



IBM and Linux: Community Innovation for your Business

## The s390-tools package in a nutshell

Dortmund, 29.04.2009



IBM Deutschland  
Hans-Joachim Picht  
[hans@linux.vnet.ibm.com](mailto:hans@linux.vnet.ibm.com)



# Trademarks

**The following are trademarks of the International Business Machines Corporation in the United States, other countries, or both.**

Not all common law marks used by IBM are listed on this page. Failure of a mark to appear does not mean that IBM does not use the mark nor does it mean that the product is not actively marketed or is not significant within its relevant market.

Those trademarks followed by ® are registered trademarks of IBM in the United States; all others are trademarks or common law marks of IBM in the United States.

For a complete list of IBM Trademarks, see [www.ibm.com/legal/copytrade.shtml](http://www.ibm.com/legal/copytrade.shtml):

\*, AS/400®, e business(logo)®, DBE, ESCO, eServer, FICON, IBM®, IBM (logo)®, iSeries®, MVS, OS/390®, pSeries®, RS/6000®, S/30, VM/ESA®, VSE/ESA, WebSphere®, xSeries®, z/OS®, zSeries®, z/VM®, System i, System i5, System p, System p5, System x, System z, System z9®, BladeCenter®

**The following are trademarks or registered trademarks of other companies.**

Adobe, the Adobe logo, PostScript, and the PostScript logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.

Cell Broadband Engine is a trademark of Sony Computer Entertainment, Inc. in the United States, other countries, or both and is used under license therefrom.

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 trademarks of Microsoft Corporation in the United States, other countries, or both.

Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

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

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

ITIL is a registered trademark, and a registered community trademark of the Office of Government Commerce, and is registered in the U.S. Patent and Trademark Office.

IT Infrastructure Library is a registered trademark of the Central Computer and Telecommunications Agency, which is now part of the Office of Government Commerce.

\* 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 s390-tools package
- \* Contained applications
- \* Whats new with version 1.8.0 & 1.8.1
- \* Shutdown Action Tools
- \* IUCV-Terminal
- \* Chzcrypt / Lszcrypt
- \* Cpuplugd
- \* Change & Display Tools
- \* DASD, Debug & Dump
- \* z/VM related tools
- \* Where to find more Information



## What is the s390-tools package?

- \* s390-tools is a package with a set of user space utilities to be used with the Linux on System z distributions.
- \* It is **the** essential tool chain for Linux on System z
- \* It contains everything from the boot loader to dump related tools for a system crash analysis .
- \* The current version is 1.8.0 and was released in November 2008
- \* **A new version will be available in May 2009**
- \* This software package is contained in all major (and IBM supported) distributions which support s390
  - RedHat Enterprise Linux 4
  - RedHat Enterprise Linux 5
  - SuSE Linux Enterprise Server 10
  - SuSE Linux Enterprise Server 11
- \* Website: <http://www.ibm.com/developerworks/linux/linux390/s390-tools.html>
- \* Feedback: [linux390@de.ibm.com](mailto:linux390@de.ibm.com)

## The Content

- \* chccwdev
- \* chchp
- \* chreipl
- \* chshut
- \* chzcrypt
- \* chzcrypt

CHANGE

- \* lscss
- \* lschp
- \* lsdasd
- \* lsluns
- \* lsqeth
- \* lsreipl
- \* lsshut
- \* lstape
- \* lszcrypt
- \* lszfcp

DISPLAY

- \* dasdfmt
- \* dasdinfo
- \* dasdview
- \* fdasd
- \* tunedasd

DASD

- \* mon\_fsstatd
- \* mon\_procd
- \* ziomon

MONITOR

- \* ip\_watcher
- \* osasmpd
- \* qetharp
- \* qethconf

NETWORK

- \* tape390\_crypt
- \* tape390\_display

TAPE

- \* dbginfo
- \* dumpconf
- \* zfcpdump
- \* zfcpdbf
- \* zgetdump

DUMP & DEBUG

- \* vmconvert
- \* vmcp
- \* vmur

z/VM

- \* cpuplugd
- \* iucvterm

MISC

- \* zipl

BOOT

NEW

## Whats new with version 1.8.0

### \* New tools

- chreipl: Change reipl device settings.
- chshut: Change actions which should be done in case of halt, poff, reboot or panic.
- lsreipl: List information of reipl device.
- lsshut: List actions which will be done in case of halt, poff, reboot or panic.
- ziomon tools: Set of tools to collect data for zfc performance analysis.
- lslns: List available SCSI LUNs depending on adapter or port.
- lszcrypt: Show information about zcrypt devices and configuration.
- chzcrypt: Modify zcrypt configuration

