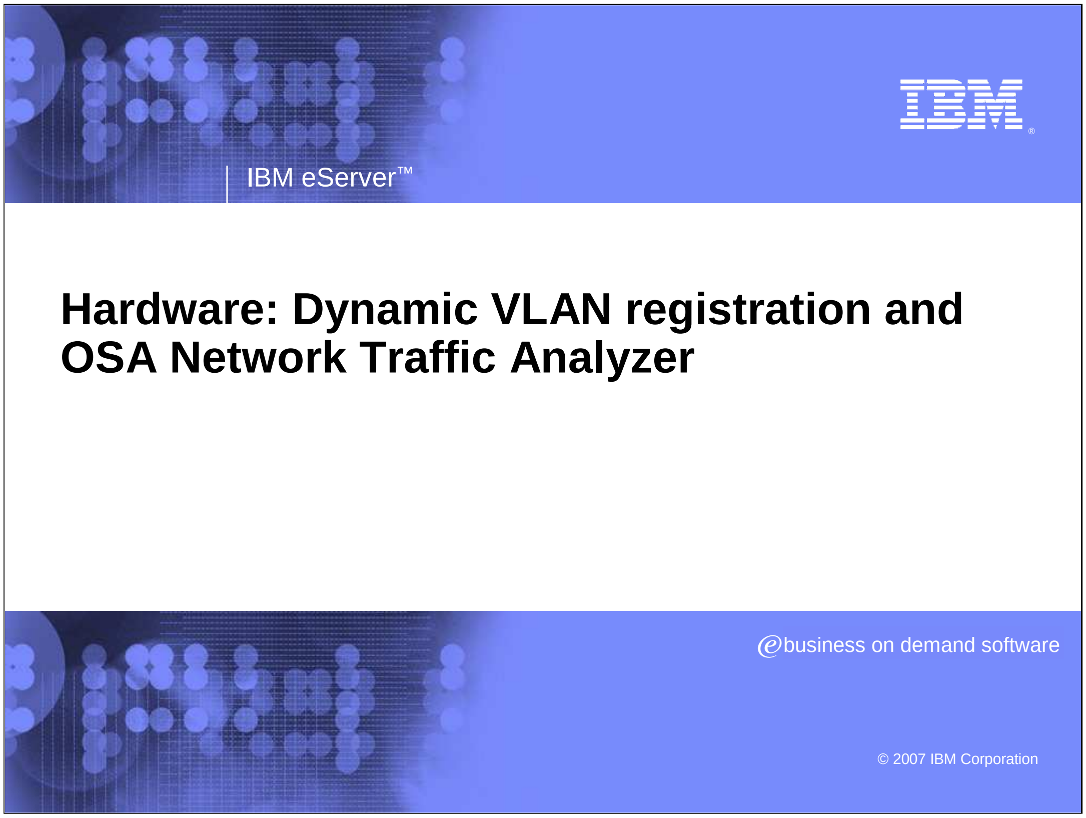
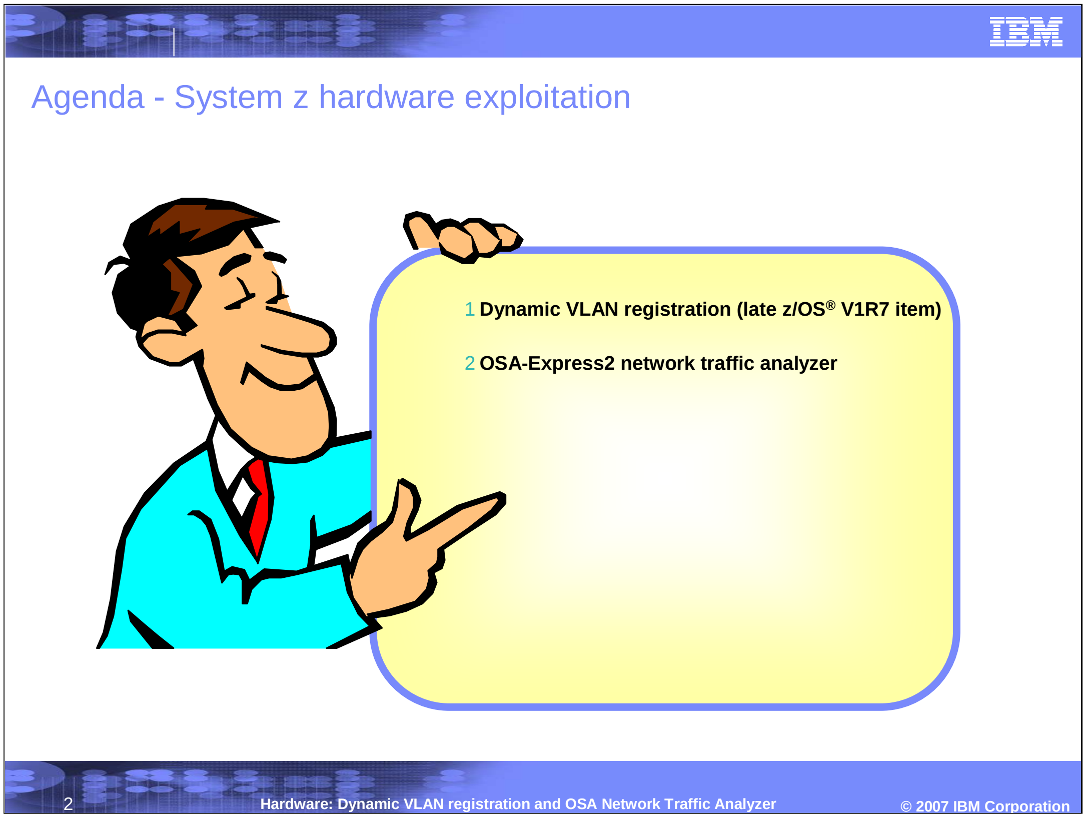


IBM eServer™

Hardware: Dynamic VLAN registration and OSA Network Traffic Analyzer

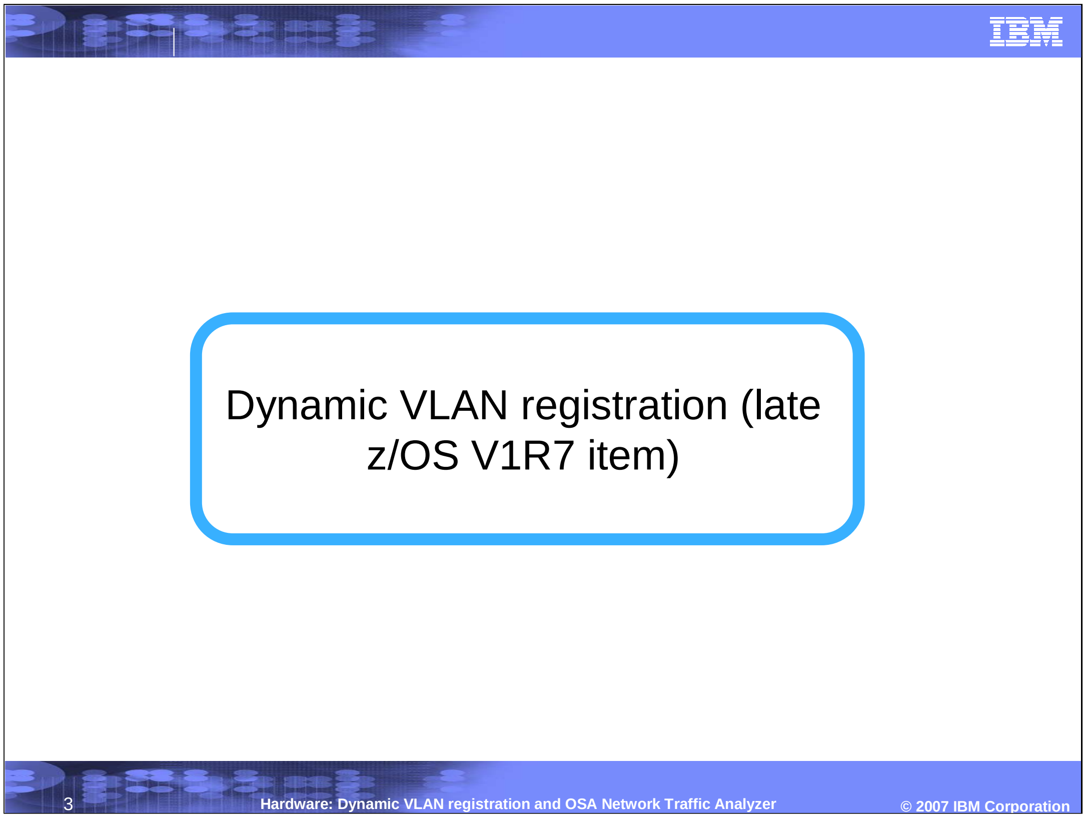
*e*business on demand software



Agenda - System z hardware exploitation

1 Dynamic VLAN registration (late z/OS® V1R7 item)

2 OSA-Express2 network traffic analyzer



Dynamic VLAN registration (late z/OS V1R7 item)

Dynamic VLAN registration - GVRP - simplifies VLAN administration

- **OSA has recently added support for dynamic VLAN registration protocols**
  - Both the OSA microcode and the switch to which the OSA port is connected must support dynamic VLAN registration
  - The protocol is referred to as GARP (Generic Attribute Registration Protocol) VLAN Registration Protocol (GVRP)
  - Simplifies management of VLAN environments

- **New TCP/IP profile keywords on the IPv4 LINK statement and the IPv6 INTERFACE statement for QDIO network interfaces are used to control if TCP/IP should request dynamic VLAN registration or not**
  - Default is to not use dynamic VLAN registration

```
LINK .... NODYNVLANREG/DYNVLANREG
INTERFACE .... NODYNVLANREG/DYNVLANREG
```

