zSeries Technical Conference

Running Linux in Less than class G

Rob van der Heij
IBM Netherlands

L84 | May 2004

Trademarks

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

BookManager*
DB2*
DFSMS/MVS*
DFSMS/VM*
e-business logo*
Enterprise Storage Server
ESCON*
FICON
GDDM*
IBM*
IBM logo*
Language Environment*
Multiprise*
MVS
NetRexx
OpenEdition*
OpenExtensions
OS/390*
Parallel Sysplex*
PR/SM
QMF
RACF*
RAMAC*
S/390*
S/390 Parallel Enterprise Server
VisualAge*
VisualGen*
VM/ESA*
VSE/ESA
VTAM*
z/Architecture
z/OS
z/VM
zSeries

The following are trademarks or registered trademarks of other companies.

Lotus, Notes, and Domino are trademarks or registered trademarks of Lotus Development Corporation; LINUX is a registered trademark of Linus Torvalds; Penguin (Tux) compliments of Larry Ewing; Tivoli is a trademark of Tivoli Systems Inc.; Java and all Java-related trademarks and logos are trademarks of Sun Microsystems, Inc., in the United States and other countries; UNIX is a registered trademark of The Open Group in the United States and other countries; Microsoft, Windows and Windows NT are registered trademarks of Microsoft Corporation; SET and Secure Electronic Transaction are trademarks owned by SET Secure Electronic Transaction LLC. * All other products may be trademarks or registered trademarks of their respective companies.

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

Agenda

- What is the problem? And is it yours too?
- CP Privilege Class
- CP Commands for Linux
- Diagnose Codes for Linux
- Various other z/VM Resources
- Conclusion

Past: VM systems in trusted environment

- **VM used to run in protected environment of the computing center**
  - Virtual machine running CMS applications
  - Exposures of the system well understood
  - Access restricted to well-behaving employees
  - Fairly large skilled staff to monitor and manage the system

Present: z/VM running Linux virtual machines

- **z/VM used to run Linux virtual machines**
  - Linux servers probably not managed by VM staff
  - Access not restricted to trusted employees
  - Maybe even connected to the Internet

- **Given enough servers and enough time, one or more may be hacked**
  - Security policy, logging, monitoring
  - Firewalls, proxy servers, multi-tier applications

Linux on z/VM different from discrete servers?

- **Linux virtual machines run on the same z/VM system and share more than just the network**
- **A hacker with root access can eventually make the virtual machine do anything that CP allows it to do**
  - Somewhat like access to your virtual raised floor
- **May impact Linux servers for other customers**
- **Systems Management work and Infrastructure may be affected**

Linux on z/VM different from CMS ?

- **CMS virtual machines are typically single-user**
  - One person for one virtual machine
- **Different audience and different exposures**
- **Linux is using things in a different way**
  - More virtual storage, more disk space
  - New function
- **Linux does not need all function that CMS uses**

The Problem

- **Make sure that a compromised Linux virtual machine does not a form risk for z/VM integrity**
  - Restrict access to z/VM function to the absolute minimum
  - Harden security for the z/VM function that Linux needs
  - Do not cripple the other virtual machines

Issues identified are at worst Denial of Service

Abuse does not provide unauthorized access,
but may restrict others with authorized access

Solving the problem

- **Start with a fairly relaxed scheme (like for CMS)**
  - Try to think of what may hurt and protect that
  - Every time some exploit is discovered, try to repair it
  - May get caught by new function in next release
- **Determine minimum requirements for Linux**
  - Analyze possible exploits or risks for each of these
  - Permit access when no risk observed
    Or find alternative to avoid the exploit

Less Work

CP Privilege Class

- **Framework to control which users can issue which CP commands**
- **Commands are classified by function in groups**
- **Much more granular than root versus non-root**

A	**System Operator**
B	**System Resource Operator**
C	**System Programmer**
D	**Spooling Operator**
E	**System Analyst**
F	**Service Representative**
G	**General User**

CP Privilege Class

- **CP Class for users specified in the CP Directory**
- **Assigned corresponding to their role**
- **General users get class G**
- **Special users get class G plus some more**

```
USER JOHNDOE ******** 16M 32M G
 INCLUDE IBMDFLT
 MDISK 191 3390  101   10 LX3W03  MR
USER RVDHEIJ ******** 16M 2047M   CEG
 INCLUDE IBMDFLT
 MDISK 191 3390    1  100 LX3W03  MR
 MDISK 192 3390 1876  125 LX3L05  MR ALL
```

