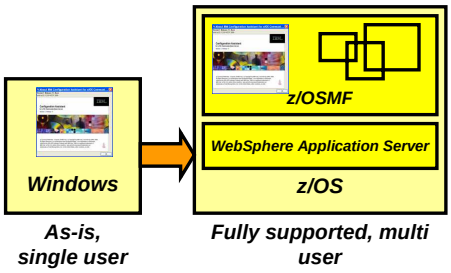
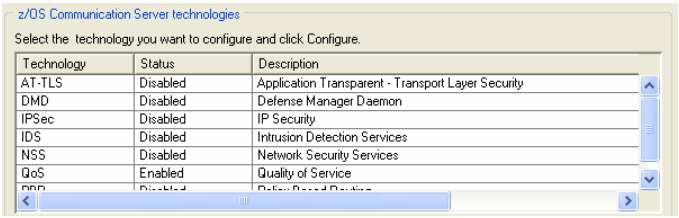
This presentation provides an overview of the new functions in z/OS V1R12 Communications Server that are contained in the system management and monitoring theme.

Simplification



- Configuration Assistant for z/OS V1R12 Communications Server
  - As new functions are added per release, gradually expand the scope of the GUI
  - Improve the Configuration Assistant integration with other z/OSMF applications
  - Improve GUI panels



You can use the Configuration Assistant, or CA, to configure a variety of policy-based functions within z/OS Communications Server, such as IP Security or Application Transparent Transport Layer Security (AT-TLS).

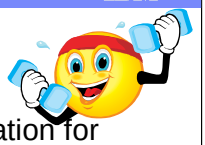
The CA began as a web download stand-alone Windows application. Versions of the code are available that support Communications Server at V1R7 to V1R12.

Now the CA is available as a plug-in component of the z/OS Management Facility, or z/OSMF. In this environment, CA is accessed through a web browser. Configuration data is generated for Communications Server at V1R11 and V1R12. This code is fully supported as part of the z/OSMF product.

Several enhancements are made to the interface of the z/OSMF client. In V1R11, small messages issued by the CA took over the entire panel. In V1R12, such messages appear in a popup window instead.

In V1R11, panels with tabs showed the tabs along the left side of the panel. Usability practices prefer tabs to be shown as a row along the top of the panel. Tabs on top are now shown for tabbed panels in V1R12. Also, several changes are made to make the CA more accessible to people with disabilities.

IBM Health Checker for z/OS OMPROUTE checks



- Large routing table in TCP/IP can cause high processor utilization for route changes (adds and deletes)
- Noticeable performance degradation in OMPROUTE, OMVS, and the TCP/IP stack as number of routes increase
- The time to process route changes can exceed OMPROUTE's Dead Router Interval for OSPF routes
  - Adjacencies with neighbors can be lost
  - Network connectivity problems can occur
- Most customer sites typically use 50-500 unique routes.
- New health checks are implemented in z/OS V1R12 to monitor the number of indirect routes in a TCP/IP stack
  - Warnings to be issued if number of indirect IPV4 or IPv6 routes exceed configurable limit (default is 2000)

A routing table that is considered to be excessive (2000 routes or more) can cause inefficiency in network design and less than optimal performance for OMPROUTE and TCP/IP. Most z/OS sites appear to have 50-500 unique routes. The overall performance degrades further with tracing enabled.

IBM service frequently tells customers with more than 2000 routes to reduce the number of routes after determining that performance degradations in OMPROUTE and TCP/IP were caused by the excessive number of routes. The IP Configuration Guide documents that the routing table size should be kept to a minimum. In OSPF, reduce the routing table size by using stub areas, route summarization, or filters. Reduce the number of RIP or static routes by sub-netting or super-netting for route summarization or use filters.

In z/OS V1R12, new counters are introduced to monitor the number of indirect routes in IPv4 and IPv6 routing tables for a TCP/IP stack. These counters are used by IBMHC for the health check monitoring and for input into the informational and warning messages. IBMHC will perform the new checks to monitor the number of IPv4 and IPv6 indirect routes. There are three times when the checks are performed. The check is performed 30 minutes after TCP/IP initialization. The check is performed at intervals based on the health checker's INTERVAL setting. And there are immediate checks whenever the maximum threshold values have been exceeded or whenever the maximum threshold values have been dynamically modified by an operator.