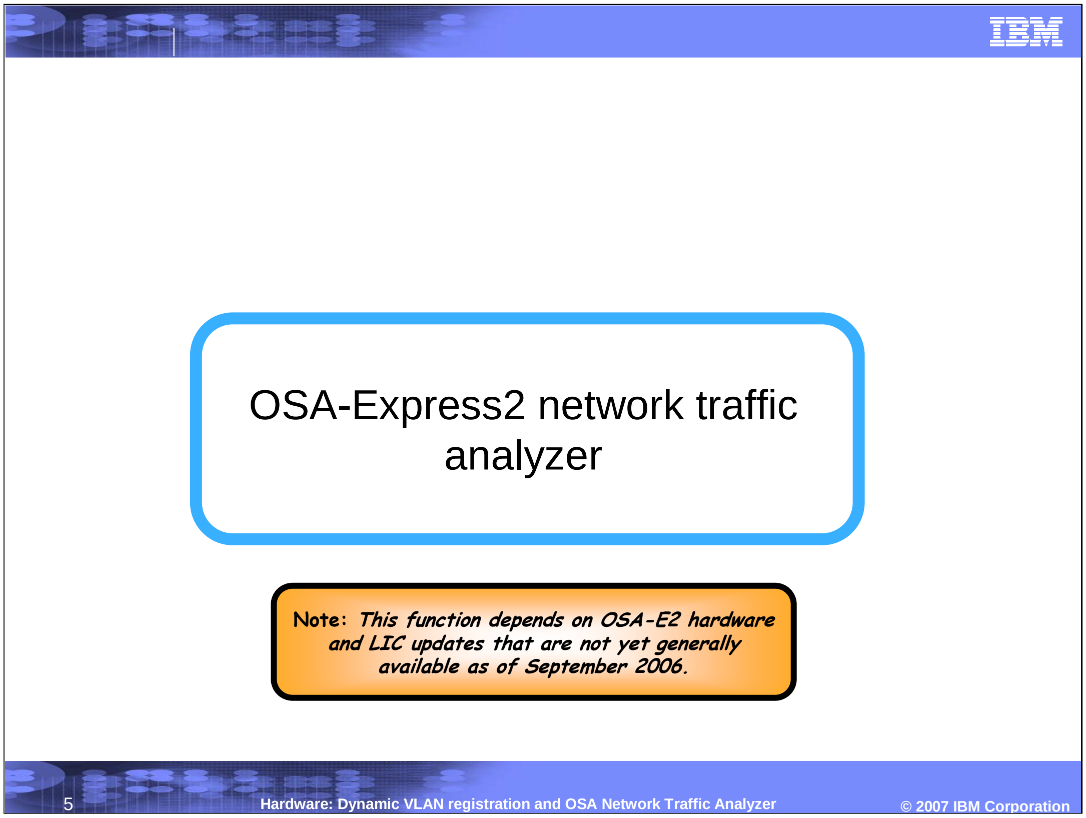
DYNVLANREG | NODYNVLANREG

This parameter controls whether or not the VLAN ID for this link is dynamically or statically registered with the physical switch on the LAN.

Restriction: This parameter is only applicable if a VLAN ID is specified on the statement. If no VLAN ID is specified then this parameter is ignored.

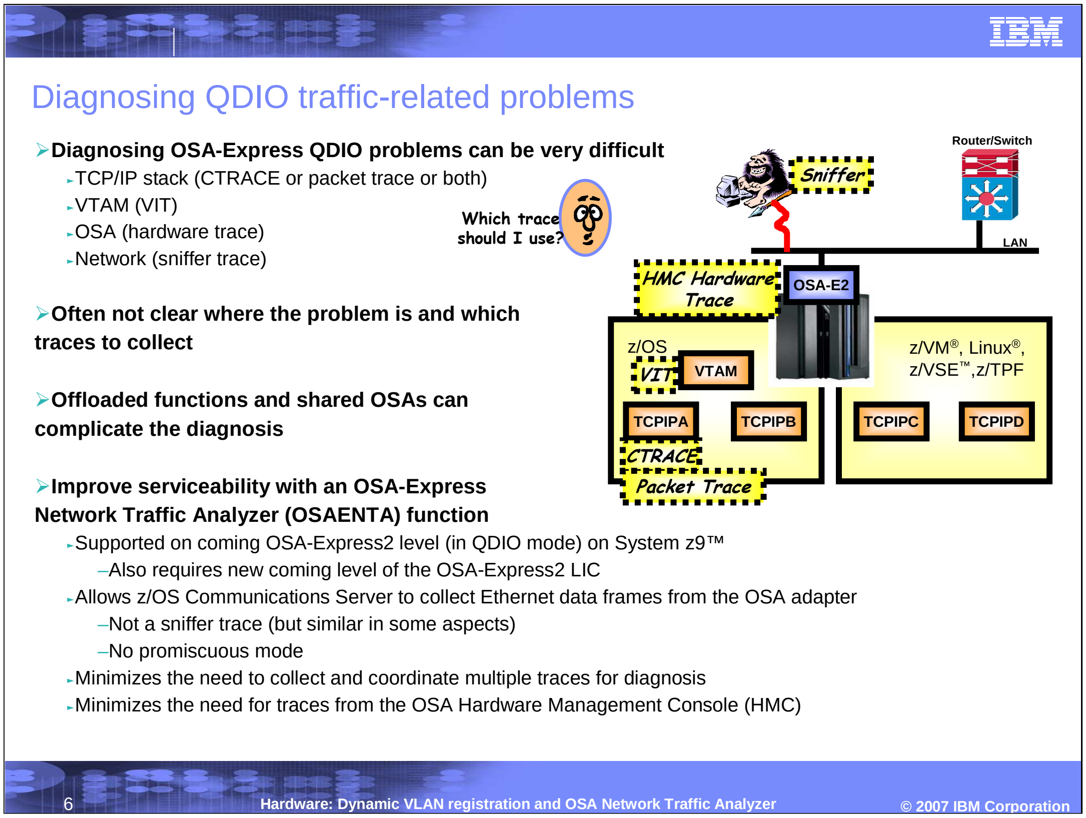
Dynamic registration of VLAN IDs is handled by the OSA feature and the physical switch on your LAN. Therefore, both must be at a level which provides the necessary hardware support for dynamic VLAN ID registration, in order for the DYNVLANREG parameter to be effective.

- **New fields on a netstat devlinks report will indicate if the dynamic VLAN registration is supported by the OSA port and if dynamic VLAN registration is configured for the link or interface**
- **CS z/OS V1R7 APAR PK05337**
  - PTFs UK06129 and UK06130
- **Part of base CS z/OS V1R8**



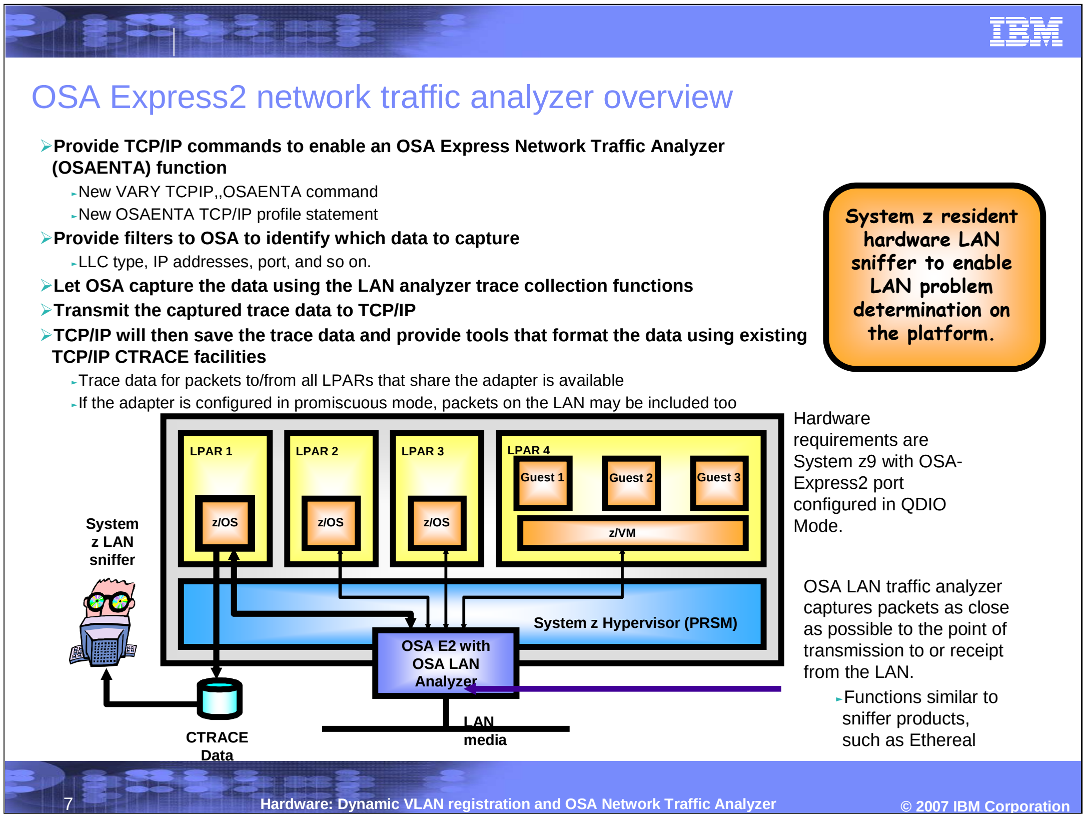
OSA-Express2 network traffic analyzer

Note: *This function depends on OSA-E2 hardware and LIC updates that are not yet generally available as of September 2006.*



Diagnosing QDIO traffic-related problems

- **Diagnosing OSA-Express QDIO problems can be very difficult**
  - TCP/IP stack (CTRACE or packet trace or both)
  - VTAM (VIT)
  - OSA (hardware trace)
  - Network (sniffer trace)
- **Often not clear where the problem is and which traces to collect**
- **Offloaded functions and shared OSAs can complicate the diagnosis**
- **Improve serviceability with an OSA-Express Network Traffic Analyzer (OSAENTA) function**
  - Supported on coming OSA-Express2 level (in QDIO mode) on System z9™
    - Also requires new coming level of the OSA-Express2 LIC
  - Allows z/OS Communications Server to collect Ethernet data frames from the OSA adapter
    - Not a sniffer trace (but similar in some aspects)
    - No promiscuous mode
  - Minimizes the need to collect and coordinate multiple traces for diagnosis
  - Minimizes the need for traces from the OSA Hardware Management Console (HMC)



OSA Express2 network traffic analyzer overview

- **Provide TCP/IP commands to enable an OSA Express Network Traffic Analyzer (OSAENTA) function**
  - New VARY TCPIP,,OSAENTA command
  - New OSAENTA TCP/IP profile statement
- **Provide filters to OSA to identify which data to capture**
  - LLC type, IP addresses, port, and so on.
- **Let OSA capture the data using the LAN analyzer trace collection functions**
- **Transmit the captured trace data to TCP/IP**
- **TCP/IP will then save the trace data and provide tools that format the data using existing TCP/IP CTRACE facilities**
  - Trace data for packets to/from all LPARs that share the adapter is available
  - If the adapter is configured in promiscuous mode, packets on the LAN may be included too

System z resident hardware LAN sniffer to enable LAN problem determination on the platform.

