Communications Server z/OS V1R5 and V1R6 Technical Update

zOS CS Security: Multi Level Security (MLS) Introduction

Topics

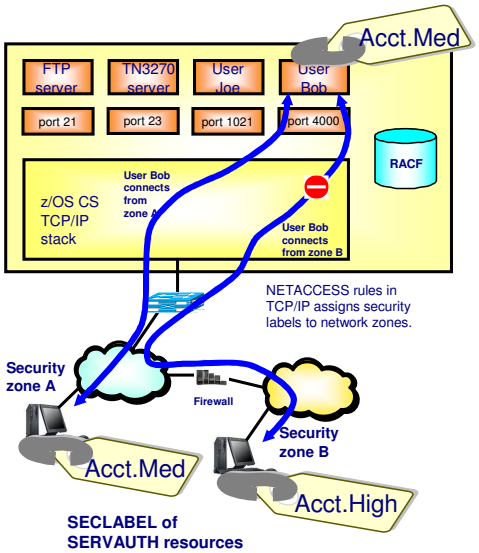
- Multi Level Security (MLS)
  - Introduction to MLS
  - MLS in z/OS V1R5
  - MLS in z/OS V1R6

Multilevel security

- Multilevel Security is an enhanced security environment that can be configured on z/OS
  - Extends the B1 security support
  - IBM's MLS for z/OS has access control implications for the entire system.
    - See z/OS Planning for Multilevel Security GA22-7509 for system-wide MLS information
  - z/OS Communications Server TCP/IP is one element of a multilevel secure z/OS system
- Goal of MLS is to prevent declassification of data
  - All data and other resources are classified
  - All users are classified
- Classification is accomplished with Security Labels which combine
  - Security levels (hierarchical)
    - e.g. Top Secret, Internal Use Only, Unclassified
  - Security categories (non-hierarchical)
    - e.g. Accounting, Sales
- MLS adds a security policy check, Mandatory Access Control (MAC), to the usual Discretionary Access Control (DAC)
  - MAC ensures that data of a certain classification is accessed by a user with authority to access that classification
    - With MAC, the security administrator using RACF, classifies the sensitivity and type of each resource using a Security Label and controls each user's access to the resource by assigning a Security Label to the user.
  - DAC ensures that data can be accessed only by a user permitted to access the data
    - With DAC, user-based permission to access resources
- MAC check is made prior to DAC check

CS z/OS TCP/IP support of MLS

- **Phase 1**
  - Network Access Control outbound (OS/390 V2R10) and inbound (z/OS V1R4)
    - RACF SERVAUTH profiles define IP security zones that local users may/may not be permitted to send to/receive from over a socket (IP addresses are defined within a zone)
- **Phase 2 (z/OS V1R5)**
  - Extend Network Access Control to an MLS environment
    - When data is sent or received, the TCP/IP stack determines if the end user's security label matches the network zone security label (user SECLABEL compared to SERVAUTH zone SECLABEL)
    - Additionally, IP packets that are sent over XCF or SAMEHOST are labeled. If a label is present in an IP packet, that label is used to check against the user's label.
  - Netacess and MLS extended to applications
    - Port of entry (PoE) checks for the FTP server (MLS and NonMLS)
      - Checks ensure a client may logon to FTP
    - Netacess controls for TN3270 server ports (nonMLS)
      - Client Netacess permitted to TN3270 port zone
    - MLS controls determine if client has equivalent SECLABEL to the TN3270 LUname selected
- **Phase 3 (z/OS V1R6)**
  - Examine networking applications to determine if they will run successfully in an MLS environment.
  - Restrict IPv4 Setsockopt() source routing options
  - Documentation of MLS issues for applications/networking