CP Privilege Class

- **Too restrictive for some installations**
  - One special command comes with a lot more power
- **Redefining Command Privilege Classes**
  - Additional new privilege classes I-Z, 1-6
  - CP commands can be assigned to the new classes

```
MODIFY CMD MSGNOH IBMCLASS B PRIVCLASS BM
```

    - Also as statements in the system configuration file
    - Previously done with User Class Restructure
  - Fine-tuned access to privileged commands

CP Privilege Class

- **Take some privileged commands out of the standard CP class**
  - E.g. move SHUTDOWN to a separate privilege class
  - Sometimes less work than defining a new class
  - Used to be the safe approach with UCR

- **External Security Manager can restrict further**
  - RACF/VM can control access to various resources

CP Commands

- **Linux does not need all power class G provides**
- **Not all can hurt, but a lot of work to investigate**

```
ADJUNCT     ADSTOP      ATTN        BEGIN       CHANGE      CLOSE
COMMANDS    COUPLE      CPFORMAT    CPU         DEFINE      DETACH
DIAL        DISCONNECT  DISPLAY     DUMP        ECHO        EXTERNAL
INDICATE    IPL         LINK        LOADVFCB    LOCATEVM    LOGON
LOGOFF      MESSAGE     NOTREADY    ORDER       PURGE       QUERY
READY       REDEFINE    REQUEST     RESET       RESTART     REWIND
SCREEN      SEND        SET         SIGNAL      SILENTLY    SLEEP
SMSG        SPOOL       SPXTAPE     STOP        STORE       SYSTEM
TAG         TERMINAL    TRACE       TRANSFER    UNCOUPLE    UNDIAL
VDELETE     VINPUT      VMDUMP      XAUTOLOG    XSPOOL
```

CP Commands used by Linux

- **A few CP commands issued by kernel and drivers**

  ```
  QUERY TERMINAL
  TERM CONS 3215         (is default)
  TERM AUTOCR OFF
  IPL                    (only for reboot)
  SET PAGEX ON           (when PFAULT lacks)
  ```

- **Create a new privilege class L with only these commands and use that class instead of G**
- **Applications in Linux could use many more**

CP Commands with privilege class ANY

- **CP commands in ANY available in any class**
- **Most can be moved to class G**
  - Some are needed pre-logon capabilities
  - Possible lockout with SET PRIVCLAS

COMMANDS
DIAL
DISCONNECT
LOGON
LOGOFF
MESSAGE
Q BYUSER
Q COMMAND
Q PRIVCLAS
Q USERS
SET PRIVCLAS
SILENTLY
SLEEP
UNDIAL

CP Commands for Linux

- **We can drastically restrict the commands available to Linux virtual machines**
  - Can be done without impacting other users
  - Should be flexible enough for production servers
  - May be too rigid for development work
- **You do not have to treat all Linux virtual machines the same**
  - Consider to match network access (e.g. public, DMZ)

Diagnose Codes

- **Pseudo instruction that allows a virtual machine to access CP function through a well defined API**
  - z/VM 4.4 defines 81 different diagnose codes
- **IBM Defined Diagnose Codes assigned CP classes**
  - User Class Restructure like with CP commands
  - Almost all are assigned privilege class ANY
    (68 in z/VM 4.4)
  - No impact for existing users when moved to class G

CP Diagnose Codes used by Linux

- **SuSE SLES8 kernel of March 2004**

08	Virtual console function	Control per CP command
44	Voluntary time slice end	
60	Get storage size	
210	Retrieve device info	
214	Pending page release	Can not be disabled
250	Block I/O	Used by diagnose I/O
258	PFAULT macro	For pseudo page fault

CP Diagnose Codes used by Linux (optional)

- **Recent Linux development can exploit more CP function through diagnose codes**

4C	**Generate Accounting Records**	**In cpint-1.1.3**
10	**Release Pages**	**Collaborative Memory Management**
64	**NSS Manipulation**	**Used by xip2 and dcss block device**
DC	**Control Application Monitor Data Collection**	**Used by applmon driver**

CP Diagnose Codes for Linux

- **Possible exploits may be less obvious**
  - Hard to use does not mean hard to abuse
  - E.g. Diagnose 7C Logical Device Support Facility
    Would bypass any fences you have in VM TCP/IP tn3270
- **Consider different privilege classes for different type of Linux virtual machines**
- **Future Linux kernels and applications will be able to use more CP function**

