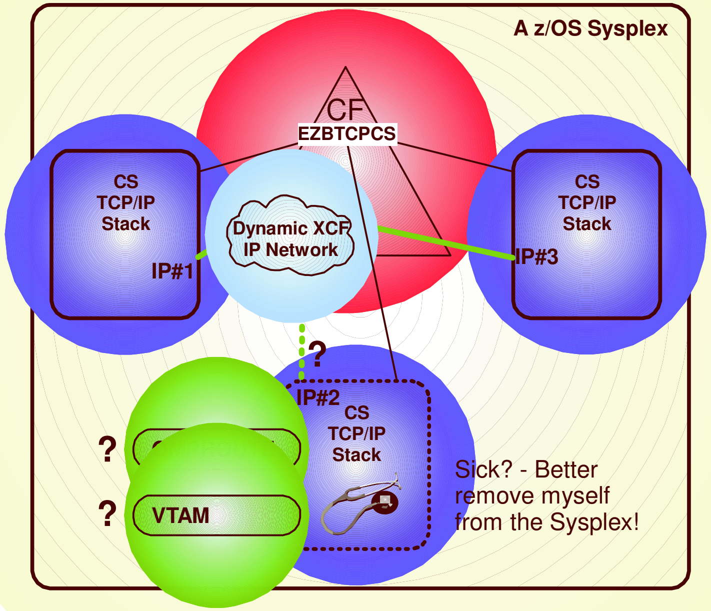
Usage and Invocation

```
   SRCIP
        JOBNAME      CUSTAJOB   9.85.112.1
        JOBNAME      CUSTBJOB   9.85.113.1
        JOBNAME      User1*     888:555::222
        DESTINATION 10.1.1.0/24 9.85.112.1
        DESTINATION 2001:0DB8:12::/64 2001:0DB8:99::2:2
   ENDSRCIP
```

- This example tells z/OS Communcations Server to use a source address of 9.85.112.1 for any sends to the 10.1.1.0/24 subnet
- IPv6 support is also shown with similar syntax.
- The priority for using these statments to assign source IP addresses is:
  1. jobname that is not a full wildcard (*)
  2. destination IP address
  3. full wildcard jobname

1.6-1.7

TCP/IP Sysplex autonomics reacts to and recovers dynamically from a range of error conditions

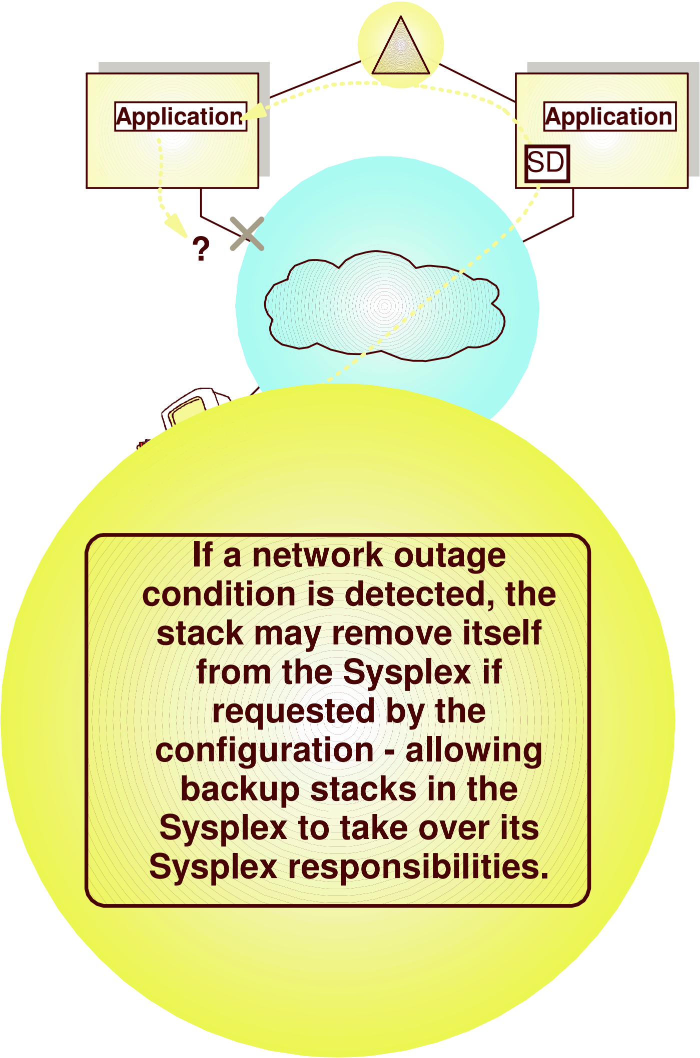


Sick? - Better remove myself from the Sysplex!

Monitoring is always done, but configuration controls in the TCPIP Profile determine if the TCPIP stack will remove itself from the sysplex.

- Autonomic functions to reduce single point of failure for distributed applications in a sysplex
  - Monitor CS health indicators
    - Storage usage - CSM, TCPIP Private & ECSA
  - Monitor dependent networking functions
    - OMPROUTE availability
    - VTAM availability
    - XCF links available
  - Monitor Communications Server component-specific functions
- Monitors determine if this TCPIP stack will remove itself from the sysplex and allow a healthy backup to take ownership of the sysplex duties (own DVIPAs, distribute workload)

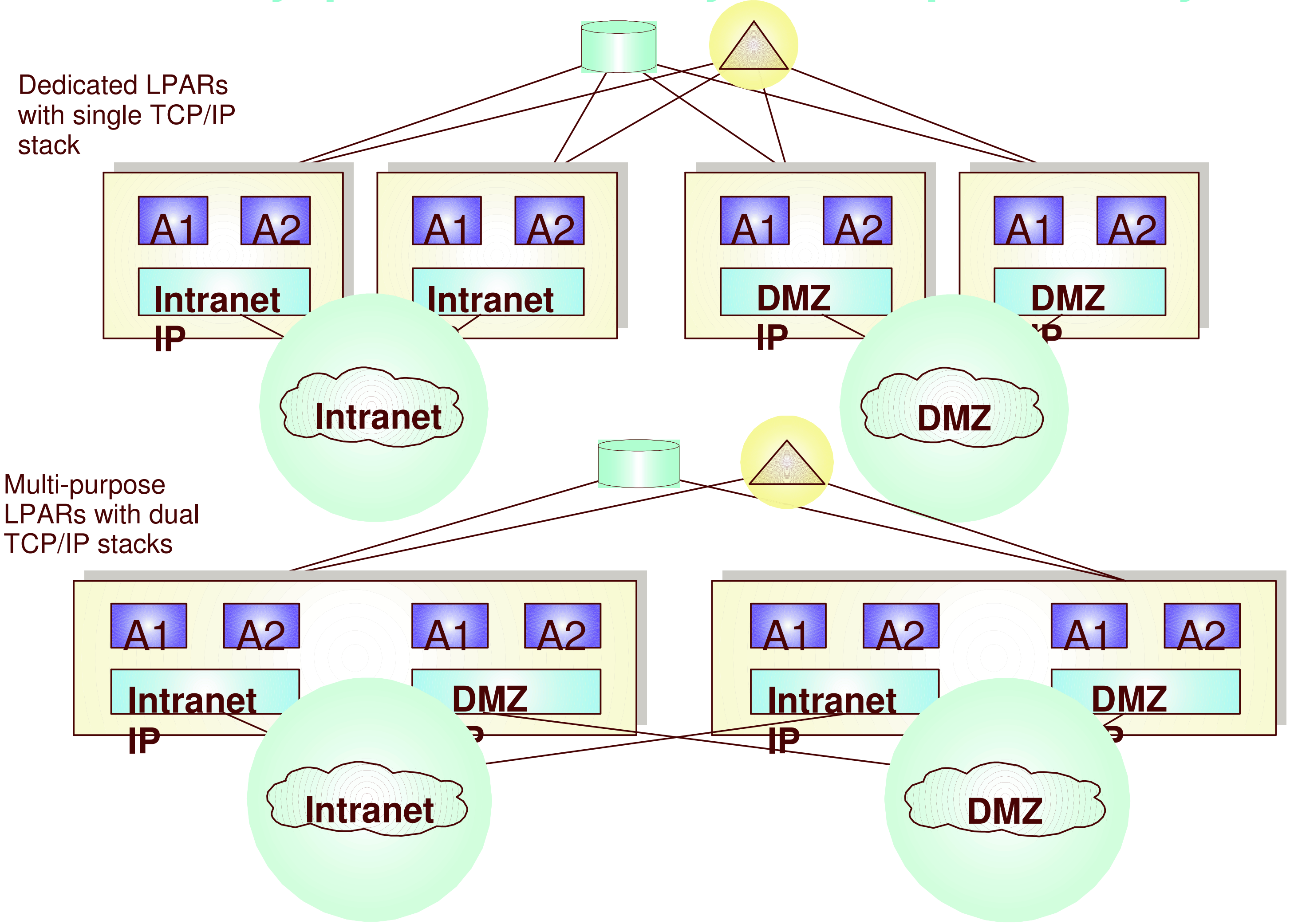
TCP/IP Sysplex autonomics adds automated recovery from network outage conditions



If a network outage condition is detected, the stack may remove itself from the Sysplex if requested by the configuration - allowing backup stacks in the Sysplex to take over its Sysplex responsibilities.