Understanding z/OS security Concepts

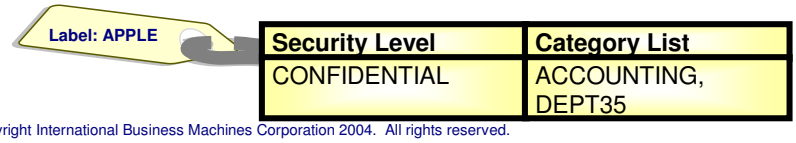
- **Resource Classes**
  - System resource managers that manage access to data, network and other resources are assigned to resource classes by the security server.
  - The SERVAUTH class is used to control general resources owned by many products. TCP/IP has been assigned to the SERVAUTH class.
    - When the RACF option MLACTIVE is set, all profiles in the SERVAUTH class must have a security label defined.
- **Resource Names**
  - Each system resource manager constructs resource names that represent the resources it wants to control access for. All resources defined by TCP/IP have a high level qualifier of EZA or EZB. Some TCP/IP resource names:
    - Authority to use a particular stack is represented by the **EZB.STACKACCESS.*systemname.stackname*** resource.
    - Authority to use the Fast Response Cache Accelerator is represented by the **EZB.FRCAACCESS.*systemname.stackname*** resource.
    - Authority to bind a socket to a port is represented by the **EZB.PORTACCESS.*systemname.stackname.safname*** resource.
    - Authority to use an IP address is represented by the **EZB.NETACCESS.*systemname.stackname.zonename*** resource.
- **Profile Names**
  - The security administrator configures profiles for resources.
    - Wildcard characters allow a profile to control multiple resources.
    - UserIds are permitted to profiles with a given access authority.

Understanding z/OS security Concepts - *continued*

- **Port of Entry**
  - This is information that identifies the origin of work entering the system. It may be the LU name of a TERMINAL, or LU6.2 peer or the IPv4 address of the client program.
  - Starting in z/OS V1R5 it may be a SERVAUTH resource name.
    - Resource managers may query a socket for the TCP/IP NETACCESS resource name covering the peer IPv4 or IPv6 address.
- **Accessor Environment Element (MVS ACEE control block)**
  - This is the z/OS security control block that is associated with every address space. Authorized programs may also associate different ACEEs with specific TASKs or THREADS.
  - Authentication
    - ACEEs are built by the security server on behalf of system resource managers when they AUTHENTICATE a user at login or otherwise associate an identity with a unit of work.
    - Port of Entry may be used to limit source of login and to set session attributes.
  - Authorization
    - System resource managers are responsible for calling the security product with a current ACEE to check the user's authorization when accessing resources on behalf of the user.

Understanding z/OS security Concepts - *continued*

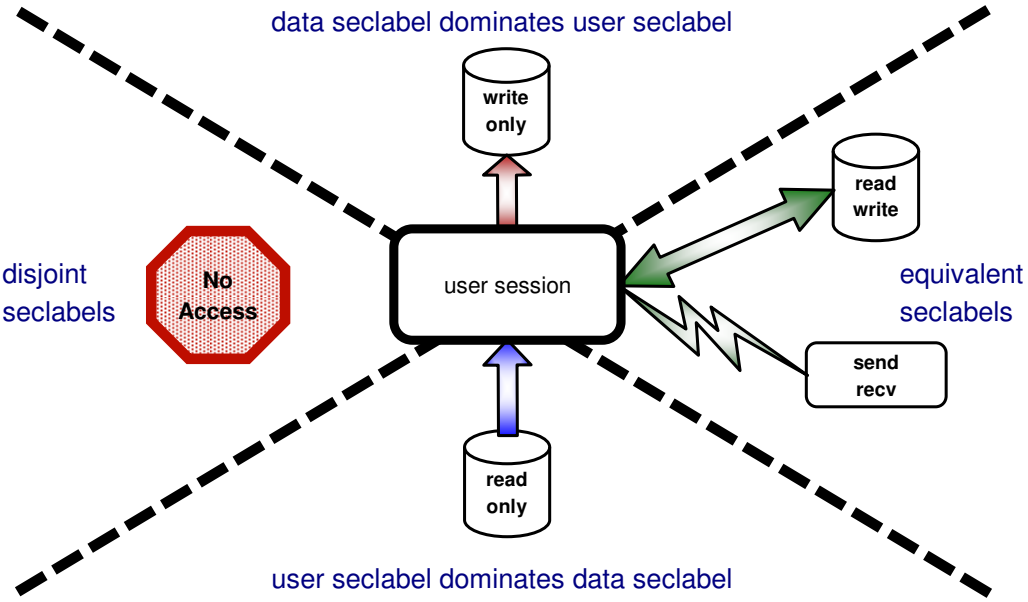
- **Security Level**
  - This term deals with the sensitivity of information and a person's clearance to it. Information is classified according to its sensitivity, such as CONFIDENTIAL or SECRET or TOPSECRET, etc.
  - Users are classified by their clearances.
- **Category**
  - This term is used to designate the department or type of information.
  - There might be a category for accounting, another for logistics and another for cryptographic methodology. There might be categories created for certain products or projects.
  - Categories are used to enforce broad "need to know" policies.
- **Security Label**
  - A security label (seclabel) is an eight character name. It represents a particular security level and a set of categories that are defined in the security server.
  - Port of Entry may be used to set or limit session security label at login.
  - Job card parameter SECLABEL= may be used to set job security label.