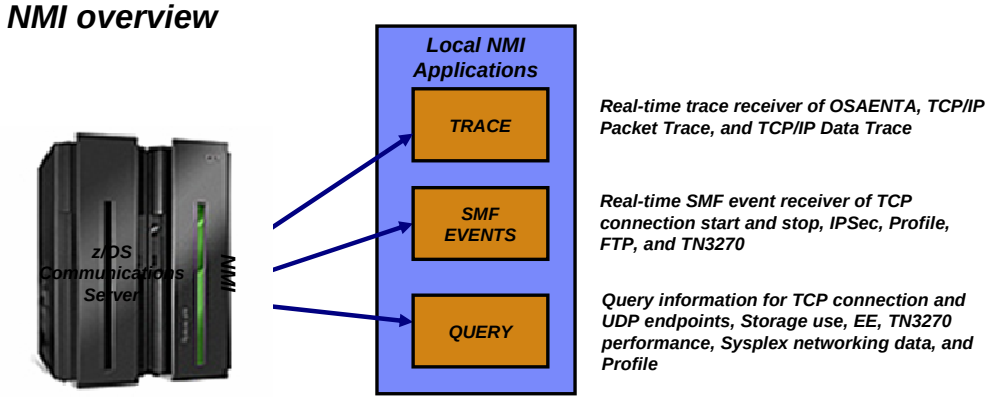
Command to drop all connections for a server

- V TCPIP,,DROP command or netstat drop command for one connection
  - Need to first issue D TCPIP,,NETSTAT,CONN to find the connection id
- Can be cumbersome if all connections with a given server need to be dropped
- z/OS V1R12 adds new parameters to the V TCPIP,,DROP command
  - VARY TCPIP,,DROP,PORT=portnum,[JOBNAME=jobname,ASID=asid]
  - VARY TCPIP,,DROP,JOBNAME=jobname,[ASID=asid]
- The extended command
  - Scans the TCP connection table for listeners matching the filters.
  - If found, scans the table again for all child connections pointing back to listener.
  - Issues RESET for each such connection found

If you want to move workload from one server application to another, for instance for maintenance purposes, you can quiesce the creation of new connections to the old server. However, you need use the Netstat DROP/-D command to end persistent connections. Issue a Netstat CONN/-c display command to get the connection ID of each persistent session to be reset. Then issue a Netstat DROP/-D command for each connection. If a server has dozens of persistent connections, this can be tedious.

To address this problem, z/OS V1R12 Communications Server extends the existing VARY TCPIP,,DROP command with new parameters to allow all TCP connections associated with a server matching the specified filter to be reset. The new parameters are modeled after the parameters of the existing VARY TCPIP,,SYSPLEX,QUIESCE command. You can specify the job name and optionally the address space ID or the port number and optionally job name and address space ID for the servers. The command processor will scan the TCP connection table for listeners matching the supplied filters. If a match is found, it will scan the table again for all child connections associated with that listener. For each one found, it will reset the connection.
If more than one server application is found to match the input filter values, the command will fail. You can re-issue the command specifying additional filter parameters to identify a specific server application.

NMI overview



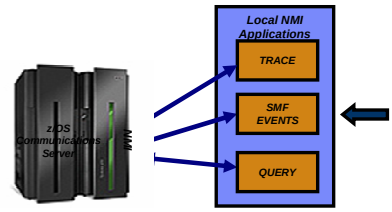
- Network Management Interface (NMI) is meant for network management solution providers
  - But can be used by anyone – fully documented in the z/OS CS library
- Three categories of APIs:
  - Real time trace data receiver (optionally also written to a CTRACE data set)
  - Real time SMF event receiver (optionally also written to the SMF data set)
  - Query interface
- Constantly being extended with new types of management data

z/OS Communications Server provides various network management interfaces. While these APIs are intended for network management applications, they are documented in the z/OS Communications Server library and can be used by anyone. The APIs fall into three categories: a real time trace data interface, a real time SMF event interface, and a query interface.