### \* Changes to existing tools

- ip\_watcher: New qeth driver support.
- lscss: Show non I/O subchannels.
- lstape: Add SCSI tape support.
- osasnmpd: New qeth driver support.
- zfcpdump\_v2: Add support for memory holes
- zipl: Support for virtio devices.

### \* Bugfixes

- cpuplugd
- lsdasd
- mon\_statd
- zfcpdump\_v2
- zipl dump tools

NEW

## What to expect in version 1.8.1

### \* New tools:

- lucvterm: A set of applications to provide terminal access via the z/VM Inter-User Communication Vehicle (IUCV).

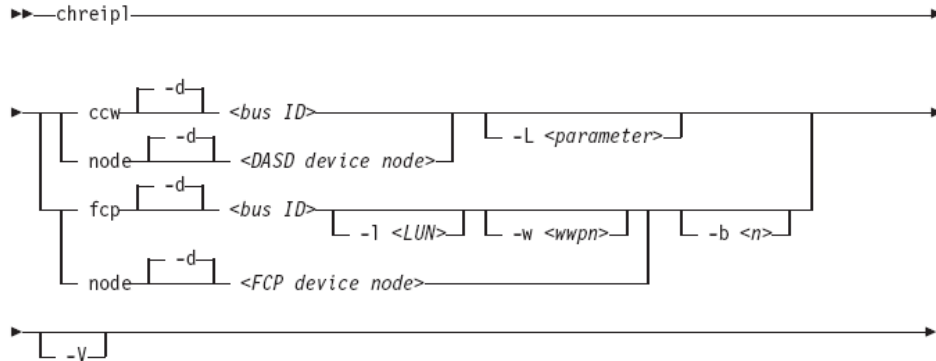
### \* Changes of existing tools:

- dump tools: Add support for "Automatic IPL after dump"
- zipl dump tools: Trigger IPL after dump, if specified.
- zfcpdump: Trigger IPL after dump, if specified.
- dumpconf: Allow to specify "dump\_reipl" in case a kernel panic occurs.
- DASD related tools: Add Large Volume Support for ECKD DASDs (dasdfmt, fdasd, dasdview, zipl, dump tool)
- ziomon: Add report utilities
- vmur: Add "--convert" option: With this option a VMDUMP file can be converted into the LKCD dump
- Isluns: Add "--active" option: With this option all activated LUNs are shown.
- dasdview: Add "--characteristic" option: With this option a list of hardware specific DASD characteristics is shown. Currently only the encryption status is shown, this may be extended for future releases.
- tunedasd: Change the scaling of DASD profile data from binary to decimal shifting. Also print the scaling factor.
- qetharp, qethconf, osasnmpd and lsqeth: removed 2.4 supporting code
- dasdfmt: Add "--norecordzero" option: With this option the permission for the subsystem to format record zero is removed.
- dasdfmt: Add "--percentage" option: With this option one line for each formatted cylinder is printed showing the number of the cylinder and percentage of formatting process.

NEW

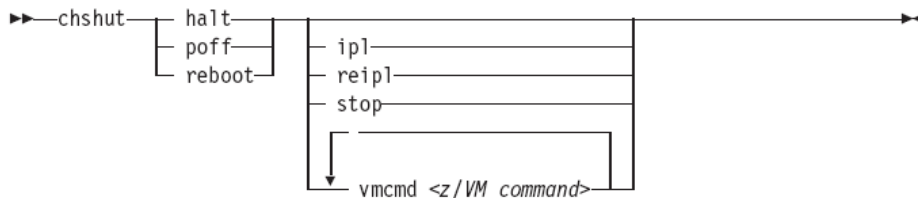
## Shutdown action tools

- \* **chreipl**: Configure a disk or change a an entry in the boot menu for the next boot cycle.



```
# chreipl node /dev/dasda
# chreipl node /dev/sda
# chreipl ccw -d 0.0.7e78 -L 1
# chreipl fcp --wwpn 0x500507630300c562 \
--lun 0x401040B300000000 -d 0.0.1700
```