Understanding z/OS security Concepts - continued

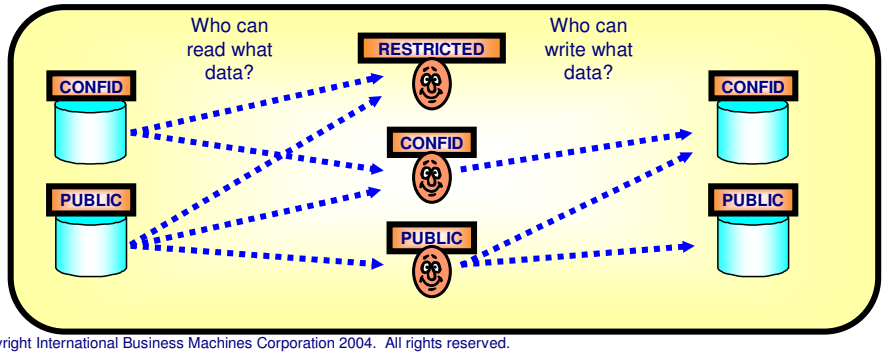
- **Equal, Equivalent, Dominant and Disjoint SECLABELs**
  - These are the seclabel checks done to insure that access to information is authorized.
    - **Disjoint:** If both seclabels have one or more categories that are not present in the other seclabel, they are said to be disjoint. No access is allowed.
    - **Dominate:** One seclabel is said to dominate another one, when its level is equal or higher and its categories are equal or a proper superset of the other.
    - **Equivalent:** Two seclabels are equivalent when their names are defined to have the same level and identical categories. Equivalent seclabels dominate each other.
    - **Equal:** Two seclabels are equal when they have the same name. Equal seclabels may be considered equivalent without asking the security server.
  - To READ data the user's seclabel must dominate the data seclabel.
  - To WRITE data the the user's seclabel must be dominated by the data seclabel.
  - To both READ and WRITE data, the user and data seclabels must be equivalent.
- **There are some predefined seclabels with special meanings:**
  - SYSLOW
    - This label is dominated by all other labels. It can be read by anyone. Distributed software is SYSLOW.
  - SYSHIGH
    - This label dominates all other labels. It can be written by anyone. System console and syslogd files are SYSHIGH. So are dumps and traces.
  - SYSNONE
    - This label is immune from seclabel checking. Useful for system catalog. (Intended for resources only, not users.)
  - SYSMULTI
    - This label is equivalent to all other labels. Given to authorized servers that run work securely on behalf of multiple users.