The network management interfaces are constantly being extended with new types of management data.

New SMF events being reported over NMI: sysplex events

- Provides support for NMI events with information similar to the earlier sysplex-related SNMP traps:
  - ibmMvsDVIPAStatusChange
  - ibmMvsDVIPARemoved
  - ibmMvsDVIPATargetAdded
  - ibmMvsDVIPATargetRemoved
  - ibmMvsDVIPATargetServerStarted
  - IbmMvsDVIPATargetServerEnded



- Enable real-time TCP/IP network monitoring NMI support using a new parameter on the NETMONITOR SMFSERVICE profile statement

Network management applications want improved access to sysplex networking information. This information was provided in z/OS V1R11 Communications Server through the TCP/IP callable NMI through poll-type requests. However, network management applications want automatic notifications of changes to the sysplex distributor configuration. Various NETSTAT reports can provide this information, but the report output can change every release (as more information is added to each report). Also, getting the information still requires that the application ‘poll’ for new updates.

SNMP has traps to automatically report changes to the sysplex distributor information, but SNMP is difficult to configure and is slower than the Real-time SMF NMI.

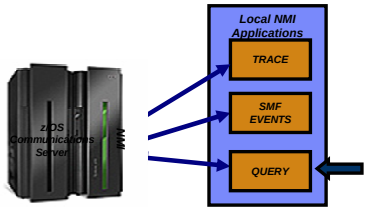
To address these difficulties, z/OS V1R12 Communications Server adds additional SMF 119 records for sysplex distributor events. These records can be written to the MVS SMF data sets or be added to the Real-time TCP/IP NMI or both. Six new SMF 119 records are defined, each corresponding to one of the existing SNMP sysplex distributor traps.

The SMFCONFIG profile statement can be used to control the writing of the six sysplex event records to the MVS SMF data sets. If keyword DVIPA is specified, the sysplex events are written to the MVS SMF data sets. If NODVIPA is specified, the events are not written to the MVS SMF data sets. NODVIPA is the default.

The NETMONITOR profile statement controls whether the six sysplex events are made available to the Real-time SMF NMI. If DVIPA is specified on the NETMONITOR statement, the records are written to the NMI interface. If NODVIPA is specified, the records are not written to the NMI interface. DVIPA is the default.

New NMI query: network interface and device information and TCP/IP global statistics

- Allows applications to obtain TCP/IP interface attributes and statistics, and TCP/IP global stack statistics using the TCP/IP query NMI:
  - GetGlobalStats to retrieve TCP/IP global stack counters
    - Similar to those on the Netstat STATS/-S report
  - GetIfs to retrieve detailed interface attribute information
    - Similar to those available on the Netstat DEVLINKS/-d report
  - GetIfsStats to retrieve interface counters
    - Similar to those available on the Netstat DEVLINKS/-d report, with the addition of some SNMP interface counters
  - GetIfsStatsExtended to retrieve DLC tuning statistics
    - Similar to those available on the VTAM TNSTAT console display



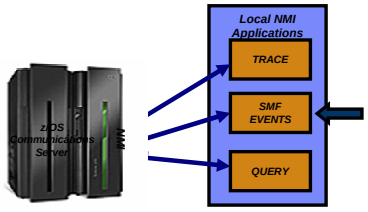
Network management applications want to provide TCP/IP interface attributes, and interface and stack global statistics to their users. Some statistics can indicate potential problems. By obtaining the statistics, the management applications can automatically alert customers to the potential problems.

Currently, there is no programming interface to obtain this information. The Netstat DEVLINKS/-d, HOME/-h, and STATS/-S reports provide some of this information, but the report output changes every release. SNMP provides some statistics, but the SNMP protocol is not performance-oriented. The VTAM TNSTAT function provides data path device statistics from the data link control (DLC) layer, but only interval counters are supported. And the VTAM TNSTAT statistics are only provided as an MVS console display or in an SMF type 50 record. So, a management application either has to parse the MVS console display output or process the SMF record.