- **Network outage detection added to the Sysplex autonomics of TCP/IP**
  - Specify which network interfaces to be monitored
  - Monitor network interface itself (active or inactive)
    - To detect interface hardware issues
  - If dynamic routing is used, optionally monitor if dynamic routes exist over the interface
    - To detect first-hop router issues
  - DELAYJOIN extended to monitor for interfaces up and dynamic routes detected

z/OS Sysplex connectivity to multiple security areas



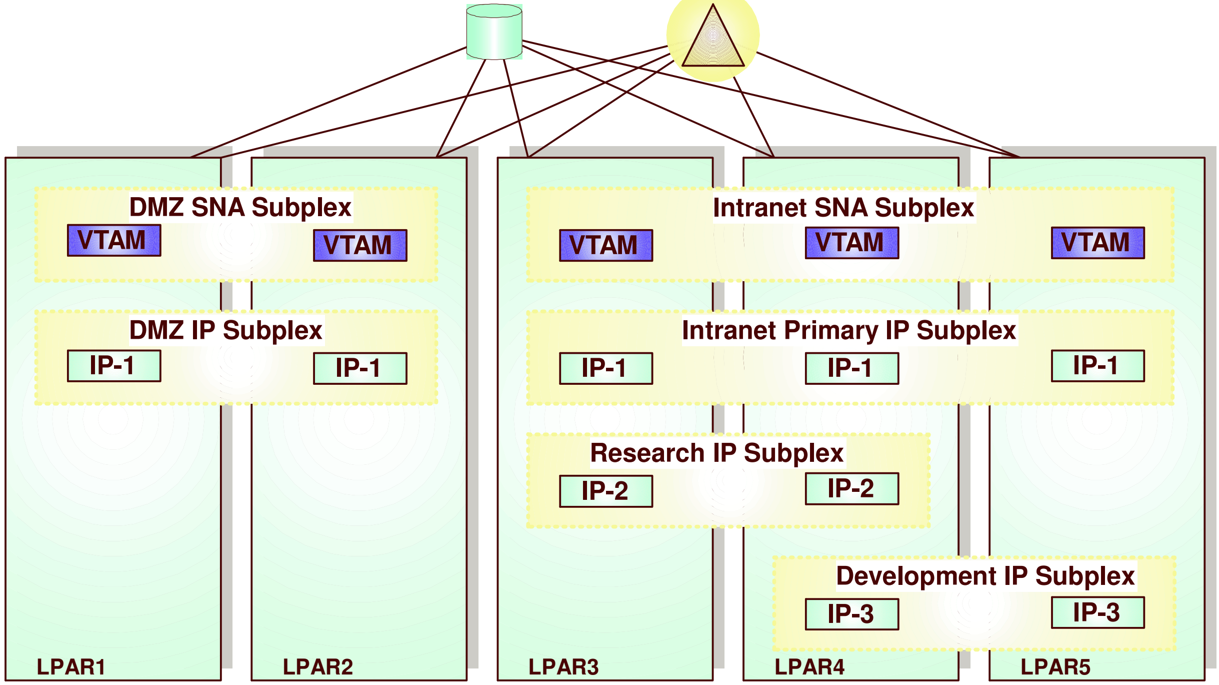
z/OS Sysplex connectivity to multiple security areas, challenges

- **How to control level of automatic connectivity**
  - XCF signalling (group name) - both IP and SNA
  - IUTSAMEH (same host IP links inside an LPAR)
  - HiperSockets (as enabled via IQDCHPID in VTAM)
- **How to control level of IP and SNA resource awareness**
  - Dynamic IP address discovery across the Sysplex
  - VTAM generic resource and MNPS resource scope spans the full Sysplex
- **How to control scope of IP workload balancing using Sysplex Distributor?**
  - SD requires Dynamic XCF to be enabled, and Dynamic XCF will establish automatic IP connectivity to all stacks in the Sysplex that also have Dynamic XCF enabled

z/OS Sysplex connectivity to multiple security areas, unacceptable solution

To support environments such as these, installations typically end up implementing complex resource controls and disabling many of the dynamic networking functions that are provided by TCP/IP and VTAM.

Enable use of networking Sysplex functions in a Sysplex that is connected to multiple security areas



- One SNA subplex per LPAR
- A TCP subplex cannot span multiple SNA subplexes

- Different IP stacks in an LPAR may belong to different IP subplexes
- Standard RACF controls for stack access and application access to z/OS resources need to be in place.

Subplexing scope

- **Networking subplex scope:**
  - VTAM Generic Resources (GR) and Multi-Node Persistent Session (MNPS) resources
  - Automatic connectivity - IP connectivity and VTAM connectivity over XCF (including dynamic IUTSAMEH and dynamic HiperSockets based on Dynamic XCF for IP)
  - IP stack IP address (including dynamic VIPA) awareness and visibility
  - Dynamic VIPA movement candidates
  - Sysplex Distributor target candidates

Subplex Configuration VTAM

- New VTAM Start Option:
- XCFGRPID vv where vv is a number between 2 and 31
- VTAM joins ISTXCFvv and ISTCFSvv Sysplex groups
- STRGR and STRMNPS CF structure names are suffixed with vv

For example, if STRMNPS=ISTMYMNPS is specified, VTAM connects to ISTMYMNPSvv

Subplex Configuration TCP/IP

- New TCP Profile parameters:
  - GLOBALCONFIG statement:
    - **XCFGRPID tt - used to partition the TCP/IP sysplex groups into subplexes**
      - tt is a numeric value between 2 and 31
    - **IQDVLANID nn - used to partition HiperSockets for Dynamic XCF connectivity into subplexes**
      - nn is a numeric value between 1 and 4094
      - IQDVLANID support for HiperSockets requires a z890 GA2 or z990 GA2 hardware level.
    - **These values cannot be modified through Vary Obeyfile processing**
- TCP will join Sysplex group EZBTvvtt, , where vv is the VTAM subplex number mentioned earlier
- SWSA and Sysplexports structure names will be suffixed by vvtt
  - EZBDVIPAvvtt and EZBEPORTvvtt
  - For example, if the TCP/IP GLOBALCONFIG specified an XCFGRPID of 05 and the supporting VTAM was started with XCFGRPID=23, this stack would connect to EZBEPORT2305

Hipersockets VLANs

Hipersockets LANs can now be partitioned into VLANs

- If multiple TCP/IP subplexes in an LPAR will be connected to Hipersockets, they need to be on different dynamic Hipersockets VLANs as well.
  - requires configuration to partition
- VLAN supported for both dynamic and manual Hipersockets.
- requires a z890 GA2 or z990 GA2 hardware level.

DNS/WLM - going away or not going away or what ?

- **However, the dynamic name registration capabilities of DNS/WLM are still very useful from an availability perspective and are not replaced by any of the currently available alternative load balancing technologies:**
  - Dynamic registration of individual application instances when they start up
  - Dynamic registration of groups of application instances when they start up
  - Dynamic registration of TCP/IP stacks when they start up

- **General dynamic registration in modern DNS servers (BIND 8 or later) is supported by a set of DNS protocols that are known as Dynamic DNS (DDNS)**
  - CS z/OS V1R8 implements a new infrastructure that will support DDNS registration of the same type of entries that were supported by DNS/WLM
  - DDNS is a standard protocol
  - Any DDNS capable name server can be the target of the DDNS registrations