Hardware requirements are System z9 with OSA-Express2 port configured in QDIO Mode.

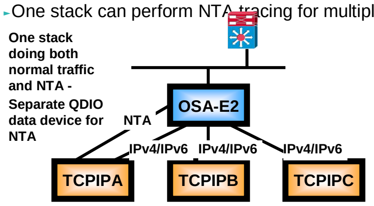
OSA LAN traffic analyzer captures packets as close as possible to the point of transmission to or receipt from the LAN.

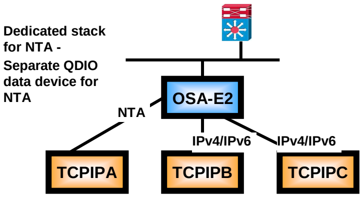
- Functions similar to sniffer products, such as Ethereal

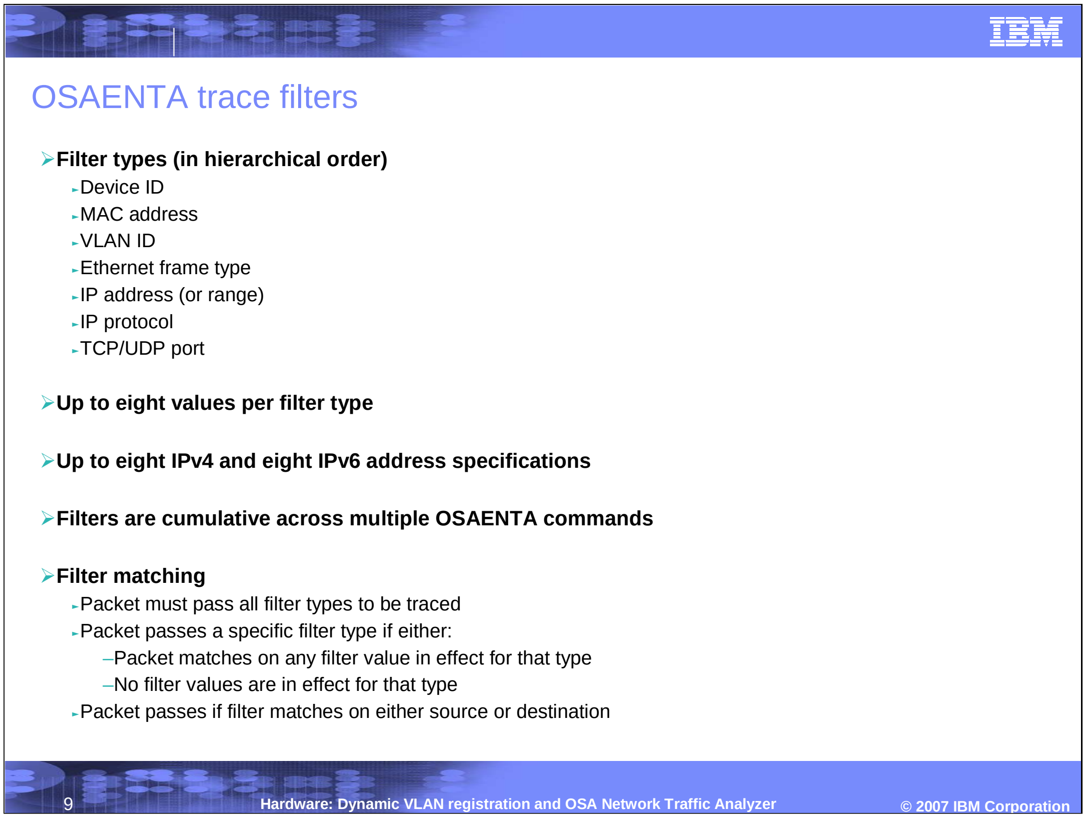
Usage scenarios

- **Controlled by z/OS Communications Server**
  - New OSAENTA MVS™ console command and TCP/IP profile statement:
    - Define trace filters and parameters
    - OSA sends trace records to the z/OS stack
  - Save and format the data using existing Ctrace facilities

- **Collected by OSA and handed up to a z/OS TCP/IP stack**
  - Ability to see:
    - ARP packets
    - MAC headers (including VLAN tags)
    - Packets to/from other stacks shared by the OSA (which could be z/VM or z/Linux)
    - SNA packets - limited to:
      - Enterprise Extender data when OSA configured in QDIO layer 3 mode
      - Data to/from Communication Controller for Linux (CCL) on System z when OSA configured in QDIO layer 2 mode
  - The OSA collects the data when it is sent across the PCI to the physical port (sometimes referred to as the NIC).
  - The OSA also collects data for LPAR-LPAR packets which do not go onto the LAN.
  - OSA supports only one stack sharing the OSA to perform NTA tracing.
  - One stack can perform NTA tracing for multiple OSAs.


One stack doing both normal traffic and NTA -
Separate QDIO data device for NTA
OSA-E2
NTA
IPv4/IPv6
IPv4/IPv6
IPv4/IPv6
TCPIPA
TCPIPB
TCPIPC





OSAENTA trace filters

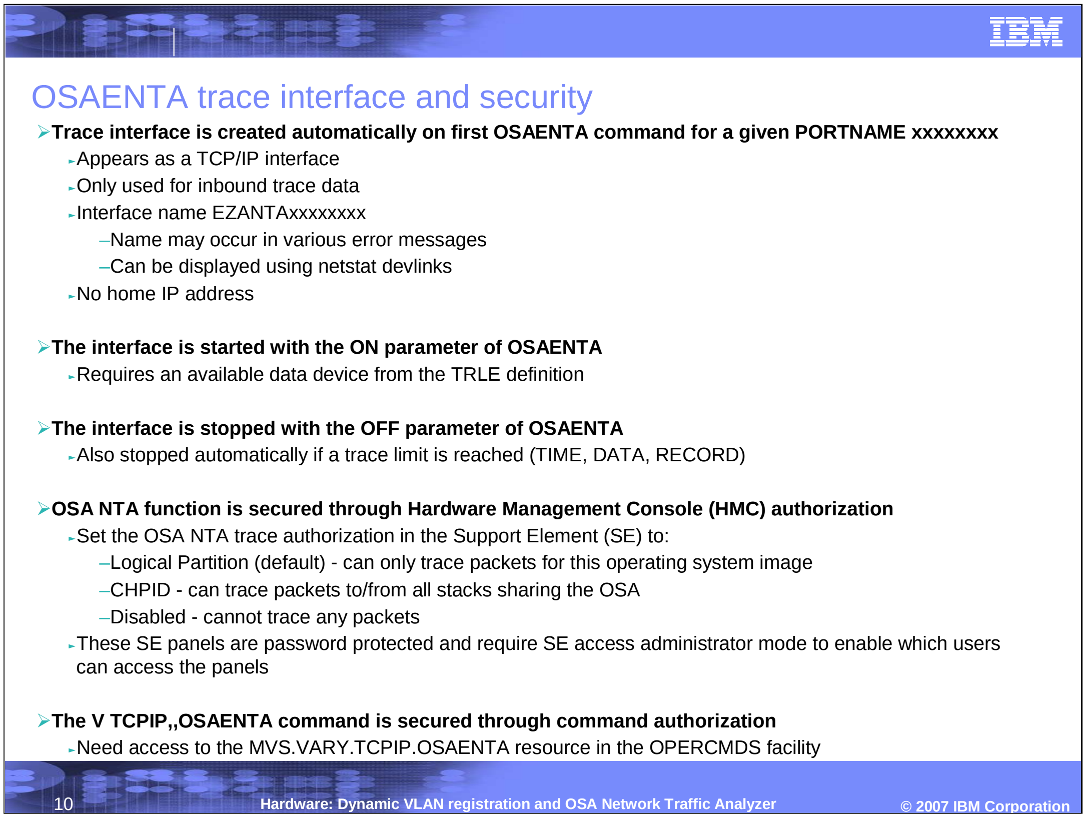
- **Filter types (in hierarchical order)**
  - Device ID
  - MAC address
  - VLAN ID
  - Ethernet frame type
  - IP address (or range)
  - IP protocol
  - TCP/UDP port

- **Up to eight values per filter type**

- **Up to eight IPv4 and eight IPv6 address specifications**

- **Filters are cumulative across multiple OSAENTA commands**

- **Filter matching**
  - Packet must pass all filter types to be traced
  - Packet passes a specific filter type if either:
    - Packet matches on any filter value in effect for that type
    - No filter values are in effect for that type
  - Packet passes if filter matches on either source or destination



OSAENTA trace interface and security

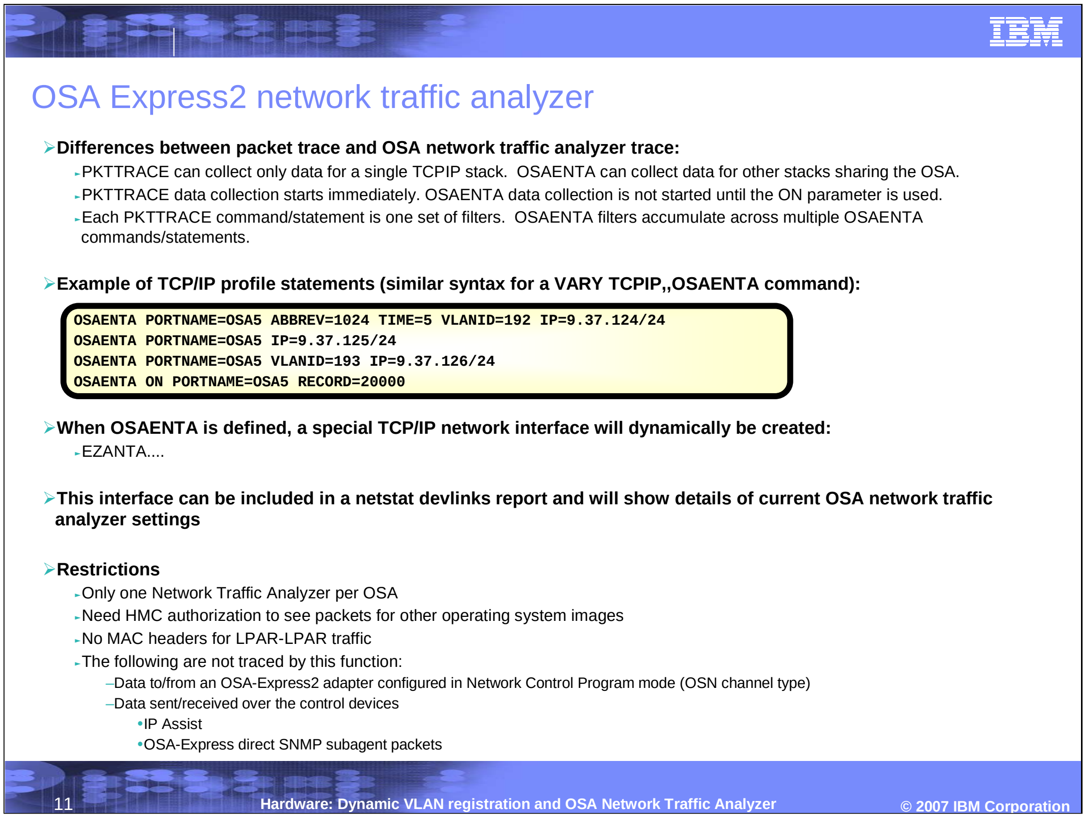
- **Trace interface is created automatically on first OSAENTA command for a given PORTNAME xxxxxxxx**
  - Appears as a TCP/IP interface
  - Only used for inbound trace data
  - Interface name EZANTAxxxxxxxx
    - Name may occur in various error messages
    - Can be displayed using netstat devlinks
  - No home IP address