- \* **chshut**: Change the entries below /sys/firmware to configure the shutdown behavior



```
# chshut halt ipl
# chshut halt vmcmd LOGOFF
# chshut poff vmcmd "MSG MASTER Going down" vmcmd "LOGOFF"
```

- \* **lsreipl**: command to see from which device your system will boot after you issue the reboot command. Further you can query the system for information about the current boot device.

```
# lsreipl
Re-IPL type:    ccw
Device:         0.0.4bb8
Loadparm:      5
```

- \* **lsshut**: command to see from which device your system will boot after you issue the reboot command. Further you can query the system for information about the current boot device.

```
# lsshut
```

Trigger	Action
Halt	stop
Panic	stop
Power off	stop
Reboot	reipl

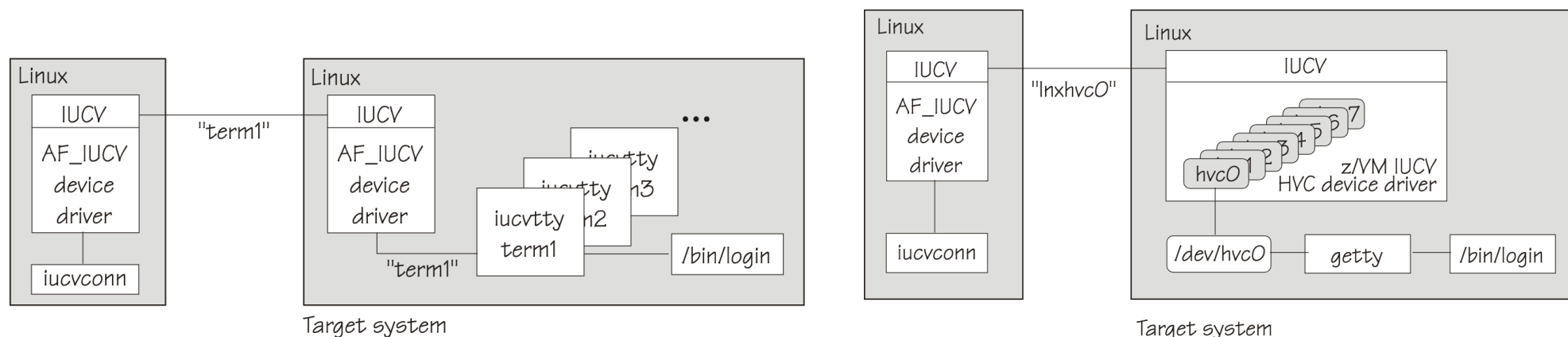




NEW

## IUCV terminal applications

- \* Full-screen terminal access to Linux guest operating systems on the same z/VM
- \* Access Linux instances with no external network because IUCV is independent from TCP/IP
- \* The IUCV terminal applications consist of:
  - **iucvconn** – Start terminal connection over IUCV
  - **iucvtty** – Allow remote logins over IUCV
  - **ts-shell** – Login shell for terminal servers over IUCV
- \* Terminal access over IUCV is provided by:
  - **iucvtty**, or
  - z/VM IUCV hypervisor console device driver (Linux kernel)



NEW

## IUCV terminal applications – examples

### \* Using the **iucvconn** program:

- To access the first z/VM IUCV HVC terminal on the Linux instance in z/VM guest LNXSYS02  

```
$ iucvconn LNXSYS02 lnxhvc0
```
- To create a transcript of the terminal session to the Linux instance in z/VM guest LNXSYS99  

```
$ iucvconn -s ~/transcripts/lnxsys99 LNXSYS99 lnxhvc0
```

### \* Using the **iucvtty** program:

- To allow remote logins using the terminal identifier „lnxterm“  

```
# iucvtty lnxterm
```
- To access the „lnxterm“ terminal on the Linux instance in z/VM guest LNXSYS01  

```
$ iucvconn LNXSYS01 lnxterm
```
- To use /sbin/sulogin instead of /bin/login for terminal “suserm”  

```
# iucvtty suserm -- /sbin/sulogin
```