Understanding z/OS security Concepts - continued



Data and User security labels (simplified!)

- A user can read a file that has a lower or equivalent security label
  - User security label dominates data security label
- A user can write (only) to a file that has an equivalent or higher security label
  - Data security label dominates user security label
- A user can read and write a file that has an equivalent security label
- If a user could write to a file with a lower security label, that user could in fact declassify data by reading data classified at that user's level and rewriting it at a lower level for other lower-level users to read it.
- Data can "flow" upward in the classification hierarchy, it cannot flow downward.
  - A CONFID user can read PUBLIC data and write it as CONFID or higher
  - A CONFID user can READ CONFID data, but cannot write it is as PUBLIC data



Understanding z/OS security Concepts - continued

- TCP/IP resources are inherently READ/WRITE.
  - User seclabels must always be equivalent to resource seclabels.
  - SERVAUTH class is marked EQUALMAC by the security server.

- The following table demonstrates the result of EQUALMAC processing with various user and resource seclabel combinations:

		Resource seclabel				
		SYSNONE	SYSMULTI	SYSHIGH	SYSLOW	SPECIFIC
User seclabel	SYSNONE	should not occur	should not occur	should not occur	should not occur	should not occur
	SYSMULTI	always	always	always	always	always
	SYSHIGH	always	always	always	never	never
	SYSLOW	always	always	never	always	never
	SPECIFIC	always	always	never	never	if equivalent

Before you enter any further into MLS - please read: z/OS Planning for Multilevel Security GA22-7509

Security labels and TCP/IP networking

- In the networking environment, the information that is being protected is the data being read and written through sockets.
  - Sockets are opened and used by applications running under USERIDs.
- In a z/OS multilevel secure environment:
  - Each USERID may be permitted to use one or more security labels.
  - Every job or login session is associated with a USERID.
  - A USERID can use only one security label for each job or login session.
  - The security label used is limited by the port of entry (source type and location) of the job or login session.
- TCP/IP may be configured to participate in the z/OS V1R5 multilevel secure environment.
  - The TCP/IP support is primarily based on StackAccess and NetworkAccess.
  - StackAccess verifies the security label of users of a stack.
  - NetworkAccess verifies the security label of information flowing to/from an IP address.
- Applications on a multilevel secure system may then securely communicate with applications on other systems.
- The packets being sent from a single IP address on the multilevel secure system may have originated from applications running under different security labels. Other systems cannot normally associate a single security label with IP addresses owned by a multilevel secure system.
- When two applications on multilevel secure systems communicate, the receiving system enforces MAC prior to delivering the information to an application. The security label of the sending application must be communicated to the receiving system:
  - Implicitly by security zone of source or destination IP
  - or explicitly by packet tag

Understanding zOS CS TCP/IP Stacks on Multilevel Secure Systems

- z/OS CS TCP/IP stack running in a z/OS multilevel secure environment may optionally be configured as either a restricted stack or an unrestricted stack.
  - Restricted stack job runs under userid with a specific security label.
  - Unrestricted stack job runs under userid with SYSMULTI security label.
- A single z/OS system may concurrently run up to eight z/OS CS TCP/IP stacks.
  - These may be any mix of restricted and unrestricted stacks.
- Restricted Stacks
  - The stack ensures that all sockets are opened by applications running with a security label that is equivalent to the security label of that stack.
  - The stack also ensures that all information received from the network and delivered to an application is equivalent to the stack's security label.
- Unrestricted Stacks
  - The stack allows sockets to be opened by applications with any security label.
  - The stack supports MAC processing that allows its applications to communicate securely with any other single level secure system or restricted stack.
  - The stack transmits security labels in a proprietary format to other z/OS TCP/IP unrestricted stacks over XCF or IUTSAMEHOST links within the same sysplex.