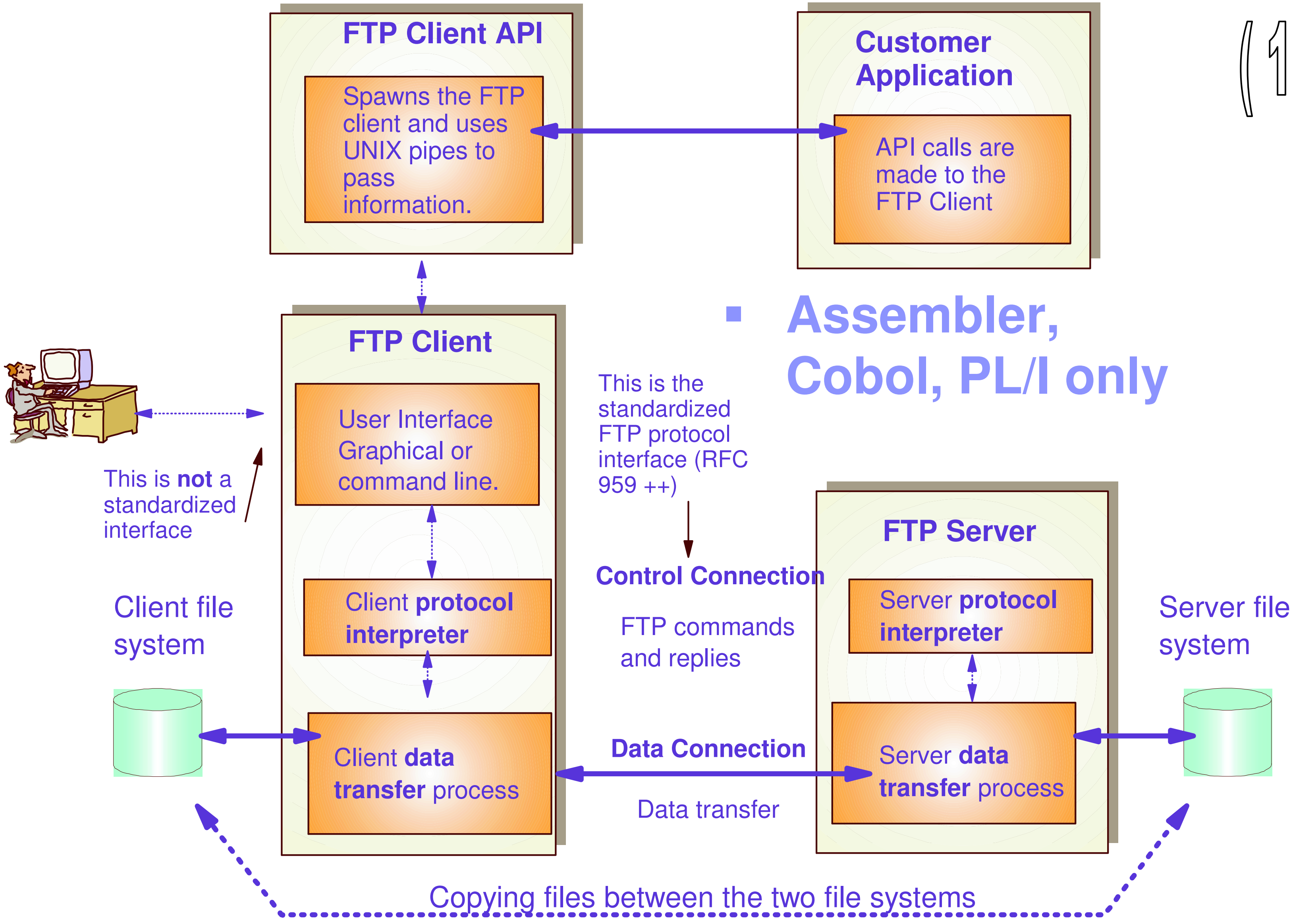
1. Sysplex Enhancements
2. Application Enhancements
3. Enterprise Extender and SNA Enhancements
4. IPv6 on z/OS Communications Server
5. Security

Application enhancements

- **Focus areas:**
  - FTP
  - TN3270

TCP/IP Application Interfaces - FTP Client API

(1.6)



FTP Client API

Spawns the FTP client and uses UNIX pipes to pass information.

Customer Application

API calls are made to the FTP Client

FTP Client

User Interface Graphical or command line.

This is **not** a standardized interface

Client **protocol interpreter**

Client **data transfer** process

Client file system

This is the standardized FTP protocol interface (RFC 959 ++)

Control Connection

FTP commands and replies

Data Connection

Data transfer

FTP Server

Server **protocol interpreter**

Server **data transfer** process

Server file system

Copying files between the two file systems

- **Assembler, Cobol, PL/I only**

TCP/IP Applications – FTP Enhancements FTP Client API Support for C/C++

- **Extend the FTP Client API to C/C++**
  - Popular, larger level audience
- **C header file (FTPCAPI.H) provided**
  - Inline static functions to facilitate calling the FTP Client API
    - FAPI_INIT initializes the interface
    - FAPI_SCMD sends an FTP subcommand
    - FAPI_POLL checks status of an outstanding subcommand
    - FAPI_GETL_COPY retrieves output related to a subcommand and copies to a user buffer
    - FAPI_GETL_FIND retrieves output related to a subcommand and searches for a line of a specific type of output
    - FAPI_TERM ends the interface.
  - Associated constants and control blocks
- **C Sample provided**

FTP client API in REXX

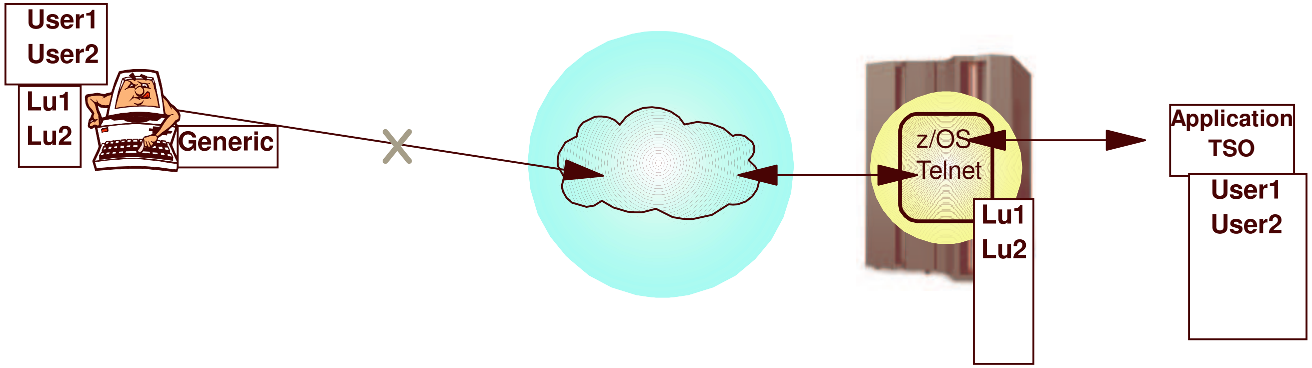
- z/OS V1R8 further
  extends the FTP
  client prog
  interfac
  provi
  API

```
/* Create FTP client control information    */
if ftp('create','fcai.', TRACEID) < 0 then do
   Say 'Unable to create the FCAI'
   exit
end
/* Enable trace                             */
if ftp('fcai.', 'set_trace', 'ON') < 0 then do
   call ftp_error 'fcai.'
end
/* Open a connection                        */
if ftp('fcai.', 'init', OPENSTRING, VAR1, VAR2) then do
   call ftp_error 'fcai.'
end
/* Send USER command                        */
if ftp('fcai.', 'scmd', USER_COMMAND, 'W') < 0 then do
   call ftp_error 'fcai.'
end
/* Send password                            */
if ftp('fcai.', 'scmd', PASS_COMMAND, 'W') < 0 then do
   call ftp_error 'fcai.'
end
```

Signifcantly improved automation capabilities for file transfer operations that are initiated on z/OS

Improved TN3270 recovery when a client is running multiple sessions

- If the z/OS CS Telnet server receives a new connection from a client IP address that already has one or more existing connections, the server will "poke" the existing connections to make sure they are still up.
  - If not, they will be cleaned up immediately
  - This improves the case where a client has telnet sessions which go down, so he starts a new session and reconnects.
    - helps avoid the "connect connect, already connected" error scenario



Misc. TN3270 enhancements

- Support for MVS system symbolics in the USS message table
  - for example, would enable the USS logon screen to report which LPAR is serving the client.
  - in addition to the USS symbolics that already exist

- Allow the LU Exit to assign the USS table and/or Interpret table names
  - LU Exit assigned name will override tables assigned by LUMAP statements
  - Only supported on TN3270E connections

More Misc. TN3270 enhancements

- Queued Session Timer
  - allows TN3270 to redrive setup and free up the session if a session manager does not bind within a set time of the previous session's unbind
  - eliminates need for user to disconnect/reconnect in some error cases

- Support removed for obsolete statements:
  - QUEUESESSION statement no longer supported
    - Use QSESSion parameter on the RESTRICTAPPL or ALLOWAPPL statement instead
  - LUSESSIONPEND, MSG07, TELNETDEVICE statements no longer supported in the BEGINVTAM block
    - Code statement in TelnetGlobals, TelnetParms, or ParmsGroup instead

TN3270 response time monitor results via SMF recording

These statistics were added in V1R5:

Life-of-connection data for life-of-connection averages

- Transaction count
- Round trip & IP response time totals
- Averages for round trip, IP, and SNA response times

Life-of-SNA session data for life-of-SNA session averages *(added in z/OS V1R8)*

- Transaction count
- Round trip & IP response time totals
- Averages for round trip, IP, and SNA response times

Sliding window data for sliding window averages

- Period transaction count
- Period round trip & IP response time totals
- Sliding window transaction count
- Sliding window round trip & IP response time totals

Sum of squares for variance and standard deviation

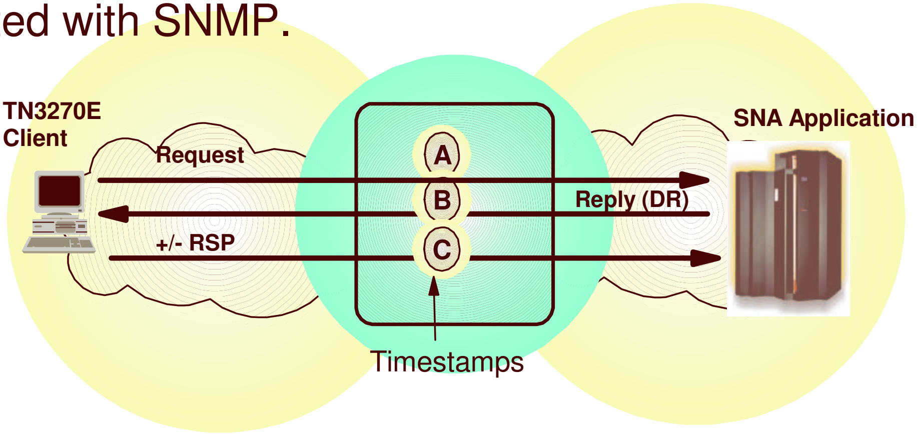
- Round trip, IP, and SNA sum of squares

Round trip response time counts by time bucket

TN3270 performance data collection improvement using NMI

You can now use the Network Management Interface (EZBNMIFR callable API) to collect TN3270 performance data.