### \* Configuring the Linux system for providing terminals over IUCV (using /etc/inittab)

- z/VM IUCV HVC terminal devices  

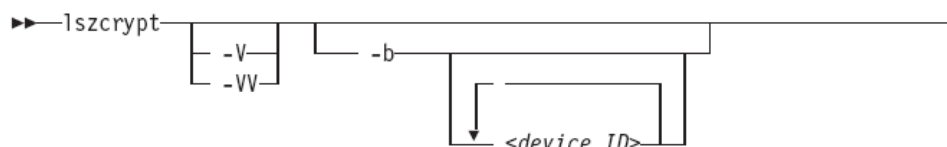
```
h0:2345:respawn:/sbin/agetty -L 9600 hvc0 linux
```
- iucvtty  

```
t1:2345:respawn:/usr/bin/iucvtty lnxterm
```

NEW

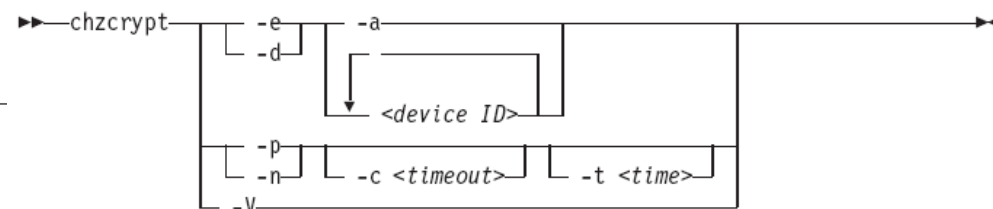
## chzcrypt / lszcrypt

- \* Use the lszcrypt command to display information about cryptographic adapters managed by zcrypt and zcrypt's AP bus attributes



- To display card type and online status of all available cryptographic adapters:  
# `lszcrypt -V`
- To display card type, online status, hardware card type, hardware queue depth, and request count for cryptographic adapters 0, 1, 10, and 12  
# `lszcrypt -VV 0 1 10 12`
- To display AP bus information:  
# `lszcrypt -b`

- \* Use the chzcrypt command to configure cryptographic adapters managed by zcrypt and modify zcrypt's AP bus attributes.



- To set the cryptographic adapters 0, 1, 4, 5, and 12 online:  
# `chzcrypt -e 0 1 4 5 12`
- To set all available cryptographic adapters offline:  
# `chzcrypt -d -a`
- To set the configuration timer for re-scanning the AP bus to 60 seconds and disable zcrypt's poll thread:  
# `chzcrypt -c 60 -n`

## cpuplugd

### \* Use the cpuplugd command to:

- Enable or disable CPUs based on a set of rules. **This increases the performance of single threaded applications within a z/VM or LPAR environment with multiple CPUs.** The rules can incorporate certain system load variables.
- Manage memory under z/VM.

### \* Configuration file: /etc/sysconfig/cpuplugd

### \* Init-Script: /etc/init.d/cpuplugd {start, stop, restart}

### \* Exemplary configuration file:

```
UPDATE="60"
```

```
CPU_MIN="2"
CPU_MAX="10"
```

```
HOTPLUG = "(loadavg > onumcpus +0.75) & (idle < 10.0)"
HOTUNPLUG = "(loadavg < onumcpus -0.25) | (idle > 50)"
```

```
CMM_MIN="0"
CMM_MAX="8192"
CMM_INC="256"
```

```
MEMPLUG = "swaprte > freemem+10 & freemem+10 < apcr"
MEMUNPLUG = "swaprte > freemem + 10000"
```

CPU management

Manage memory under z/VM

The cmm Kernel module has to be loaded

## Change & Display

- \* Use `chch` (Change channel path status) to set channel paths online or offline.
  - The actions are equivalent to performing a Configure Channel Path Off or Configure Channel Path On operation on the hardware management console.
  - To set channel path 0.19 into standby state issue:
 