- **The interface is started with the ON parameter of OSAENTA**
  - Requires an available data device from the TRLE definition

- **The interface is stopped with the OFF parameter of OSAENTA**
  - Also stopped automatically if a trace limit is reached (TIME, DATA, RECORD)

- **OSA NTA function is secured through Hardware Management Console (HMC) authorization**
  - Set the OSA NTA trace authorization in the Support Element (SE) to:
    - Logical Partition (default) - can only trace packets for this operating system image
    - CHPID - can trace packets to/from all stacks sharing the OSA
    - Disabled - cannot trace any packets
  - These SE panels are password protected and require SE access administrator mode to enable which users can access the panels

- **The V TCPIP,,OSAENTA command is secured through command authorization**
  - Need access to the MVS.VARY.TCPIP.OSAENTA resource in the OPERCMDS facility



OSA Express2 network traffic analyzer

- **Differences between packet trace and OSA network traffic analyzer trace:**
  - PKTTRACE can collect only data for a single TCPIP stack. OSAENTA can collect data for other stacks sharing the OSA.
  - PKTTRACE data collection starts immediately. OSAENTA data collection is not started until the ON parameter is used.
  - Each PKTTRACE command/statement is one set of filters. OSAENTA filters accumulate across multiple OSAENTA commands/statements.

- **Example of TCP/IP profile statements (similar syntax for a VARY TCPIP,,OSAENTA command):**

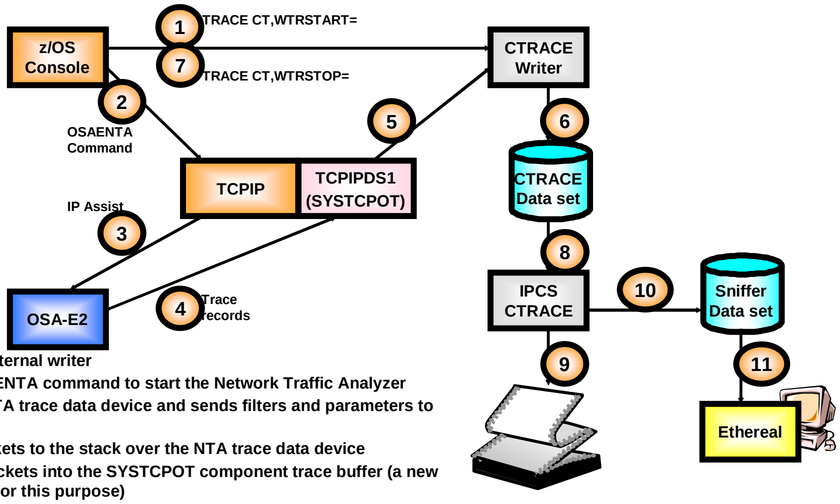
```
OSAENTA PORTNAME=OSA5 ABBREV=1024 TIME=5 VLANID=192 IP=9.37.124/24
OSAENTA PORTNAME=OSA5 IP=9.37.125/24
OSAENTA PORTNAME=OSA5 VLANID=193 IP=9.37.126/24
OSAENTA ON PORTNAME=OSA5 RECORD=20000
```

- **When OSAENTA is defined, a special TCP/IP network interface will dynamically be created:**
  - EZANTA....

- **This interface can be included in a netstat devlinks report and will show details of current OSA network traffic analyzer settings**

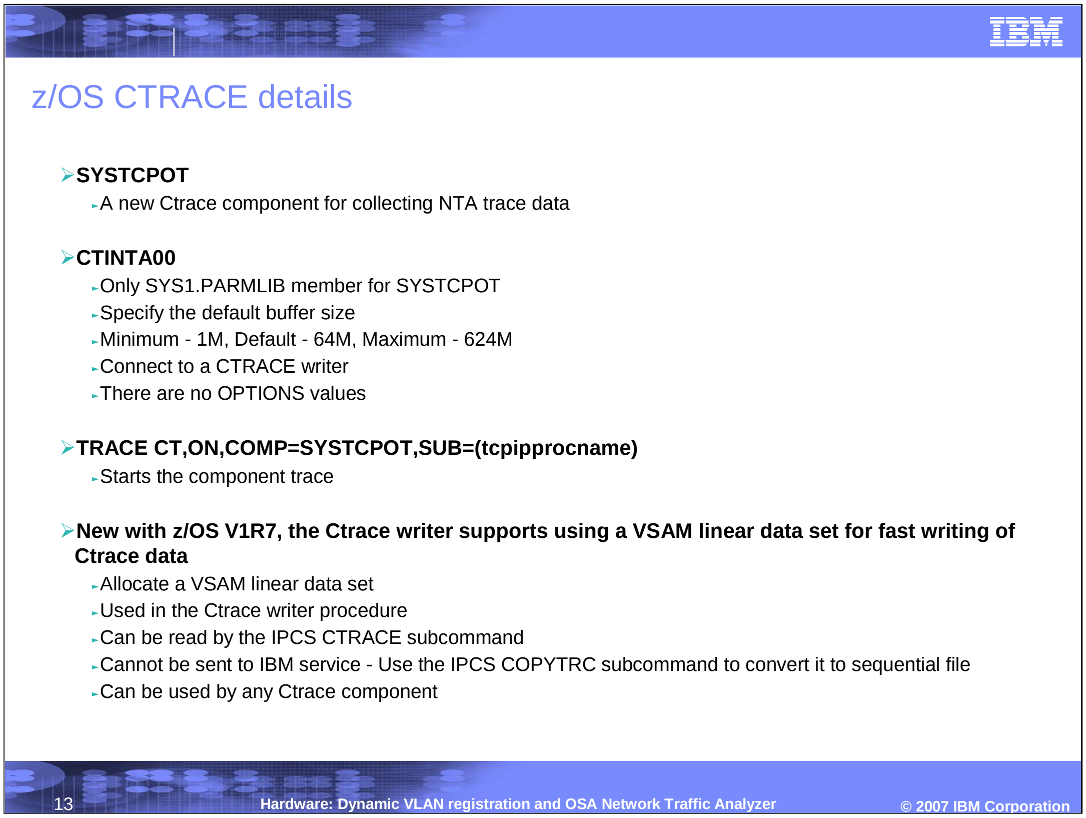
- **Restrictions**
  - Only one Network Traffic Analyzer per OSA
  - Need HMC authorization to see packets for other operating system images
  - No MAC headers for LPAR-LPAR traffic
  - The following are not traced by this function:
    - Data to/from an OSA-Express2 adapter configured in Network Control Program mode (OSN channel type)
    - Data sent/received over the control devices
      - IP Assist
      - OSA-Express direct SNMP subagent packets

NTA functional overview and trace flows



1 Operator starts the external writer
2 Operator issues OSAENTA command to start the Network Traffic Analyzer
3 Stack activates the NTA trace data device and sends filters and parameters to OSA using IP Assist
4 OSA sends trace packets to the stack over the NTA trace data device
5 Stack copies trace packets into the SYSTCPOT component trace buffer (a new CTRACE component for this purpose)
6 Ctrace writer copies the component trace buffer into a Ctrace data set
7 Operator stops the external writer
8 IPCS reads the Ctrace data set
9 IPCS formats the trace data to produce IPCS reports

- IPCS optionally formats the trace data into a Sniffer data set
- Ethereal is optionally used to process the Sniffer data set



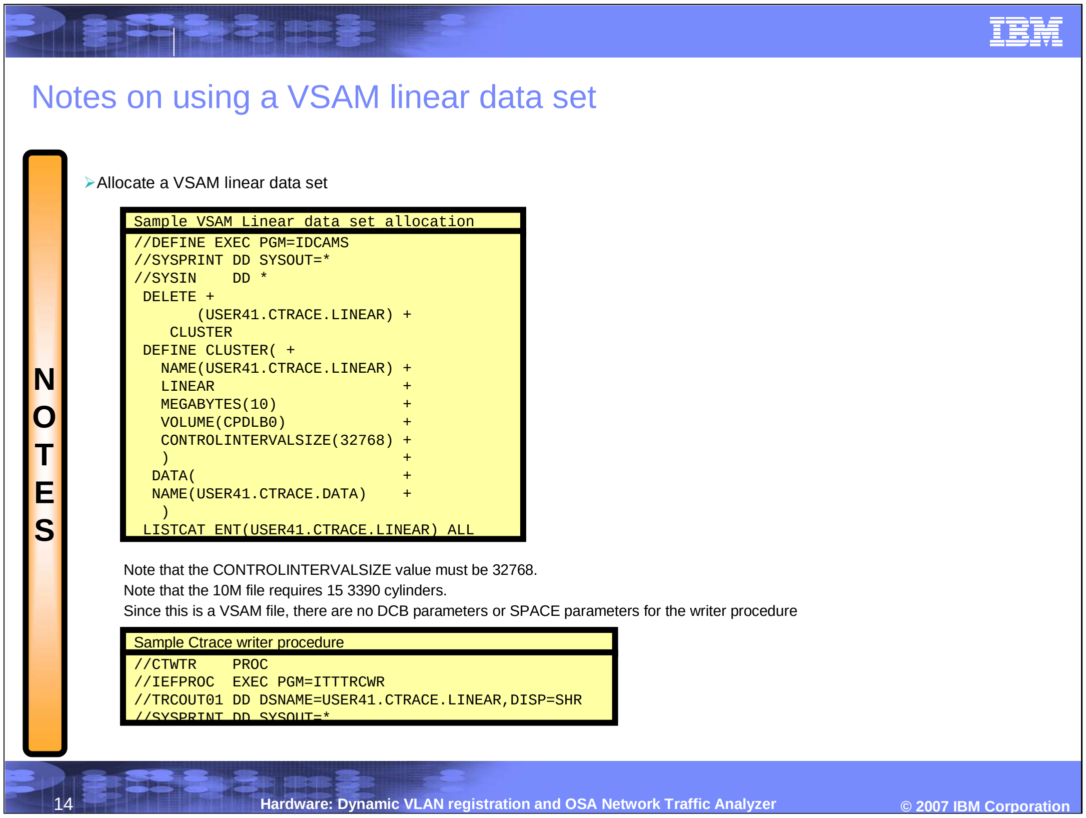
z/OS CTRACE details

- **SYSTCPOT**
  - A new Ctrace component for collecting NTA trace data

- **CTINTA00**
  - Only SYS1.PARMLIB member for SYSTCPOT
  - Specify the default buffer size
  - Minimum - 1M, Default - 64M, Maximum - 624M
  - Connect to a CTRACE writer
  - There are no OPTIONS values