- Bypasses SNMP and calls Telnet directly.
- Avoids need to filter out non-TN3270 connections.
- Returns all data in a single large data block instead of returning data for each connection.
- The same data is reported in the EZBNMIFR quadruplet as is reported with SNMP.



1. Sysplex Enhancements
2. Application Enhancements
3. Enterprise Extender and SNA Enhancements
4. IPv6 on z/OS Communications Server
5. Security

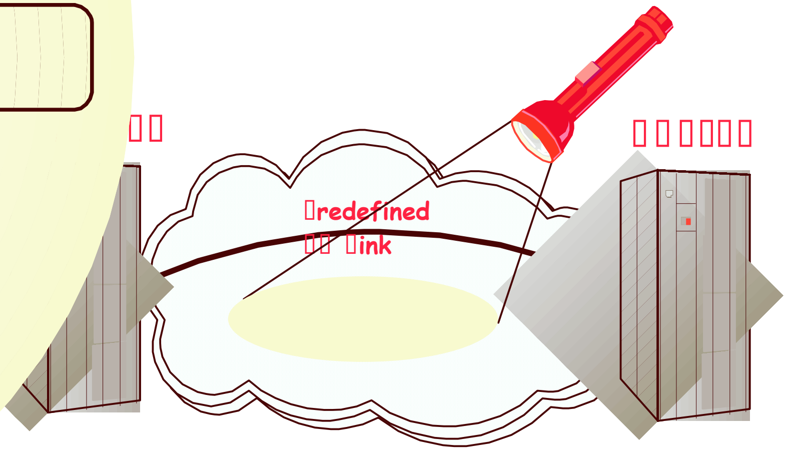
Enterprise Extender and SNA Enhancements

- **Focus areas:**
  - Enhanced operations
  - Enhanced configuration control

Enterprise Extender connectivity test

- The Enterprise Extender connectivity test command is useful in debugging various network problems. This command can be used to test an existing Enterprise Extender connection, or it can be used to assist in diagnosing why an EE connection cannot be established.
- It provides an end-to-end connectivity test and diagnosis

```
D NET,EEDIAG,TEST=YES,IPADDR=(9.67.1.1,9.67.1.6)
```



Removal of AnyNet

- **Enterprise Extender, TN3270, and distributed Communications Server Remote API functions are the strategic protocols for SNA/IP integration**
  - AnyNet has not been enhanced in years

- **As of z/OS V1R8, AnyNet will no longer be included as a component of Communications Server**

Dynamic Update of VTAM application major nodes

- This function provides the ability to modify the Application major node by allowing the UPDATE operand on the vary activate command. This change allows:
  - Adding APPL resources
    - specify "update=add" to add resources
  - Deleting APPL resources
    - specification of "update=all" is required to delete APPL resources
  - Changing APPL resources
    - specification of "update=all" is required to change existing APPL values

1. Sysplex Enhancements
2. Application Enhancements
3. Enterprise Extender and SNA Enhancements
4. IPv6 on z/OS Communications Server
5. Security

IPv6 on z/OS Communications Server

- **Focus areas:**
  - IPv6 network management
  - IPsec

The Journey to IPv6 for z/OS Communications Server

The first phase (z/OS V1R4)

- Stack support for IPv6 base functions - (APIs, Protocol layers)
- Resolver
- High speed attach (OSA Express QDIO))
- Service tools (Trace, Dump, etc.)
- Configuration and netstat, ping, traceroute, SMF
- Static Routing
- FTP, otelnetd,unix rexec, unix rshd/rexecd

The second phase (z/OS V1R5)

- Network Management
  - Applications and DPI
  - Version-neutral TCP/IP Standard MIBs
  - Additional SMF records
- Applications/Clients/APIs
  - Tn3270 server,CICS Sockets, sendmail,ntp,dcas, rxserve,rsh client
- Enterprise Extender
- Point to Point - type DLCS
- Dynamic Routing Protocol w/ OMPROUTE (only RIPng)

The third phase (z/OS V1R6)

- Sysplex Exploitation (Dynamic VIPA, Sysplex Distributor functions)
- Dynamic Routing Protocol w/ OMPROUTE (OSPFv3)
- Additonal Network Management MIBs

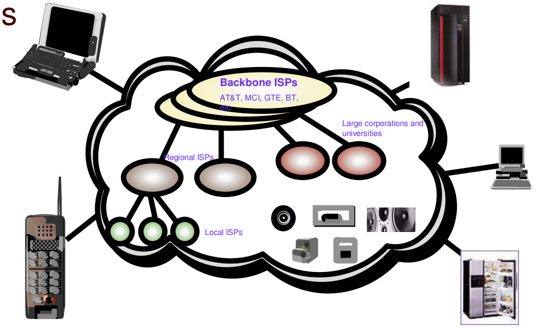
The fourth phase (z/OS V1R7)

- SNMP UDP standard MIB (RFC2013) and IBM MVS TCP/IP Enterprise-specific MIB for UDP
- Advanced Socket API support - RFC3542
- IPv6 Two Default Routers - required for IPv6 compliance
- HiperSockets DLC

After z/OS V1R7

- Integrated IPSec
- Complete Advanced Socket APIs
- Extended Stats MIB, OSPFv3 MIB
- Intrusion Detection Services
- IPv6 mobility support

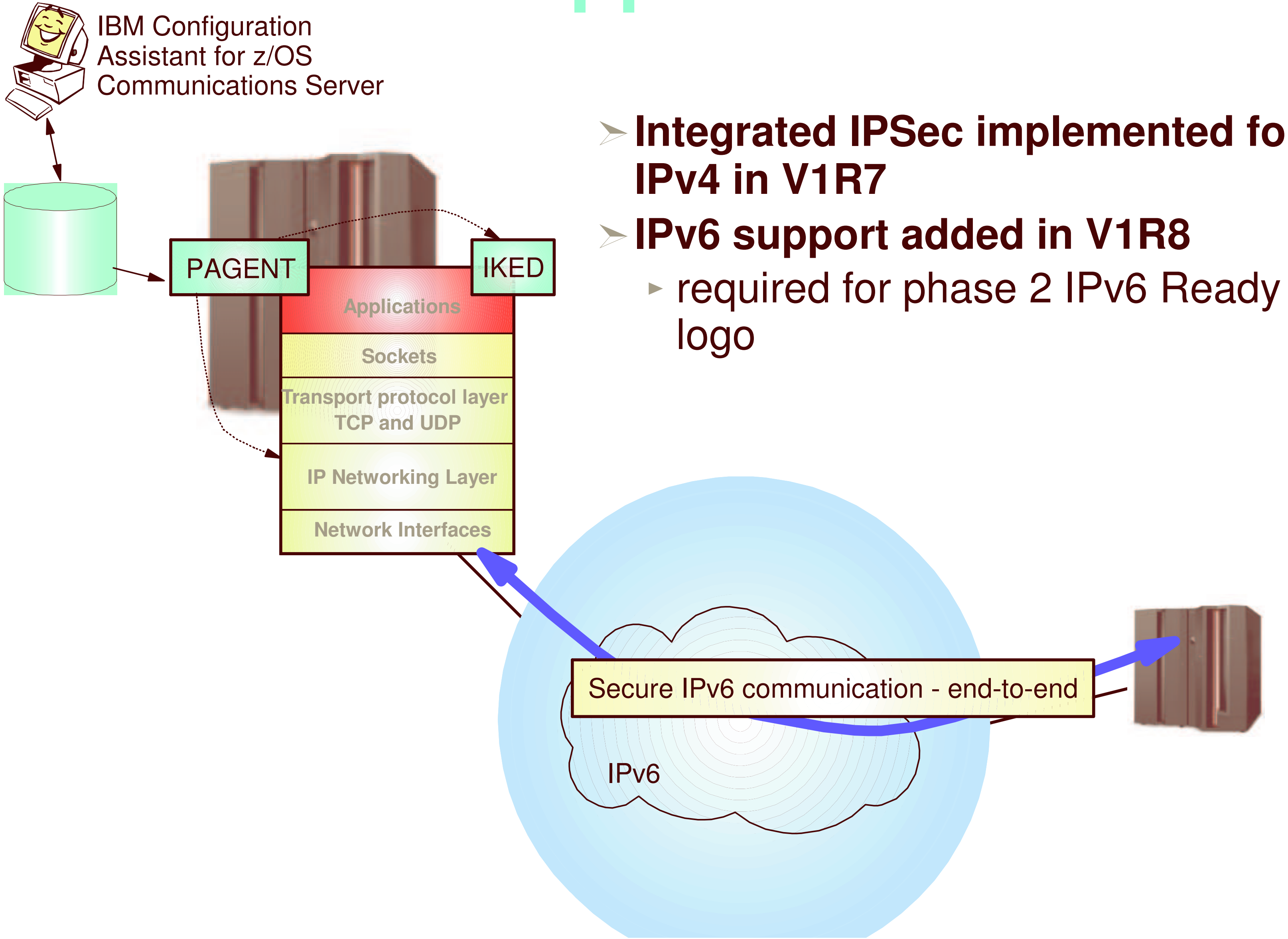
The Internet - a worldwide digital utility.