In response to these requests from network management applications, z/OS V1R12 Communications Server provides new TCP/IP callable NMI requests to obtain TCP/IP interface attributes, and interface and stack global statistics. The NMI requests can be invoked by authorized network management applications. See the "Network management interfaces" chapter of z/OS V1R12 Communications Server: IP Programmer's Guide and Reference for detailed information about these new NMI requests.

New SMF records and new NMI SMF events for CSSMTP

- Use SMF to capture records about the activity of CSSMTP
  - Can be used for accounting, performance, and billing purposes
  - Records written to both SMF and the real-time SMF events NMI interface



- CSSMTP SMF records:
  - A configuration record when CSSMTP is started and when the configuration is refreshed
  - A spool-related record when CSSMTP has completed processing a spool file
  - A connection record when a connection to a target server ends
  - A mail record when CSSMTP has completed processing mail message
  - A statistical record when a recording interval ends and when CSSMTP ends

There were several requirements requesting accounting, statistics, and performance data regarding the mail message processing. The method used in z/OS is to create records using the System Management Facility (SMF).

Accounting data included a description of the spool file with the job name, job identifier, the size of the spool file and other fields. For each mail message, the source of the mail message (which spool file), the mail from name, the recipients of the mail message and subject. Statistics data included time stamps about spool file and mail message processing, the size of the spool file and each mail message. Performance data includes the traffic over each connection and each target server.

By providing APPLDATA for the server connections, the state of the CSSMTP connections can be monitored.

The current RETRYLIMIT value is two hours. Some customers requested a longer interval.

CSSMTP provides SMF records about mail processing. There are five new subtype records for the type 119 records.

- Subtype 48 contain the configuration data from initialization and after each MODIFY REFRESH command.

- Subtype 49 contain statistics about each connection to a target server that transfer mail messages.

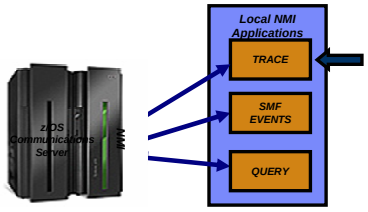
- Subtype 50 contain identification and statistics about each mail message.

- Subtype 51 contain identification and statistics about each spool file.

- Subtype 52 contain global and health check statistics at each SMF interval and at termination.

New data trace records to indicate start and end of a "data flow"

- The TCP/IP data trace includes data buffers as they pass between the PFS layer and the TCP/IP transport protocol layer
  - TCP, UDP, and RAW data buffers
- There has so far not been any indication in the data trace about when a TCP connection or a UDP association started and stopped
  - Data flow start and stop records are added to the data trace in z/OS V1R12
- Only supported for TCP and UDP sockets
  - Start record written on the first socket read or write operation
  - End record written when the socket is closed
  - Start/End records are created by default. No changes to VARY TCPIP,,DATTRACE command
- Removes the guess-work for connection start and end when interpreting data traces

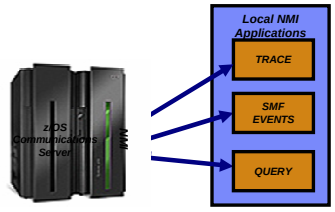


In release z/OS V1R11 and earlier, data trace records do not provide an indication of the start or end of socket data flow. This makes it difficult for management applications and users to understand the flow of data for a socket. Packet trace records do provide an indication of the start or end of socket data flow. But, when using AT-TLS and IPSec, the information in the packet trace record is encrypted. The encryption makes it impossible to interpret the start and end information thereby making analysis problematic. Except for AT-TLS packets, data trace records are created before outbound information has been encrypted, and after inbound information has been decrypted. So the records are not affected by encryption. In order to view AT-TLS packet data in data trace, the AT-TLS policy must be changed.

To resolve this problem, data trace has been changed to generate two additional records to indicate the start and end of data flow for the socket. This is currently supported only for TCP and UDP packets. The start record is written when the first socket read or write operation is performed. The end record is written when the socket is closed. It is worth mentioning that these new start and end records are created by default, thus no changes have been made to the VARY TCPIP,,DATTRACE command.