- **TRACE CT,ON,COMP=SYSTCPOT,SUB=(tcpipprocname)**
  - Starts the component trace

- **New with z/OS V1R7, the Ctrace writer supports using a VSAM linear data set for fast writing of Ctrace data**
  - Allocate a VSAM linear data set
  - Used in the Ctrace writer procedure
  - Can be read by the IPCS CTRACE subcommand
  - Cannot be sent to IBM service - Use the IPCS COPYTRC subcommand to convert it to sequential file
  - Can be used by any Ctrace component



Notes on using a VSAM linear data set

NOTES

- Allocate a VSAM linear data set

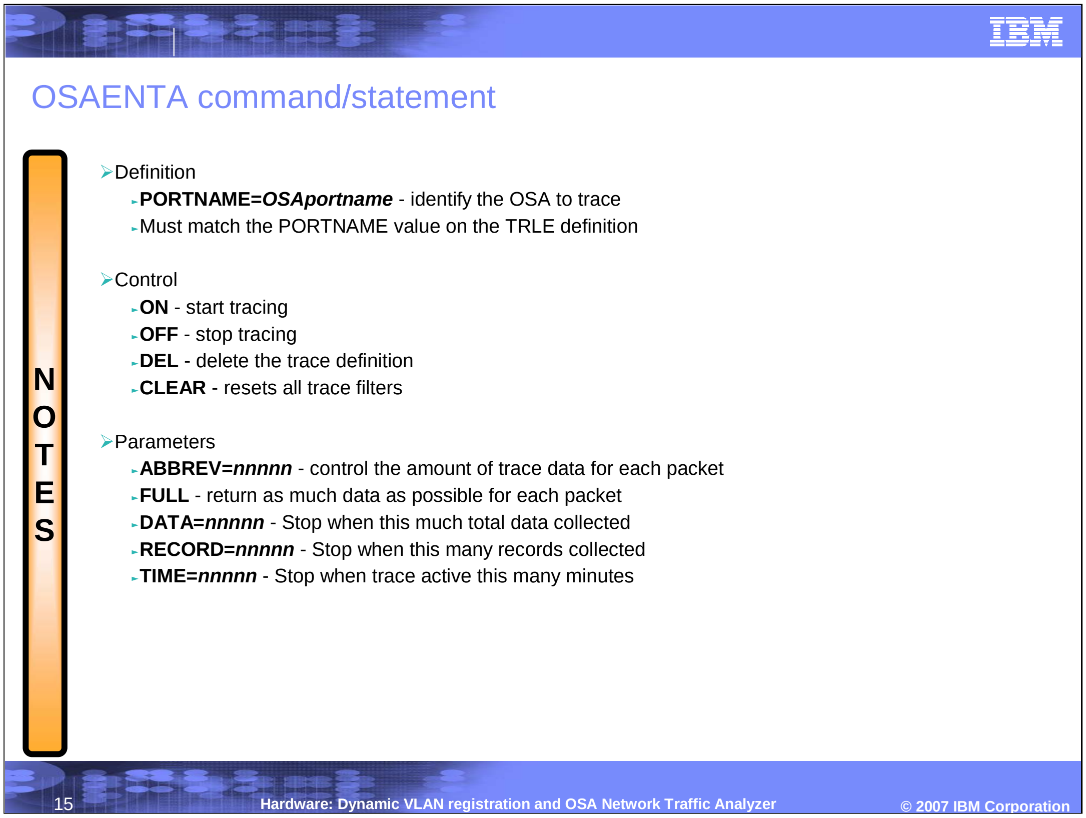
Sample VSAM Linear data set allocation

```
//DEFINE EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN    DD *
 DELETE +
       (USER41.CTRACE.LINEAR) +
    CLUSTER
 DEFINE CLUSTER( +
   NAME(USER41.CTRACE.LINEAR) +
   LINEAR                     +
   MEGABYTES(10)              +
   VOLUME(CPDLB0)             +
   CONTROLINTERVALSIZE(32768) +
   )                          +
  DATA(                       +
  NAME(USER41.CTRACE.DATA)    +
   )
 LISTCAT ENT(USER41.CTRACE.LINEAR) ALL
```

Note that the CONTROLINTERVALSIZE value must be 32768.
Note that the 10M file requires 15 3390 cylinders.
Since this is a VSAM file, there are no DCB parameters or SPACE parameters for the writer procedure

Sample Ctrace writer procedure

```
//CTWTR    PROC
//IEFPROC  EXEC PGM=ITTTRCWR
//TRCOUT01 DD DSNAME=USER41.CTRACE.LINEAR,DISP=SHR
//SYSPRINT DD SYSOUT=*
```



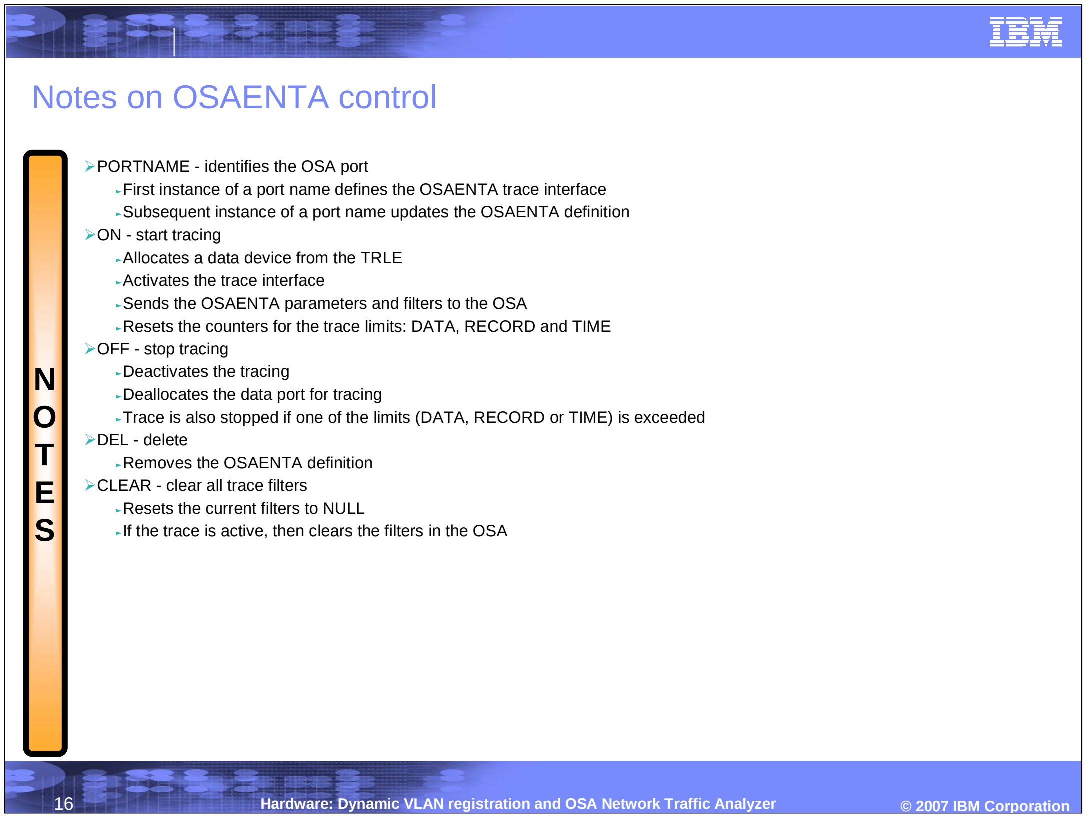
OSAENTA command/statement

NOTES

- Definition
  - **PORTNAME=*OSAportname*** - identify the OSA to trace
  - Must match the PORTNAME value on the TRLE definition

- Control
  - **ON** - start tracing
  - **OFF** - stop tracing
  - **DEL** - delete the trace definition
  - **CLEAR** - resets all trace filters