Connectivity for ***anyone*** from ***anywhere*** (car, home, office) to ***anything***!

Objective is to have IPv6 production ready on the platform when you need it!

IPv6 Support for IPSec



- ➢ **Integrated IPSec implemented for IPv4 in V1R7**
- ➢ **IPv6 support added in V1R8**
  - required for phase 2 IPv6 Ready logo

Misc IPV6 enhancements

- IPv6 Fast Response Cache Accelerator support

- IPv6 support for RPC

- RPCBIND is a new server in z/OS V1R8
  RPCBIND supports RFC 1833
  udp, udp6, tcp and tcp6 transports only

- Allows NFS disks to be used in a IPv6 Network

- Has improved Reliability, Availability, and Servicability over PORTMAP

- No application change required to move to RPCBIN

1. Sysplex Enhancements
2. Application Enhancements
3. Enterprise Extender and SNA Enhancements
4. IPv6 on z/OS Communications Server
5. Security

Security

- **Focus areas:**
  - Application-transparent IP security technologies
    - IPSec
    - Application-transparent TLS (AT-TLS)
    - Configuration assistant

Universal Access to Business Data Without Universal Exposure

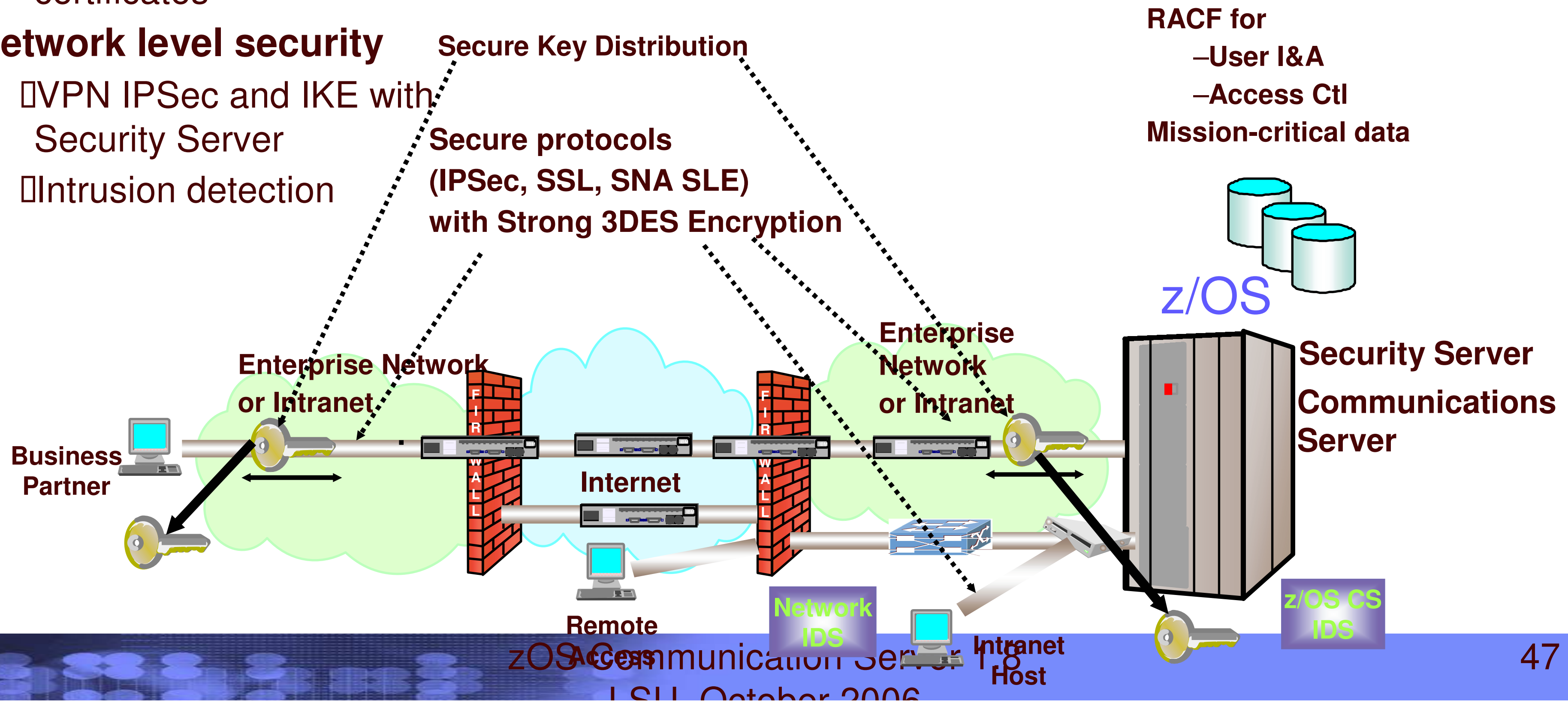
Application level security

- Base is platform security
- SSL for TN3270E and Web
- SSL for FTP (z/OS V1R2)
- Kerberos (z/OS V1R2)
- Client authentication based on digital certificates

- Secure access to both TCP/IP and SNA applications
- Focus on end-to-end security and self-protection
- Exploits strengths of S/390 and z900 HW and SW

Network level security

- VPN IPSec and IKE with Security Server
- Intrusion detection



Security simplification 1.6->1.7->1.8

1.6

FUNC	IMPL	TOOL	STORE
IPSEC + Filtering	Firewall functions	GUI AIX/Linux	Files
QOS	Pagent	QOS tool windows	LDAP (or files)
IDS	Pagent	IDS tool windows	LDAP
SSL/TLS	TN3270 FTP WAS ...	Misc	Misc

1.7

IPsec + filtering + SSL/TLS	Pagent	Conf Assistant (Windows)	Files
QOS	Pagent	QOS GUI (windows)	LDAP (or files)
IDS	Pagent	IDS GUI (windows)	LDAP