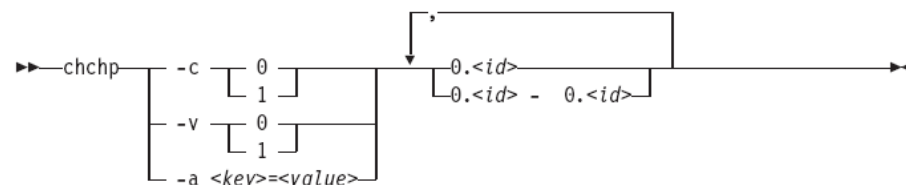
```
# chchp -a configure=0 0.19
```
  - To set the channel path with the channel path ID 0.40 to the standby state
 

```
# chchp --configure 0 0.40
Configure standby 0.40... done.
```
  - To set a channel-path to the configured state,
 

```
# chchp --configure 1 0.40
Configure online 0.40... done.
```
  - To set channel-paths 0.65 to 0.6f to the configured state issue:
 

```
# chchp -c 1 0.65-0.6f
```
  - To set channel-paths 0.12, 0.7f and 0.17 to 0.20 to the logical offline state issue:
 

```
# chchp -v 0 0.12,0.7f,0.17-0.20
```



- \* Use the `lsluns` command to discover and scan LUNs in Fibre Channel Storage Area Networks (SANs).
  - **This example shows all LUNs for port 0x500507630300c562:**

```
# lsluns --port 0x500507630300c562
Scanning for LUNs on adapter 0.0.5922
at port 0x500507630300c562:
0x4010400000000000
0x4010400100000000
[...]
```

- **This example shows all LUNs for adapter 0.0.5922:**

```
# lsluns -c 0.0.5922
at port 0x500507630300c562:
0x4010400000000000
[...]
```

```
at port 0x500507630303c562:
0x4010400000000000
[...]
```

## Crypto & DASD

- \* The `icostat` command is used to indicate whether libica uses hardware or works with software fallbacks.
  - It shows also which specific functions of libica are used.

```
# icastats
function | # hardware | # software
-----+-----+-----
SHA1     | 33210     | 49815
SHA224   | 171992    | 328312
SHA256   | 189565    | 440615
SHA384   | 172081    | 323235
SHA512   | 205170    | 266679
RANDOM    | 6716896   | 0
MOD EXPO | 29        | 53
RSA CRT  | 15        | 18
DES ENC  | 2366808   | 0
DES DEC  | 2366808   | 0
3DES ENC | 0         | 0
3DES DEC | 0         | 0
AES ENC  | 576713    | 414708
AES DEC  | 576688    | 414700
```

- \* Large Volume Support is a feature that allows to use ECKD devices with more than 65520 cylinders.
  - This features is available with DS8000 R4.0
  - Included in Linux Kernel 2.6.29
  - All DASD related tools have been updated for Large Volume Support
    - `dasdfmt`
    - `fdasd`
    - `dasdview`
    - `zipl`
    - `dump tools`

## Dump: Multi Volume Dump

### \* *How to prepare a set of ECKD DASD devices for a multi-volume dump? (64-bit systems only)*

- You can specify up to 32 partitions on ECKD DASD volumes for a multi-volume dump. We use two disk in this example:

```
# dasdfmt -f /dev/dasdc -b 4096
# dasdfmt -f /dev/dasdd -b 4096
```

- Create the partitions with fdasd. The sum of the partition sizes must be sufficiently large (the memory size + 10 MB):

```
# fdasd /dev/dasdc
# fdasd /dev/dasdd
```

- Create a file called sample\_dump\_conf containing the device nodes of the two partitions, separated by one or more line feed characters
  - Prepare the volumes using the zipl command.
- ```
# zipl -M sample_dump_conf
[...]
```

- **Display information on the involved volumes:**

```
# zgetdump -d /dev/dasdc
'/dev/dasdc' is part of Version 1 multi-volume dump, which is spread along the following DASD volumes:
0.0.4711 (online, valid)
0.0.4712 (online, valid)
[...]
```

### \* *To obtain a dump with the multi-volume DASD dump tool, perform the following steps:*

- Stop all CPUs, Store status on the IPL CPU.
- IPL the dump tool using one of the prepared volumes, either 4711 or 4712.
- After the dump tool is IPLed, you'll see a messages that indicates the progress of the dump. Then you can IPL Linux again

### \* *Copying a multi-volume dump to a file*

- Use zgetdump without any option to copy the dump parts to a file:
- ```
# zgetdump /dev/dasdc > multi_volume_dump_file
```

## Debug & Dump

- \* `dbginfo.sh` is a script to collect various system related files, for debugging purposes. It generates a tar-archive which can be attached to PMRs / Bugzilla entries

```
# dbginfo.sh
Create target directory /tmp/DBGINFO-
2009-04-15-22-06-20-t6345057
Change to target directory
/tmp/DBGINFO-2009-04-15-22-06-20-
t6345057
Get procfs entries
Saving runtime information into
runtime.out
Get file list of /sys
Get entries of /sys
[...]
```

- \* The `dumpconf` tool configures a dump device that is used for automatic dump in case of a kernel panic.
  - The command can be installed as service script under `/etc/init.d/dumpconf` or can be called manually.