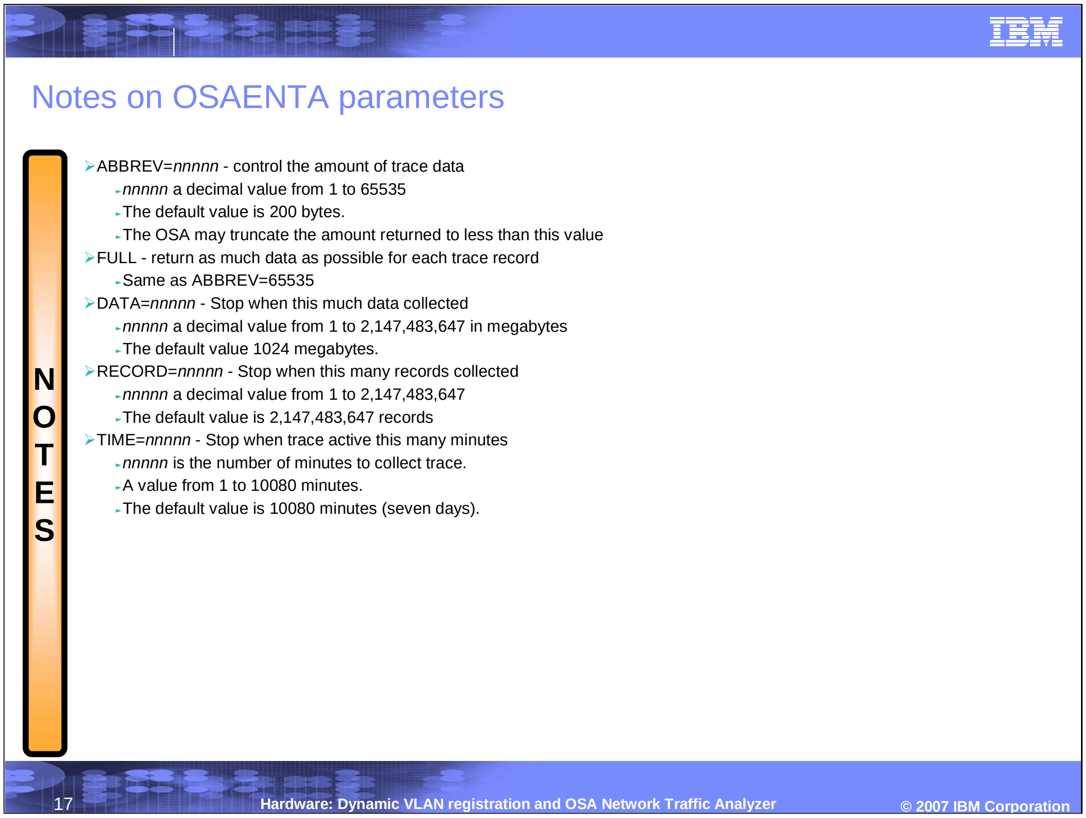
- Parameters
  - **ABBREV=*nnnnn*** - control the amount of trace data for each packet
  - **FULL** - return as much data as possible for each packet
  - **DATA=*nnnnn*** - Stop when this much total data collected
  - **RECORD=*nnnnn*** - Stop when this many records collected
  - **TIME=*nnnnn*** - Stop when trace active this many minutes



Notes on OSAENTA control

NOTES

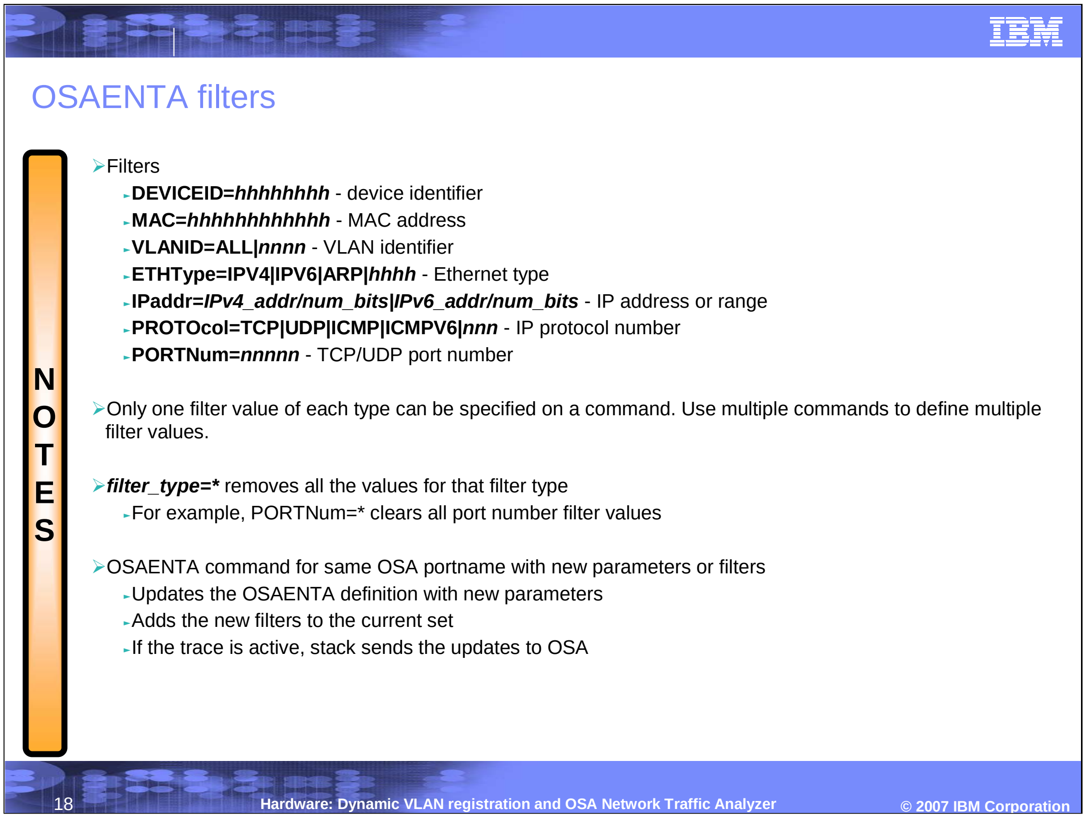
- PORTNAME - identifies the OSA port
  - First instance of a port name defines the OSAENTA trace interface
  - Subsequent instance of a port name updates the OSAENTA definition
- ON - start tracing
  - Allocates a data device from the TRLE
  - Activates the trace interface
  - Sends the OSAENTA parameters and filters to the OSA
  - Resets the counters for the trace limits: DATA, RECORD and TIME
- OFF - stop tracing
  - Deactivates the tracing
  - Deallocates the data port for tracing
  - Trace is also stopped if one of the limits (DATA, RECORD or TIME) is exceeded
- DEL - delete
  - Removes the OSAENTA definition
- CLEAR - clear all trace filters
  - Resets the current filters to NULL
  - If the trace is active, then clears the filters in the OSA



Notes on OSAENTA parameters

NOTES

- ABBREV=*nnnnn* - control the amount of trace data
  - *nnnnn* a decimal value from 1 to 65535
  - The default value is 200 bytes.
  - The OSA may truncate the amount returned to less than this value
- FULL - return as much data as possible for each trace record
  - Same as ABBREV=65535
- DATA=*nnnnn* - Stop when this much data collected
  - *nnnnn* a decimal value from 1 to 2,147,483,647 in megabytes
  - The default value 1024 megabytes.
- RECORD=*nnnnn* - Stop when this many records collected
  - *nnnnn* a decimal value from 1 to 2,147,483,647
  - The default value is 2,147,483,647 records
- TIME=*nnnnn* - Stop when trace active this many minutes
  - *nnnnn* is the number of minutes to collect trace.
  - A value from 1 to 10080 minutes.
  - The default value is 10080 minutes (seven days).



IBM

OSAENTA filters

NOTES

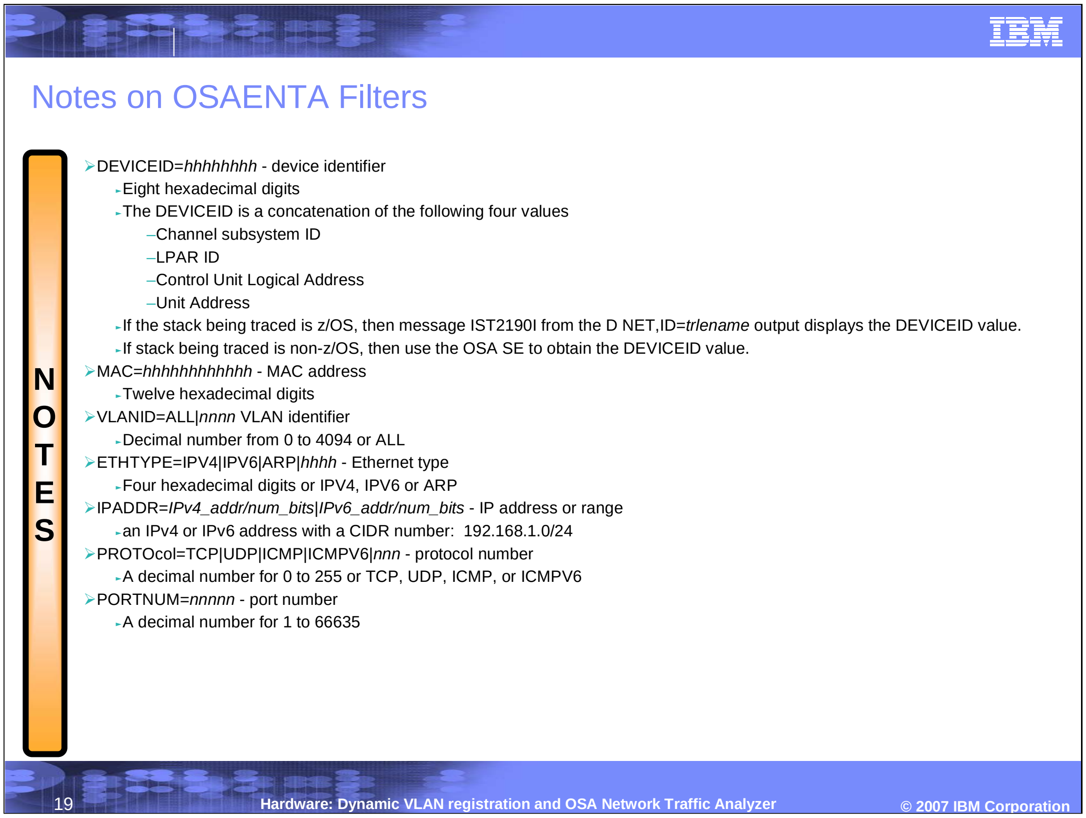
- Filters
  - **DEVICEID=*hhhhhhhh*** - device identifier
  - **MAC=*hhhhhhhhhhhh*** - MAC address
  - **VLANID=ALL|*nnnn*** - VLAN identifier
  - **ETHType=IPV4|IPV6|ARP|*hhhh*** - Ethernet type
  - **IPaddr=*IPv4_addr/num_bits|IPv6_addr/num_bits*** - IP address or range
  - **PROTOcol=TCP|UDP|ICMP|ICMPV6|*nnn*** - IP protocol number
  - **PORTNum=*nnnnn*** - TCP/UDP port number

- Only one filter value of each type can be specified on a command. Use multiple commands to define multiple filter values.