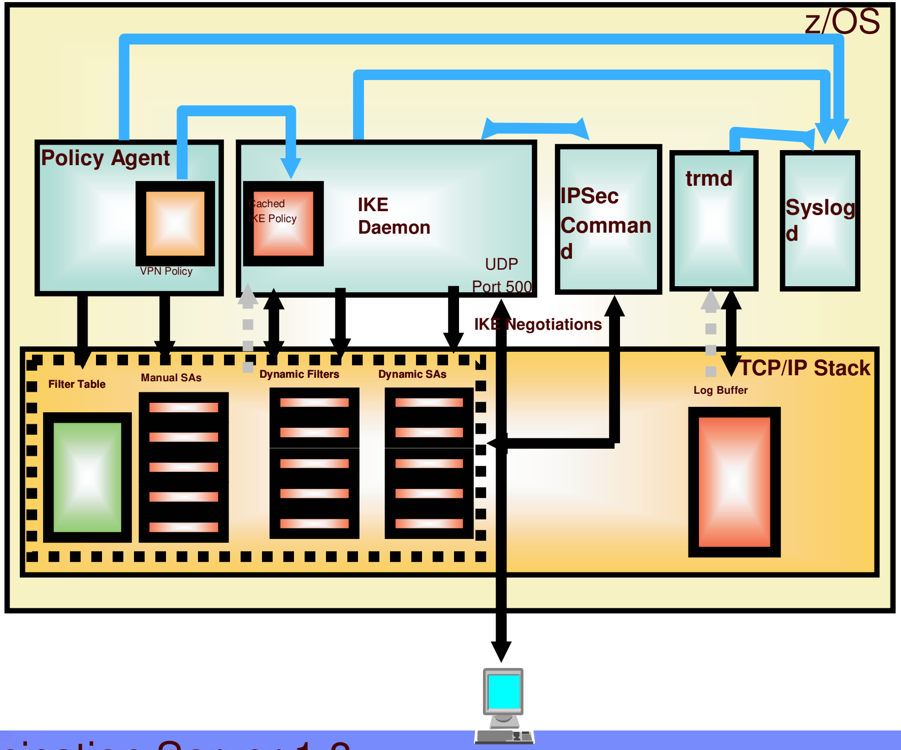
1.8

All	Pagent	Config Ass (windows)	Flat files

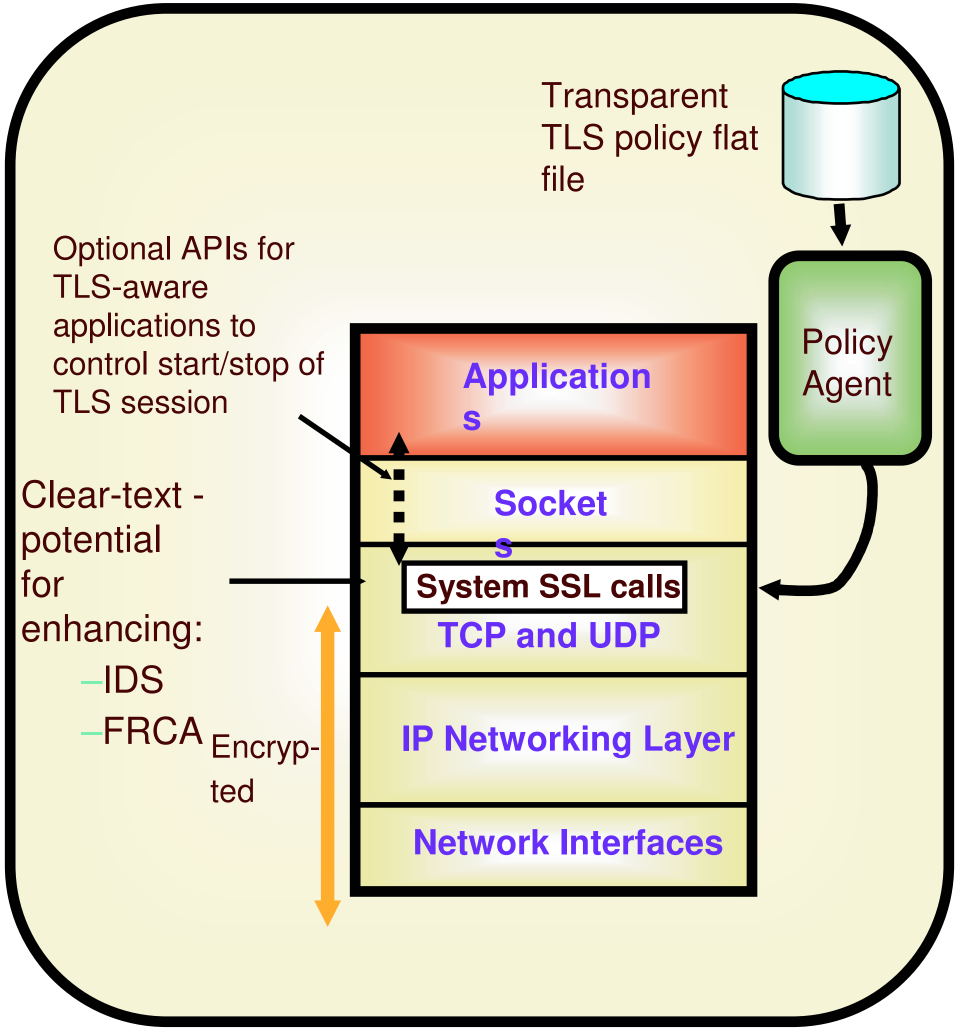
Integrated IPSec/VPN support including NAT traversal - ease of use and performance

- Features
  - Configuration support
    - Optimized for z/OS host-to-host and z/OS host-to-gateway (z/OS gateway still supported)
    - NAT traversal support
  - Simplified infrastructure
    - Eliminates need for FW technologies daemons
  - Simplified configuration
    - New configuration GUI for both new and expert users
    - Direct file edit into local configuration file
    - Reduced definition, more "wildcarding"
  - Improved serviceability
    - Improved messages and traces
  - Default filters part of TCP profile
    - More granular control before policy is loaded
  - Administrative controls
    - pasearch, new IPSec command

- Complete IPSec, filtering, and IKE solution part of z/OS Communications Server
  - Alternative to firewall technologies
    - New IKE daemon and configuration
- Makes use of existing Communications Server Infrastructure
  - TCP/IP stack - IPSec and IP filtering
  - Policy agent - reads and manages IPSec and IKE policy
  - trmd - monitors TCP/IP stacks for log messages

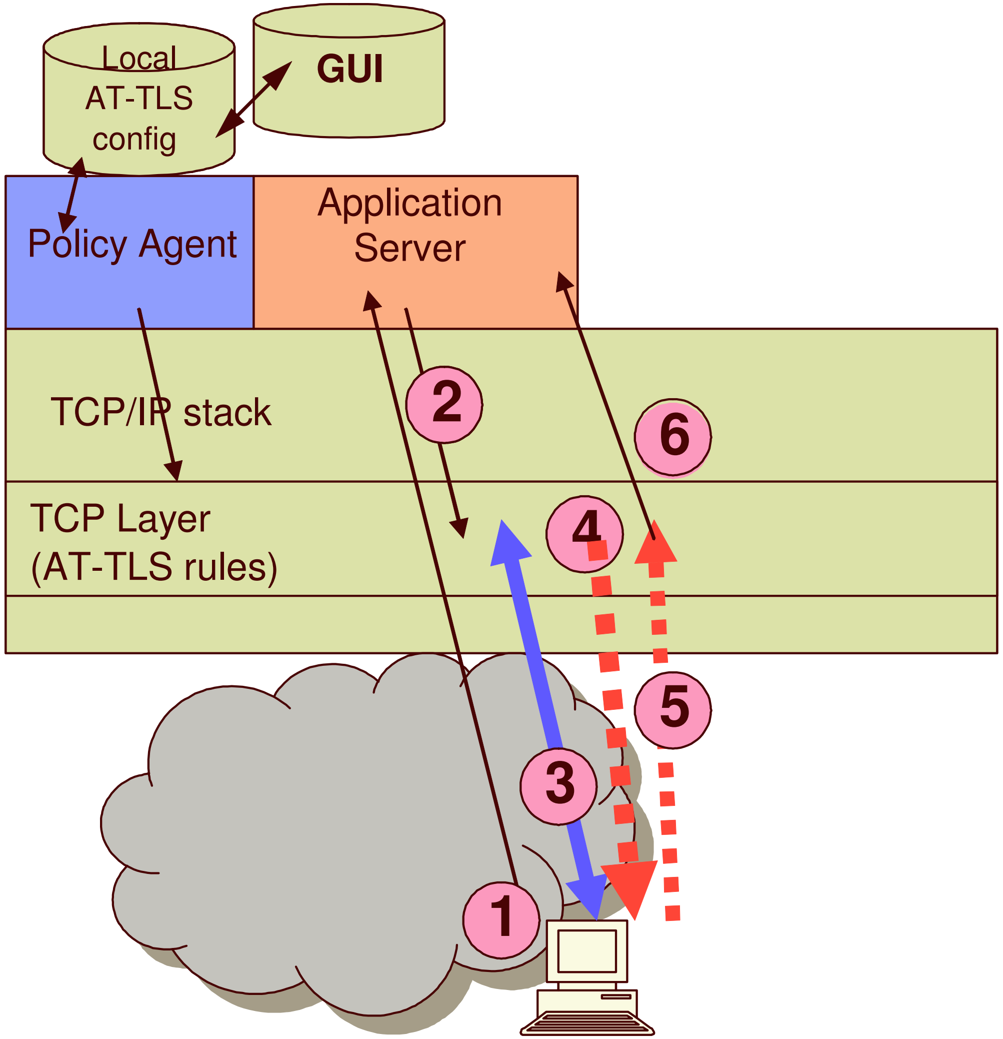


Transparent application security: policy-controlled transparent SSL/TLS support - SSL/TLS for all z/OS sockets applications


Transparent TLS policy flat file
Optional APIs for TLS-aware applications to control start/stop of TLS session
Policy Agent
Applications
Sockets
System SSL calls
TCP and UDP
IP Networking Layer
Network Interfaces
Clear-text - potential for enhancing:
–IDS
–FRCA
Encryp-ted

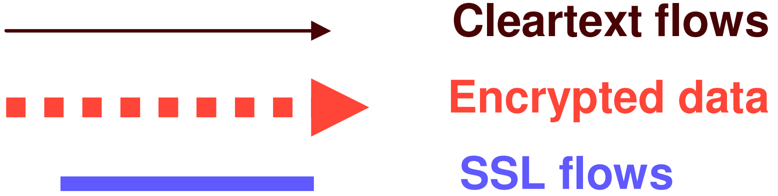
- **Basic TCP/IP stack-based TLS**
  - TLS process performed at TCP layer without requiring any application change (transparent)
  - All connections to specified port are designated as TLS required
    - Can be further qualified by source/destination IP addresses
  - Transparent TLS policies managed via Policy Agent
- **Transparent TLS can be requested by aplication**
  - Application issues transparent TLS API calls to indicate that connection should start/stop using TLS
- **TCP/IP stack-based TLS with client identification services for application**
  - Application issues TLS API calls to receive user identity information based on X.509 client certificate
- **Available to any TCP application**
  - CICS Sockets and JES/NJE are primary focus of this support
  - All programming languages supported