Shared Segments

- **Allows virtual machine to attach shared storage**
  - Shared segment identified by name
  - Linux support maps segment into kernel address space
- **Access is through Diagnose 64**
  - Granted default access depends on definition

SR	**Shared R/O**	**Binaries, Libraries, Data**
EW	**Exclusive R/W**	**Swap space**
SW	**Shared R/W**	

Shared Segments

- **Reasons to restrict access to DCSS**
  - Licensed code or secret data to be read
  - Resource consumption (especially for EW type)
- **Plain z/VM control: NAMESAVE**
  - Per-segment option in the CP directory
  - Provides only one level of access (possibly harmful)

Shared Segments

- **Additional control through ESM like RACF/VM**
  - Possible to audit all segment access
  - Control access to restricted segments

- **Relatively new development for Linux on zSeries**
  - Still need to learn what the proper use is and how to prevent what is not
  - Be aware that Linux DCSS is much bigger than for CMS

Network Access

- **With z/VM 4.4 probably use VSWITCH**
  - Scales better than dedicated OSA devices
    - Less memory resources
    - Cheaper than virtual router
    - No obvious maximum number of connections
  - Supports VLAN to separate traffic

Virtual Networking with z/VM Guest LANs Session V25

- **Looks like a LAN but is not that bad**
  - Does not allow sniffing other’s traffic

Network Access

- **Unrestricted Guest LAN**
  - Allows any VM user to couple to the Guest LAN
- **Restricted Guest LAN**
  - Requires explicit permission (through GRANT)

RESTRICTED
ACCESS

- **Without the COUPLE it makes no difference**
  - CP Directory entry takes care of COUPLE
  - But you need it if you want VLAN support

Network Access

- **Virtual Router may be necessary**
  - When no outboard switch is available with VLAN support
  - For advanced things like bandwidth management
- **Connection between server and virtual router**
  - Separate Guest LAN per customer or group of servers
  - IUCV based point-to-point connection
  - CTC based point-to-point connection
    - May want the virtual router to initiate the COUPLE

Inter-user Communication Vehicle - IUCV

- **Between virtual machines**
  - CMS clients and application servers
  - Point-to-point connection with TCP/IP server
  - Protected by CP directory statements
- **Between CP and a virtual machine**
  - System services (like *MONITOR, *ACCOUNT)
- **Linux use of IUCV is limited**
  - Connection to a virtual router
  - Connecting to legacy applications

Inter-user Communication Vehicle - IUCV

- **Protection is frequently overlooked**
  - IUCV ALLOW and IUCV ANY bypass all protection
  - Default DDR install shows some bad examples
- **Some applications allow screening of connections**
  - This may not prevent illegal open connections: DoS
- **Protect against a Denial of Service**
  - Set MAXCONN as low as possible for Linux
  - Set MAXCONN high enough for trusted servers

Spool Space

- **Linux does not currently use spool files**
  - Linux applications use different means to communicate
  - The Unit Record drivers are not commonly used
- **Possible Denial of Service if Linux used spool**
  - Filling spool space with large spool files
  - Confuse (poorly written) servers with unwanted files
  - Could use RACF/VM to protect spool
- **Without the devices and CP commands: no risks**

Spool Space

- **One exception: the virtual console**
- **Would be nice to spool the virtual console**
  - Without the SPOOL command we can not start spooling
  - If Linux can start spooling, it can also stop or suspend it
- **Logging the secondary console**
  - Could make PROP the secondary console
  - Be aware that Linux could flood the console and create problems

Alternative Approach – Two Phase Startup

- **Define privilege class in the directory as GL**
- **Specify IPL CMS in the directory**
  - Do all you have to do in the PROFILE EXEC
  - Issue a SET PRIVCLAS =L and then IPL Linux
  - Only use resources controlled by privilege class (no SFS)
  - Make sure the 191 can not be modified for next IPL
  - Class L exclude the SET PRIVCLAS to prevent going back
- **Instead of CMS one could use an “IPLer”**
  - Small stand-alone program to issue CP commands and IPL

Conclusion

- **Known issues are all “Denial of Service” type**
  - A DoS can turn into an exposure if it locks the guards out
- **z/VM offers a lot of function to harden the system**
- **You need to make your own compromises (if any)**
  - Understand your environment, your audience and which risks to accept
  - Running hostile Linux servers in an existing environment offers some challenges
- **You’re never done – it always can be done better**