Enhance packet trace for sysplex distributor VIPAROUTE traffic

- Apply Packet Trace filters to Sysplex Distributor VIPAROUTE traffic
  - Sysplex Distributor encapsulates VIPAROUTE traffic with GRE header, for IPv4 traffic, or an IPv6 header, for IPv6 traffic
    - Existing filter support only operates on the outer packet header, not the encapsulated packet
  - Packet Trace can now filter on the destination DVIPA address, the ports located inside the encapsulated packet, or both
- In addition, the next hop address is now included in the packet trace



When VIPAROUTE statements are defined to a sysplex distributor to select routes, the sysplex distributor encapsulates the IPv4 packet with a GRE header before sending it to the target stack. IPv6 packets are encapsulated with an additional IPv6 header. Multiple versions of GRE headers have been defined, but the sysplex distributor uses version 0 defined in RFC1701.

Packets cannot be filtered by the client or target's IP addresses or ports for VIPAROUTE traffic since the packet is encapsulated by an additional header. In addition, packet trace does not provide any information about the next hop chosen for outbound packets. Both issues make VIPAROUTE related problems more difficult to diagnose.

The solution is to add code to packet trace to look beyond the encapsulation header for VIPAROUTE traffic. This allows filtering to be performed on the inner packet. Additionally, the next hop IP address is provided for all outbound packets. This information will only be viewable if the packet trace is formatted with the "FULL" option.

Enhancements to TCP/IP storage command

- D TCPIP,,STOR
- Common (ECSA) usage information includes the size of the TCP/IP load modules loaded into common by dynamic LPA
  - Load module size is a stable value
  - Might be a large percentage of common usage value
  - Might mask workload related fluctuations/growth in common storage usage

```
TCPCS     STORAGE          CURRENT    MAXIMUM     LIMIT
TCPCS     ECSA               9645K     10087K   NOLIMIT
TCPCS     POOL              14017K     14171K   NOLIMIT
TCPCS     64-BIT COMMON         1M         1M   NOLIMIT
DISPLAY TCPIP STOR COMPLETED SUCCESSFULLY
```

- In z/OS V1R12, ECSA usage for load modules moved to separate line of the display
- Similar changes made to the storage callable NMI interface

```
TCPCS     STORAGE          CURRENT    MAXIMUM     LIMIT
TCPCS     ECSA               2822K     2935K   NOLIMIT
TCPCS     POOL              14194K    14194K   NOLIMIT
TCPCS     64-BIT COMMON         1M        1M   NOLIMIT
TCPCS     CSA MODULES        7419K     7419K   NOLIMIT
DISPLAY TCPIP STOR COMPLETED SUCCESSFULLY
```


The “D TCPIP,,STOR” command displays information about the use of storage by z/OS Communications Server. The amounts of extended common storage (ECSA) in use, pooled private storage in use, and 64-bit common storage in use are displayed. The information is also available through the Network Management Interface using the GetStorageStatistics request.

The value displayed for ECSA storage includes the size of the TCP/IP load modules which are loaded using dynamic LPA functions. The size of these load modules is a stable value and might be a large percentage of the value displayed for common storage usage. This makes it difficult to recognize significant storage growth in common storage.

For example, assume the current ECSA usage value is 10 megabytes, of which eight megabytes is load module storage. That leaves two megabytes actually being used for control blocks. If the ECSA storage usage increases two megabytes to 12 megabytes. Using the current display this looks like a 20 percent increase in ECSA storage usage. But it is actually a 100 percent increase in ECSA storage used for control blocks.

The ECSA storage value is updated to only reflect the amount of storage used for control blocks and does not include the size of the load modules loaded into common storage. The display is updated to include a new line which shows the amount of common storage used for load modules.

The NMI is also updated to remove the load module storage from the common storage value and to add a new value for load modules in common storage.

Feedback

Your feedback is valuable

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

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

Click to send e-mail feedback:

mailto:iea@us.ibm.com?subject=Feedback_about_wnmgmt.ppt

This module is also available in PDF format at: ../wnmgmt.pdf

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

Trademarks, copyrights, and disclaimers

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

Windows, and the Windows logo are registered trademarks of Microsoft Corporation in the United States, other countries, or both.

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