- ***filter_type*=*** removes all the values for that filter type
  - For example, PORTNum=* clears all port number filter values

- OSAENTA command for same OSA portname with new parameters or filters
  - Updates the OSAENTA definition with new parameters
  - Adds the new filters to the current set
  - If the trace is active, stack sends the updates to OSA

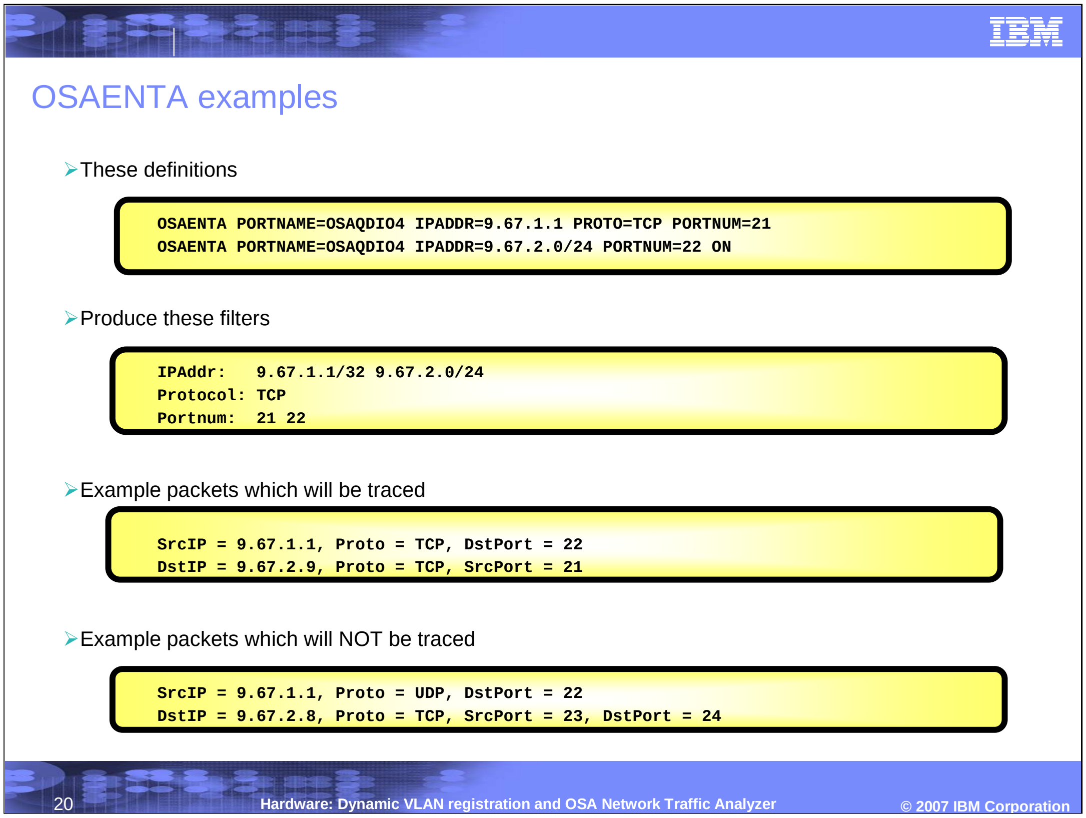
18 Hardware: Dynamic VLAN registration and OSA Network Traffic Analyzer © 2007 IBM Corporation



Notes on OSAENTA Filters

NOTES

- DEVICEID=*hhhhhhhh* - device identifier
  - Eight hexadecimal digits
  - The DEVICEID is a concatenation of the following four values
    - Channel subsystem ID
    - LPAR ID
    - Control Unit Logical Address
    - Unit Address
  - If the stack being traced is z/OS, then message IST2190I from the D NET,ID=*trlename* output displays the DEVICEID value.
  - If stack being traced is non-z/OS, then use the OSA SE to obtain the DEVICEID value.
- MAC=*hhhhhhhhhhhh* - MAC address
  - Twelve hexadecimal digits
- VLANID=ALL|*nnnn* VLAN identifier
  - Decimal number from 0 to 4094 or ALL
- ETHTYPE=IPV4|IPV6|ARP|*hhhh* - Ethernet type
  - Four hexadecimal digits or IPV4, IPV6 or ARP
- IPADDR=*IPv4_addr/num_bits*|*IPv6_addr/num_bits* - IP address or range
  - an IPv4 or IPv6 address with a CIDR number: 192.168.1.0/24
- PROTOcol=TCP|UDP|ICMP|ICMPV6|*nnn* - protocol number
  - A decimal number for 0 to 255 or TCP, UDP, ICMP, or ICMPV6
- PORTNUM=*nnnnn* - port number
  - A decimal number for 1 to 66635



OSAENTA examples

- These definitions

```
OSAENTA PORTNAME=OSAQDIO4 IPADDR=9.67.1.1 PROTO=TCP PORTNUM=21
OSAENTA PORTNAME=OSAQDIO4 IPADDR=9.67.2.0/24 PORTNUM=22 ON
```

- Produce these filters

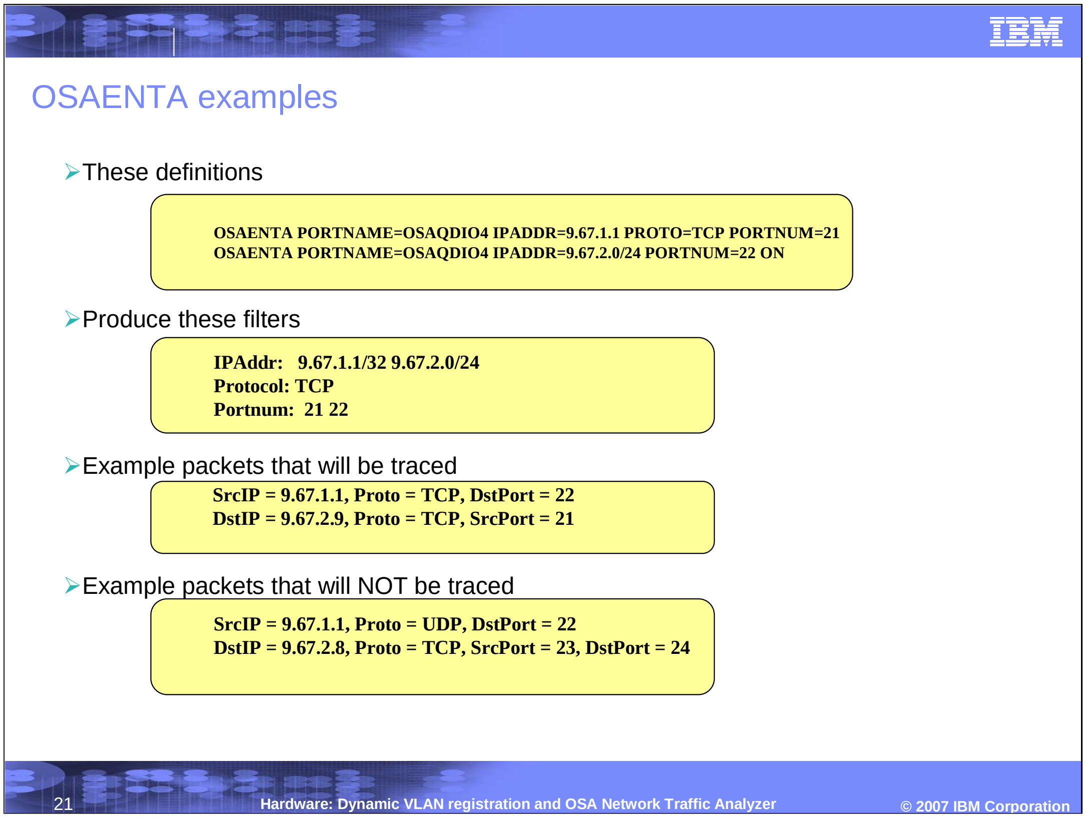
```
IPAddr:   9.67.1.1/32 9.67.2.0/24
Protocol: TCP
Portnum:  21 22
```

- Example packets which will be traced

```
SrcIP = 9.67.1.1, Proto = TCP, DstPort = 22
DstIP = 9.67.2.9, Proto = TCP, SrcPort = 21
```

- Example packets which will NOT be traced

```
SrcIP = 9.67.1.1, Proto = UDP, DstPort = 22
DstIP = 9.67.2.8, Proto = TCP, SrcPort = 23, DstPort = 24
```



OSAENTA examples

- These definitions

```
OSAENTA PORTNAME=OSAQDIO4 IPADDR=9.67.1.1 PROTO=TCP PORTNUM=21
OSAENTA PORTNAME=OSAQDIO4 IPADDR=9.67.2.0/24 PORTNUM=22 ON
```

- Produce these filters

```
IPAddr:   9.67.1.1/32 9.67.2.0/24
Protocol: TCP
Portnum:  21 22
```

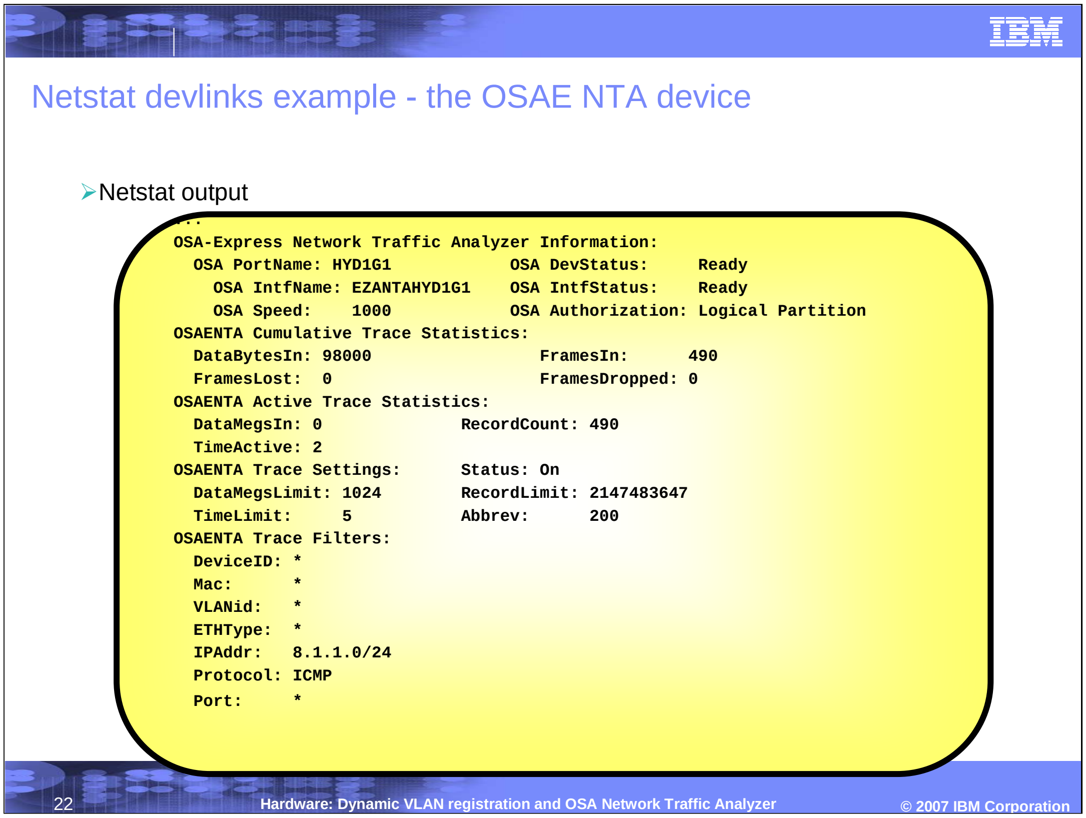
- Example packets that will be traced

```
SrcIP = 9.67.1.1, Proto = TCP, DstPort = 22
DstIP = 9.67.2.9, Proto = TCP, SrcPort = 21
```