- It reads the configuration file `/etc/sysconfig/dumpconf`.
- Example configuration for CCW dump device (DASD) and `reipl` after dump:
 

```
ON_PANIC=dump_reipl
DUMP_TYPE=ccw
DEVICE=0.0.4714
```
- Example configuration for FCP dump device (SCSI disk):
 

```
ON_PANIC=dump
DUMP_TYPE=fcpl
DEVICE=0.0.4711 WWPAN=0x5005076303004712
LUN=0x4713000000000000
BOOTPROG=0
BR_LBA=0
```
- Example configuration for re-IPL if a kernel panic occurs:
 

```
ON_PANIC=reipl
```
- Example of executing a CP command, and rebooting from device 4711 if a kernel panic occurs:
 

```
ON_PANIC=vmcmd
VMCMD_1="MSG MASTER Starting VMDUMP"
VMCMD_2="VMDUMP"
VMCMD_3="IPL 4711"
```



## z/VM: vmur

\* The vmur command provides all functions required to work with z/VM spool file queues:

- Receive: Read data from the z/VM reader file queue
- Punch or print: Write data to the z/VM punch or printer file queue and transfer it to another user's virtual reader, optionally on a remote z/VM node.
- List: Display detailed information about one or all files on the specified spool file queue.
- Purge: Remove one or all files on the specified spool file queue.
- Order: Position a file at the top of the specified spool file queue.

\* Produce and read Linux guest machine dump

- Produce guest machine dump: # `vmcp vmdump`
- Find spool ID of VMDUMP spool file in the output of the vmur li command:

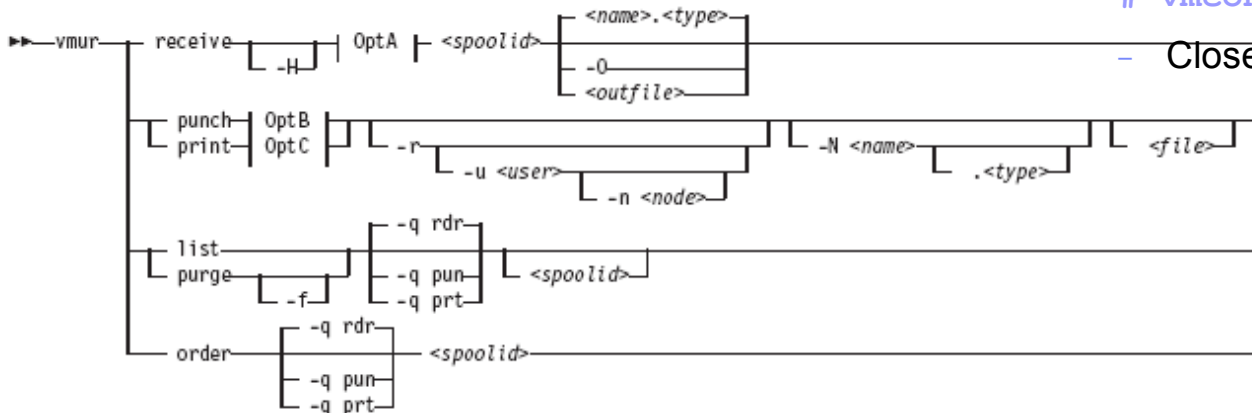
```
# vmur li
ORIGINID FILE CLASS RECORDS CPY HOLD DATE
TIME NAME TYPE DIST
T6360025 0463 V DMP 00020222 001 NONE 06/11
15:07:42 VMDUMP FILE T6360025
```

- Move vmdump file to top of reader queue with the vmur order command: # `vmur or 463`
- Read and convert the vmdump file to a file on the Linux file system in the current working directory:

```
# vmconvert /dev/vmrdr-0.0.000c linux_dump
```