Understanding Stack Recognition of a Multilevel Secure Environment

- Issue RACF command SETROPTS MLACTIVE.
  - Requires SETROPTS CLASSACT(SECLABEL)
  - and SETROPTS RACLIST(SECLABEL)
- Place NetAccess statement(s) in the TCPIP PROFILE
  - Initial stack startup or Vary Obey command
  - SETROPTS MLACTIVE must occur before first NetAccess statement is processed or stack must be re-cycled to recognize multilevel secure environment.
- When you start several TCP/IP stacks under OMVS, you are using the Common INET PFS.
  - There may be a mix of Restricted and Unrestricted stacks on a system.
  - StackAccess may be used to limit the subset of stacks a user session or job has access to.
  - Users and jobs may optionally establish affinity to a single stack or they may allow Common INET to choose a stack from the subset the user has StackAccess to.

Understanding Network Security Zones

- A network security zone is an administrative name for a collection of systems that require the same access control policy.
- IP addresses are used to map systems into security zones.
- Single level secure systems must be in a security zone with the same security label as their subnet broadcast addresses (host portion of address all 1s or all 0s)
  - Typically use subnet scope security zones.
- A "trusted" subnet contains only multilevel secure systems and SYSHIGH single level secure systems
  - Multilevel secure systems may be Unrestricted or Restricted with any security label.
  - These single level secure systems are typically administrator PCs.
  - Typically use SYSHIGH subnet scope security zone with individual addresses in different security zones.
- z/OS systems that are not configured for multilevel security or that run TCP/IP stacks that are not configured for multilevel security. Must be treated as untrusted single level secure systems
  - Single security label for all data and users
  - Externally managed network security
- z/OS systems that are configured for multilevel security and that run TCP/IP stacks that are configured for multilevel security. May be treated as trusted multilevel secure systems
  - Multiple security labels for data and users
  - Self-managed network security

MLS consistency cross check

MLS configuration complexity

- Secure communication in a multilevel secure environment is complex.
- It requires configuration in multiple places, often involving several people:
  - statements in the TCPIP.PROFILE
    - real interface addresses
    - virtual interface addresses
    - netaccess security zones
  - security server resource profiles in several classes:
    - SERVAUTH
      - EZB.STACKACCESS...
      - EZB.NETACCESS...
    - SECDATA
    - SECLABEL
    - STARTED
      - STDATA segment associates user ID with procedure.job name
    - USER
      - default SECLABEL
  - Implications of RACF Options - SETROPTS
    - RACLIST(SECLABEL)
    - MLACTIVE
    - SECLBYSYSTEM
    - MLS
    - MLSTABLE
    - MLQUIET

MLS sequencing restrictions

- There are sequence restrictions on starting a TCP/IP stack in a multilevel secure environment:
  - SECLABEL class must be active.
  - Stack job must be started with a security label.
  - SETROPTS MLACTIVE must be set prior to the stack processing the first NetAccess statement in TCPIP.PROFILE or VARY TCPIP,,OBEYFILE command file.
  - Cycling SETROPTS MLACTIVE to NOMLACTIVE and back to MLACTIVE requires the stack to be stopped and restarted.
  - Restrictions on the sequence of enabling portions of the environment and starting TCP/IP stacks are confusing.
- Coordination and consistency of a multilevel security system configuration can be a difficult administrative task.
- Inconsistencies in this configuration can allow unintended communication or prevent intended communication. Running a stack with an inconsistent configuration in a production environment can compromise data security.
- Changes to a resource's security label must be preventable while running production workloads.
- Determining the network security zone and security label associated with a given IP address can be tedious.

Remove most sequence restrictions in z/OS V1R6

- Dynamic changes to RACF MLACTIVE option is supported:
  - SETROPTS NOMLACTIVE or MLACTIVE may be done at any time without restarting stack.
  - Stack detection of this change is dependent on
    - RACF ENF signal following SETROPTS RACLIST of SERVAUTH or SECLABEL class
    - VARY TCPIP,,OBEYFILE command

- Dynamic changes to RACF profiles are supported:
  - SERVAUTH and SECLABEL classes may be
    - activated,
    - deactivated,
    - modified and refreshed while the stack is running.
  - Stack detection of these changes is dependent on RACF RACLIST ENF signal.
    - With other security servers, a VARY TCPIP,,OBEYFILE command to replace the NETACCESS statement may be required.