Application Transparent TLS (AT-TLS) Overview



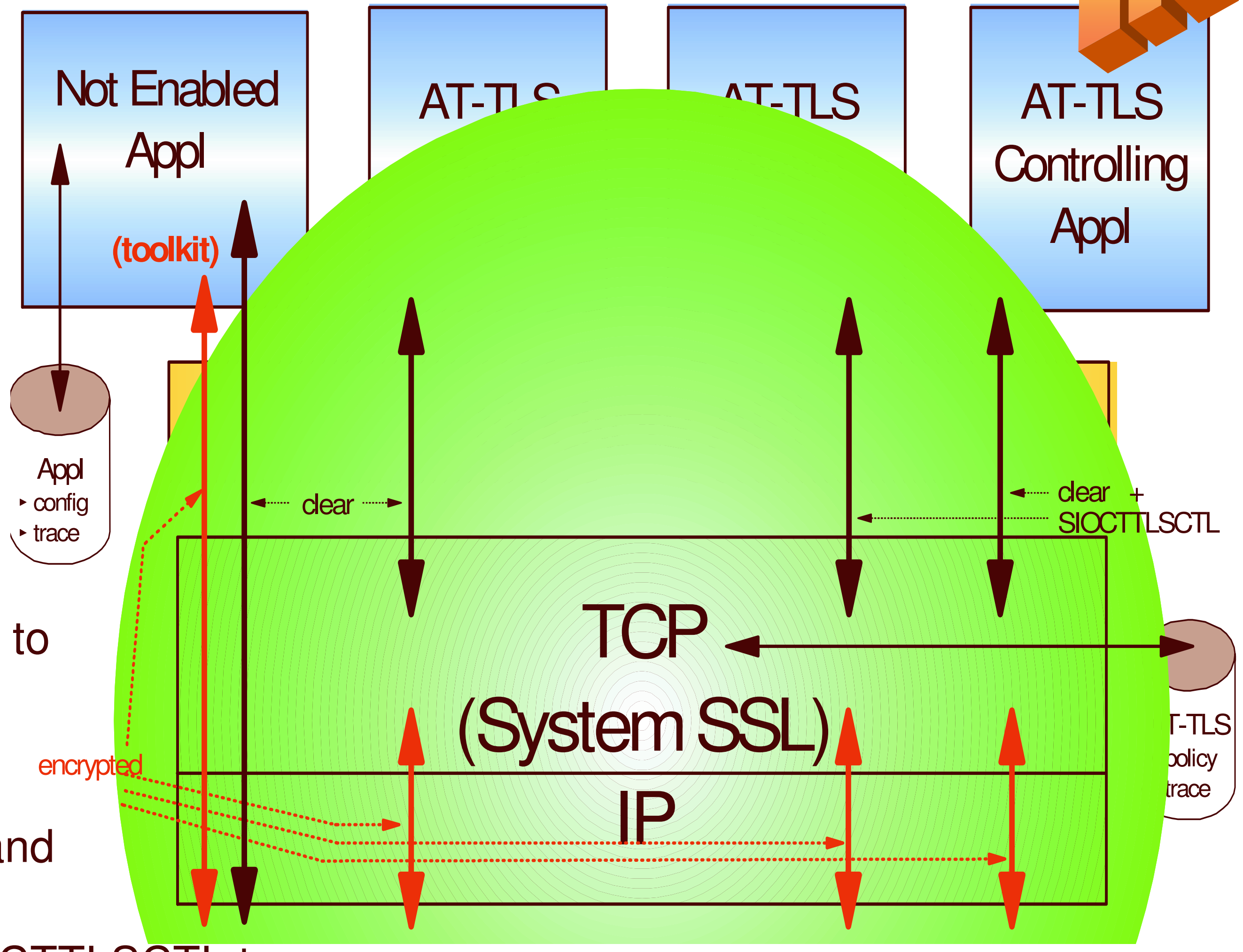
Configured AT-TLS Policy for the Application Server to use TTLS:

1. Client connects to server and connection becomes established
2. Server sends data in the clear and TCP layer queues it.
3. TCP layer invokes System SSL to perform SSL handshake under identity of the server.
4. TCP layer invokes System SSL to encrypt queued data and sends it to client.
5. Client sends encrypted data, TCP layer invokes System SSL to decrypt.
6. Server receives data in the clear.

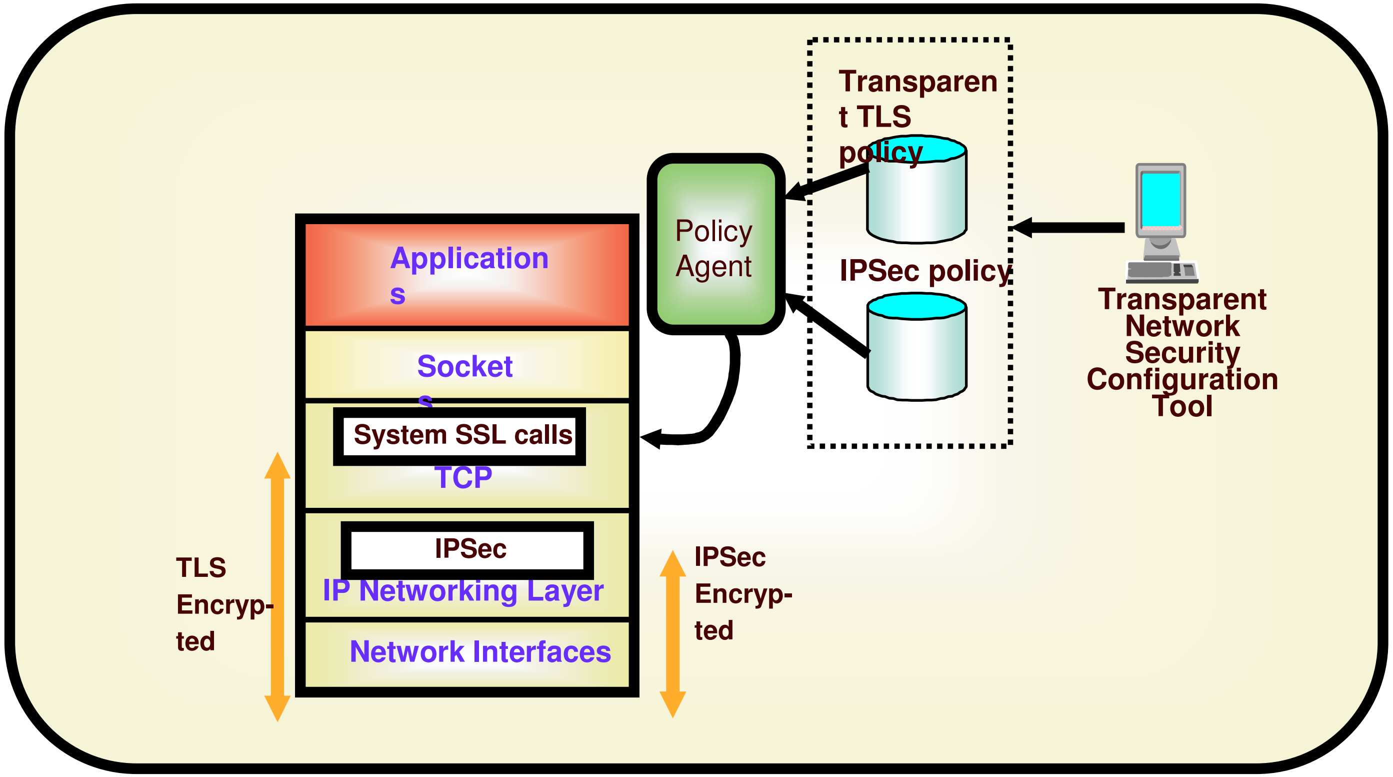


Application Transparent TLS (AT-TLS) Overview

- **4 types of applications:**
  - Not enabled
    - Enabled OFF in policy
    - Appl may do its own TLS
  - Basic
    - Policy says Enabled ON
    - Application unaware
    - TLS done transparently
  - Aware
    - Policy says Enabled ON
    - Appl uses SIOCTTLSCTL to extract TLS info
  - Controlling
    - Policy says Enabled ON and ApplicationControlled ON
    - Application also uses SIOCTTLSCTL to
      - Start Secure session
      - Cipher, reset session, etc.