- Close virtual reader: # `vmcp cl c`

VMDUMP file spool ID



## z/VM: vmur (cont)

### \* Log and read Linux guest machine console

- Begin console spooling:  
# vmcp sp cons start
- Produce output to VM console (for example, with CP TRACE).
- Close the console file and transfer it to the reader queue, find the spool ID behind the FILE keyword in the corresponding CP message.  
# vmcp sp cons clo \\* rdr  
RDR FILE 0398 SENT FROM T6360025 CON  
WAS 0398 RECS 1872 CPY 001 T NOHOLD  
NOKEEP
- Read the guest machine console file into a file on the Linux file system in the current working directory:  
# vmur re -t 398 linux\_cons

### \* Prepare z/VM reader to IPL Linux image

- Send parmfile to VM punch and transfer it to the reader queue:  
# vmur pun -r /boot/parmfile
- Find the parmfile spool ID in message:  
Reader file with spoolid 0465 created.
- Send image to VM punch and transfer it to reader queue:  
# vmur pun -r /boot/vmlinuz -N image
- Find the image spool ID in message:  
Reader file with spoolid 0466 created.
- Move image to first and parmfile to the second position in the reader queue:  
# vmur or 465  
# vmur or 466
- 6. Prepare re-IPL from the VM reader:  
# echo 0.0.000c >  
/sys/firmware/reipl/ccw/device
- Boot the Linux image in the VM reader:  
# reboot

## z/VM: vmcp

- \* Using the z/VM CP interface device driver (vmcp), you can send control program (CP) commands to the VM hypervisor and display VM's response.

```
# modprobe vmcp
# vmcp "q dasd"|grep T6345057
DASD 4DE0 ATTACHED TO T6345057 4DE0 R/W 0X4DE0
DASD 4DE1 ATTACHED TO T6345057 4DE1 R/W 0X4DE1
DASD 4DE2 ATTACHED TO T6345057 4DE2 R/W 0X4DE
DASD 4DE3 ATTACHED TO T6345057 4DE3 R/W 0X4DE3
```

# More Information

IBM developerWorks : Linux : Linux on System z : Development stream : s390-tools - Iceweasel <2>

File Edit View History Bookmarks Tools Help

http://www.ibm.com/developerworks/linux/linux390/s390-tools.html

Country/region [ select ]

IBM

Home Solutions Services Products Support & download

developerWorks

Linux on System z

s390-tools

What's new

Development stream

- Kernel
- s390-tools
- Other packages

Distribution hints

Tuning hints & tips

Archive

Feedback

Introduction

This page is the 'homepage' of the Linux on S390. s390-tools is a package with a set of user space tools.

Overview

The following table lists the available tools in the s390-tools package.

- Name of the tool and any major features in parentheses.
- Short functional description of the tool.

Tool/feature
chccdev (v1.3.0)
chchp (v1.6.2)
cpuplugd (v1.6.3)
dasdfmt (v1.0.0)
<ul style="list-style-type: none"> <li>DASD tool harmonization (v1.5.0)</li> </ul>
dasdinfo (v1.6.0)
dasdview (v1.0.0)
<ul style="list-style-type: none"> <li>DASD tool harmonization (v1.5.0)</li> </ul>
dbginfo (v1.1.0)
<ul style="list-style-type: none"> <li>adapptions for linux kernel 2.6 (v1.3.0)</li> </ul>
dumpconf (v1.6.0)
<ul style="list-style-type: none"> <li>VMCMD support (1.7.0)</li> </ul>
fdasd (v1.0.0)
<ul style="list-style-type: none"> <li>DASD tool harmonization (v1.5.0)</li> </ul>

Done

Linux on System z

## Using the Dump Tools

### November, 2008

*Linux Kernel 2.6 - Development stream*

Linux on System z

## Device Drivers, Features, and Commands

### November, 2008

*Linux Kernel 2.6 - Development stream*

SC39-8411-01

# Questions?



***Hans-Joachim Picht***

*Linux Technology Center*

*Linux on System z Kernel  
Development & Red Hat  
Liaison*

*IBM Deutschland Research  
& Development GmbH  
Schönaicher Strasse 220  
71032 Böblingen, Germany*

*Phone +49 (0)7031-16-1810  
Mobile +49 (0)175 - 1629201  
[hans@linux.vnet.ibm.com](mailto:hans@linux.vnet.ibm.com)*