- NetAccess statement may be processed before or after SETROPTS MLACTIVE.
  - Initial TCPIP.PROFILE
  - VARY TCPIP,, OBEYFILE command file

- These restrictions on starting TCP/IP in a multilevel secure environment remain:
  - SECLABEL class must be active when stack job is started.
  - Stack job must be started with a security label.

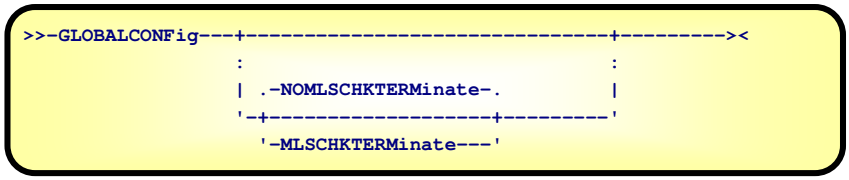
Automatic Consistency Check added in z/OS V1R6

- Every stack running on a system with the RACF option MLACTIVE does an internal consistency check on several TCPIP.PROFILE statements and their associated SERVAUTH profiles.

- This consistency checking occurs
  - at the end of TCPIP.PROFILE processing:
    - after initial TCPIP.PROFILE processing,
    - after the VARY TCPIP,,OBEYFILE,*dataset* command modifies the profile
    - after Sysplex statements that are deferred to a second pass of profile processing
    - when OMPROUTE implicitly modifies the BSD Routing Parms for an interface
  - and whenever RACF sends an ENF signal indicating that SETROPTS RACLIST was issued for the SERVAUTH or SECLABEL class.

- The stack may optionally be configured to terminate itself when consistency checking fails.
  - Dynamic changes to mandatory access controls are strongly discouraged when production workloads are running.
  - Use of RACF options (SETROPTS MLSTABLE, SETROPTS MLQUIET) to prevent changes to mandatory access controls during production is recommended.

Production Workload Protection



- NOMLSCHKTERMINATE
  - Specifies that the stack should remain active after writing an informational message when inconsistent configuration information is discovered in a multilevel-secure environment.
- MLSCHKTERMINATE
  - Specifies that the stack should be terminated after writing an informational message when inconsistent configuration information is discovered in a multilevel-secure environment.
- By default, the stack will continue running when inconsistencies are found.
  - It is recommended that you override this default by specifying GLOBALCONFIG MLSCHKTERMINATE in the TCPIP.PROFILE or in a VARY TCPIP,,OBEYFILE command before starting production workloads.
  - Before making security related configuration changes, it is recommended that you first stop all production workloads. You may then specify GLOBALCONFIG NOMLSCHKTERMINATE in the TCPIP.PROFILE or in a VARY TCPIP,,OBEYFILE command.
  - This parameter may only be changed from MLSCHKTERMINATE to NOMLSCHKTERMINATE when the RACF option NOMLSTABLE is set or when both MLSTABLE and MLQUIET are set.

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	CICS	IMS	MQSeries	Tivoli
IBM(logo)	Cloudscape	Informix	OS/390	WebSphere
e(logo)business	DB2	iSeries	OS/400	xSeries
AIX	DB2 Universal Database	Lotus	pSeries	zSeries

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

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

Intel, ActionMedia, LANDesk, MMX, Pentium and ProShare are trademarks of Intel Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Linux is a registered trademark of Linus Torvalds.

Other company, product and service names may be trademarks or service marks of others.

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 and/or changes in the product(s) and/or program(s) 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 (e.g., 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.

© Copyright International Business Machines Corporation 2004. All rights reserved.