- Example packets that will NOT be traced

```
SrcIP = 9.67.1.1, Proto = UDP, DstPort = 22
DstIP = 9.67.2.8, Proto = TCP, SrcPort = 23, DstPort = 24
```

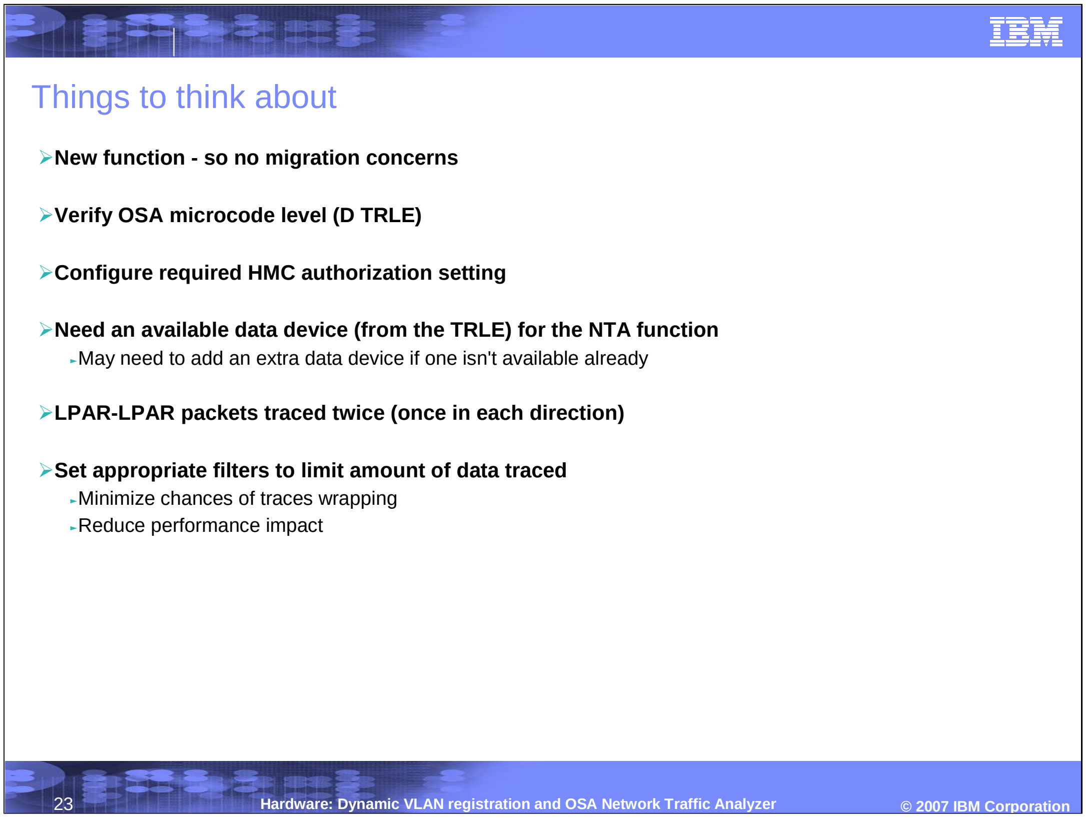
21 Hardware: Dynamic VLAN registration and OSA Network Traffic Analyzer © 2007 IBM Corporation



Netstat devlinks example - the OSAE NTA device

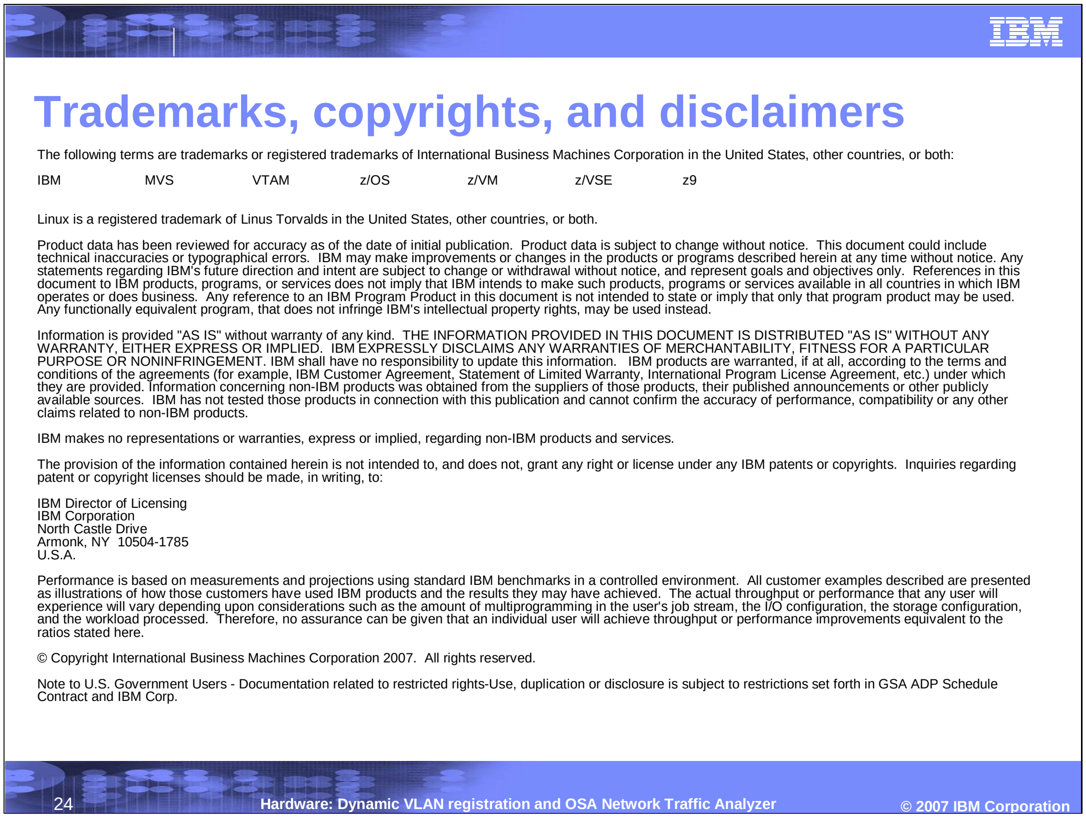
- Netstat output

```
...
OSA-Express Network Traffic Analyzer Information:
  OSA PortName: HYD1G1            OSA DevStatus:     Ready
    OSA IntfName: EZANTAHYD1G1    OSA IntfStatus:    Ready
    OSA Speed:    1000            OSA Authorization: Logical Partition
OSAENTA Cumulative Trace Statistics:
  DataBytesIn: 98000                 FramesIn:      490
  FramesLost:  0                     FramesDropped: 0
OSAENTA Active Trace Statistics:
  DataMegsIn: 0              RecordCount: 490
  TimeActive: 2
OSAENTA Trace Settings:      Status: On
  DataMegsLimit: 1024        RecordLimit: 2147483647
  TimeLimit:     5           Abbrev:      200
OSAENTA Trace Filters:
  DeviceID: *
  Mac:      *
  VLANid:   *
  ETHType:  *
  IPAddr:   8.1.1.0/24
  Protocol: ICMP
  Port:     *
```



Things to think about

- **New function - so no migration concerns**
- **Verify OSA microcode level (D TRLE)**
- **Configure required HMC authorization setting**
- **Need an available data device (from the TRLE) for the NTA function**
  - May need to add an extra data device if one isn't available already
- **LPAR-LPAR packets traced twice (once in each direction)**
- **Set appropriate filters to limit amount of data traced**
  - Minimize chances of traces wrapping
  - Reduce performance impact



Trademarks, copyrights, and disclaimers

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

IBM MVS VTAM z/OS z/VM z/VSE z9

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Product data has been reviewed for accuracy as of the date of initial publication. Product data is subject to change without notice. This document could include technical inaccuracies or typographical errors. IBM may make improvements or changes in the products or programs described herein at any time without notice. Any statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only. References in this document to IBM products, programs, or services does not imply that IBM intends to make such products, programs or services available in all countries in which IBM operates or does business. Any reference to an IBM Program Product in this document is not intended to state or imply that only that program product may be used. Any functionally equivalent program, that does not infringe IBM's intellectual property rights, may be used instead.

Information is provided "AS IS" without warranty of any kind. THE INFORMATION PROVIDED IN THIS DOCUMENT IS DISTRIBUTED "AS IS" WITHOUT ANY WARRANTY, EITHER EXPRESS OR IMPLIED. IBM EXPRESSLY DISCLAIMS ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NONINFRINGEMENT. IBM shall have no responsibility to update this information. IBM products are warranted, if at all, according to the terms and conditions of the agreements (for example, IBM Customer Agreement, Statement of Limited Warranty, International Program License Agreement, etc.) under which they are provided. Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products in connection with this publication and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products.

IBM makes no representations or warranties, express or implied, regarding non-IBM products and services.

The provision of the information contained herein is not intended to, and does not, grant any right or license under any IBM patents or copyrights. Inquiries regarding patent or copyright licenses should be made, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. All customer examples described are presented as illustrations of how those customers have used IBM products and the results they may have achieved. The actual throughput or performance 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 or performance improvements equivalent to the ratios stated here.

Note to U.S. Government Users - Documentation related to restricted rights-Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract and IBM Corp.