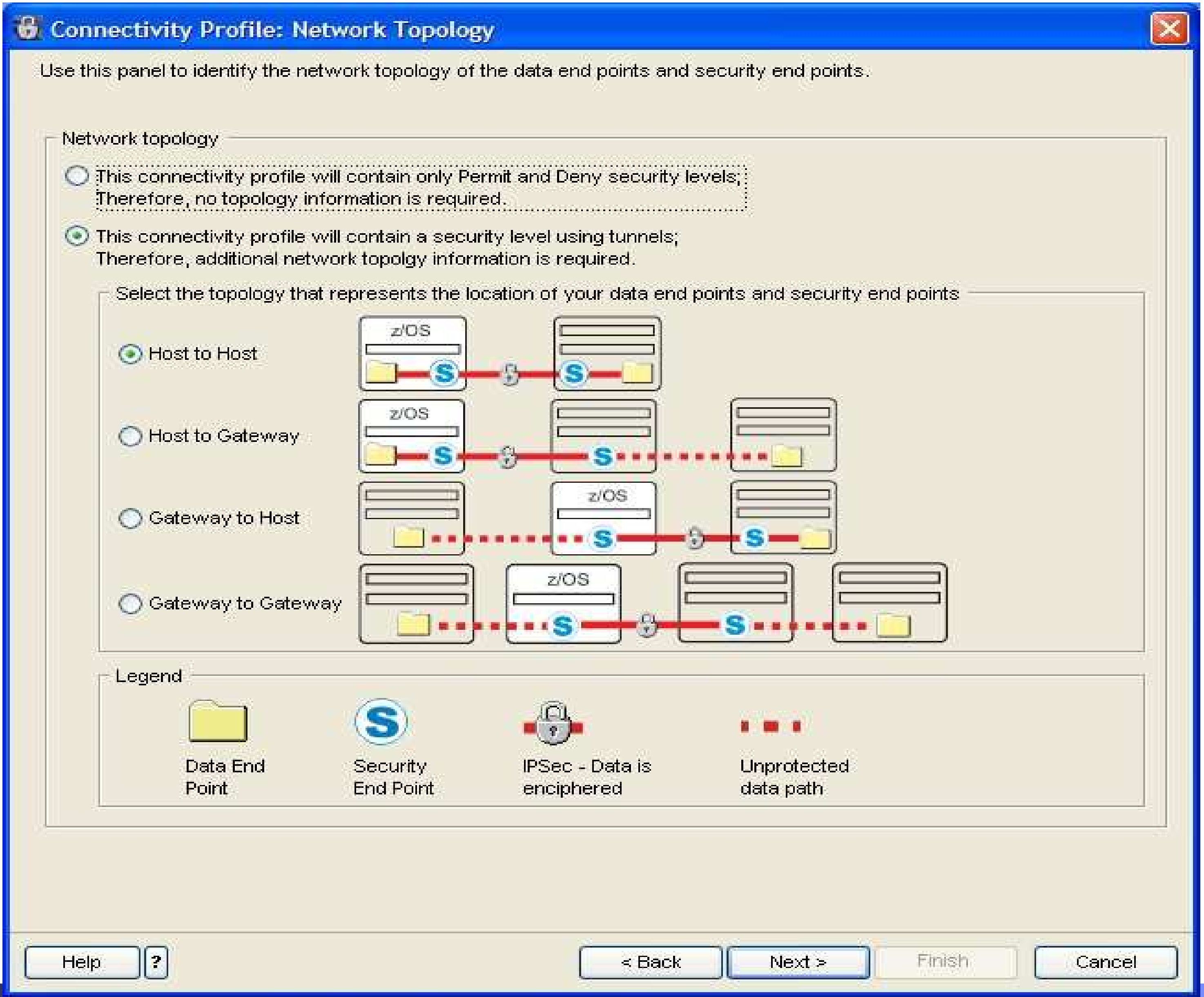


Policy-controlled application-transparent network security



- **Network security without requiring application changes**
  - IPSec
  - Transparent TLS
- **Configuration single administrative task**
  - Higher level of abstraction
    - Focus on what traffic to protect and how to protect
    - Less focus on low-level details (though available on expert panels)

IPSec configuration dialog example



Connectivity Profile: Network Topology

Use this panel to identify the network topology of the data end points and security end points.

Network topology

○ This connectivity profile will contain only Permit and Deny security levels; Therefore, no topology information is required.

● This connectivity profile will contain a security level using tunnels; Therefore, additional network topolgy information is required.

Select the topology that represents the location of your data end points and security end points

● Host to Host

○ Host to Gateway

○ Gateway to Host

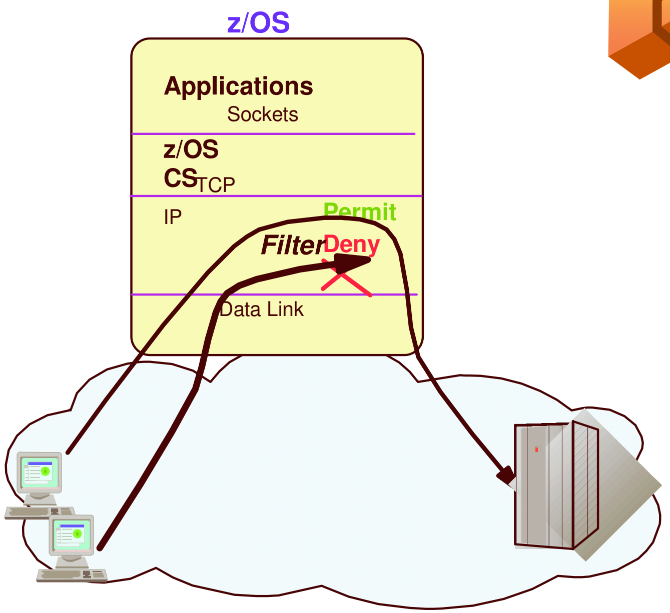
○ Gateway to Gateway

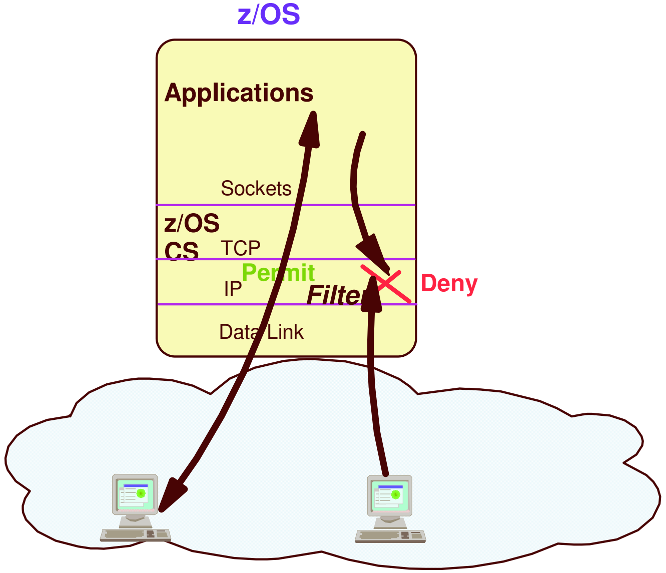
Legend

- Data End Point
- Security End Point
- IPSec - Data is enciphered
- Unprotected data path

IPSecurity Overview – IP Packet Filtering

- **Filter rules defined to match packets based on:**
  - Packet information
  - Network attributes
  - Time
- **Used to control:**
  - Traffic being routed
  - Access at destination host
- **Possible actions:**
  - Permit
  - Deny
  - Permit with manual IPSEC
  - Permit with dynamic IPSEC
  - Log





Ip filtering criteria

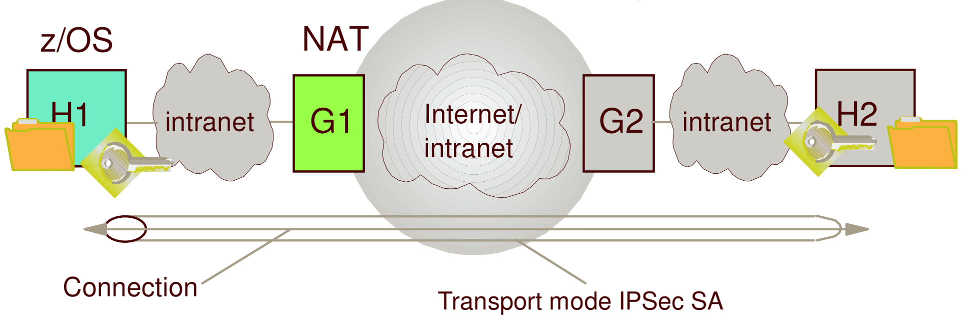
Criteria	**Description**
From packet	
Source address	Source address in IP header
Destination address	Destination address in IP header
Protocol	Protocol in the IP header
Source port	For TCP and UDP, the source port in transport header
Destination port	For TCP and UDP, the destination port in transport header
ICMP type and code	For ICMP, type and code in ICMP header
OSPF type	For OSPF, type located in OSPF header
Network attributes	
Direction	Direction of packet (inbound, outbound, both)
Routing	Packet is local if source or destination IP address exists on local host, otherwise it is routed
Security class	A virtual class that allow you to group interfaces with similar security requirements. Non-VIPA interfaces can be assigned a security class. A packet inherits the security class of the interface over which the packet is sent/received.
Time condition	
Time, Day, Week, Month	Indicates when filter rule is active

IPv4 Integrated IPSEC/VPN NAT Traversal Support

- **New IETF RFCs address the problem**
  - ►RFCs 3947 and 3948 – Negotiation of NAT Traversal and UDP Encapsulation
  - ►Does not deal with address translation of data addresses in payload
  - ►ESP only/ AH not allowed

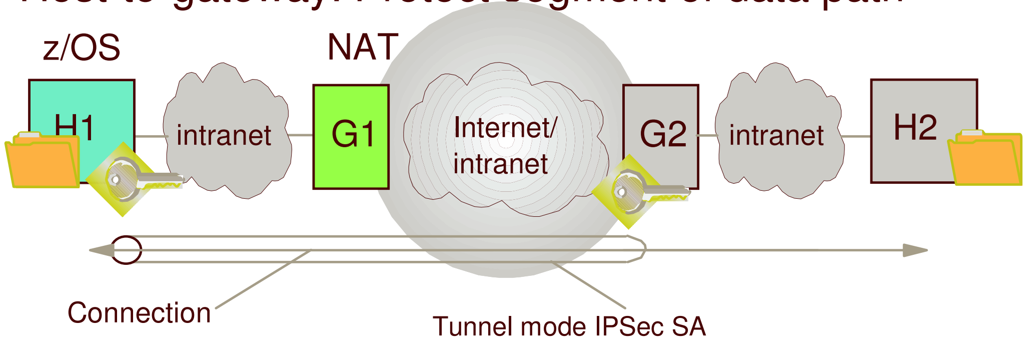
- **In V1R7, z/OS CS Host-to-Host, transport or tunnel mode supported**

Host-to-Host: End-to-End Security Association



- **In V1R7, z/OS CS Host-to-Gateway, tunnel mode only supported**

Host-to-gateway: Protect segment of data path



- **No z/OS Gateway support**
- **No NAPT Traversal support**