Troubleshooting

Version 1 Release 1
Troubleshooting

Version 1 Release 1
## Contents

### Chapter 1. Troubleshooting and support for Linux on System z
- Techniques for troubleshooting Linux on System z problems ........................................ 1
- Troubleshooting checklist for Linux on System z .............................................................. 3
- Collecting data for general Linux on System z problems .................................................. 4
- Collecting data for performance problems ........................................................................ 4
- Collecting data for network problems .............................................................................. 5
- Collecting data for hung system problems ....................................................................... 5
- Collecting data for middleware problems ....................................................................... 5

### Chapter 2. Tools for troubleshooting
- General tools ....................................................................................................................... 7
  - dbginfo - Collect information for debugging ................................................................. 7
  - supportconfig - SUSE Linux Enterprise Server troubleshooting ...................................... 7
  - sosreport - Generate debugging information for Red Hat Enterprise Linux systems .......... 8
- Performance tools ............................................................................................................ 9
  - sadc - System activity data collector ............................................................................. 9
  - iostat - Monitor input/output device load ..................................................................... 11
  - z/VM MONWRITE - Collect CP *MONITOR data ........................................................... 12
- Collecting data using DASD statistics .............................................................................. 13
- Collecting data using SCSI statistics .............................................................................. 13
- zimon - Collect FCP performance data ............................................................................ 14
- Special tools ...................................................................................................................... 15
  - s390dbf traces - Use the kernel debug feature .............................................................. 16
  - top - See resource usage .............................................................................................. 16
  - ps - Report a snapshot of the current processes ............................................................. 17
  - netstat - Show information about the Linux networking subsystem .............................. 17
  - tcpdump - Collect traffic information for a network interface ...................................... 18
  - oprofile - Profiling of all running code on Linux systems ........................................... 18
- Dump tools ......................................................................................................................... 18

### Chapter 3. Contacting IBM Support ............................................................................. 21

### Chapter 4. Exchanging information with IBM
- Sending information to IBM Support .............................................................................. 23
- Receiving information from IBM Support ........................................................................ 23

### Accessibility .................................................................................................................. 25

### Notices ............................................................................................................................ 27
- Trademarks ....................................................................................................................... 28

### Index .................................................................................................................................. 29
Chapter 1. Troubleshooting and support for Linux on System z

To isolate and resolve problems with Linux on System z, you can use the troubleshooting and support information. This information contains instructions for using the problem-determination resources that are provided with Linux on System z.

Techniques for troubleshooting Linux on System z® problems

Troubleshooting is a systematic approach to solving a problem. The goal of troubleshooting is to determine why something does not work as expected and how to resolve the problem. Certain common techniques can help with the task of troubleshooting.

The first step in the troubleshooting process is to describe the problem completely. Problem descriptions help you and the IBM technical-support representative know where to start to find the cause of the problem. This step includes asking yourself basic questions:

- What are the symptoms of the problem?
- Where does the problem occur?
- When does the problem occur?
- Under which conditions does the problem occur?
- Can the problem be reproduced?

The answers to these questions typically lead to a good description of the problem, which can then lead you to a problem resolution.

What are the symptoms of the problem?

When starting to describe a problem, the most obvious question is “What is the problem?” This question might seem straightforward; however, you can break it down into several more-focused questions that create a more descriptive picture of the problem. These questions can include:

- Who, or what, is reporting the problem?
- What are the error codes and messages?
- How does the system fail? For example, is it a loop, hang, crash, performance degradation, or incorrect result?

Where does the problem occur?

Determining where the problem originates is not always easy, but it is one of the most important steps in resolving a problem. Many layers of technology can exist between the reporting and failing components. Networks, disks, and drivers are only a few of the components to consider when you are investigating problems.

The following questions help you to focus on where the problem occurs to isolate the problem layer:

- Is the problem specific to one platform or operating system, or is it common across multiple platforms or operating systems?
- Is the current environment and configuration supported?
- Do all users have the problem?
(For multi-site installations.) Do all sites have the problem?

If one layer reports the problem, the problem does not necessarily originate in that layer. Part of identifying where a problem originates is understanding the environment in which it exists. Take some time to completely describe the problem environment, including the operating system and version, all corresponding software and versions, and hardware information. Confirm that you are running within an environment that is a supported configuration; many problems can be traced back to incompatible levels of software that are not intended to run together or have not been fully tested together.

When does the problem occur?

Develop a detailed timeline of events leading up to a failure, especially for those cases that are one-time occurrences. You can most easily develop a timeline by working backward: Start at the time an error was reported (as precisely as possible, even down to the millisecond), and work backward through the available logs and information. Typically, you need to look only as far as the first suspicious event that you find in a diagnostic log.

To develop a detailed timeline of events, answer these questions:

- Does the problem happen only at a certain time of day or night?
- How often does the problem happen?
- What sequence of events leads up to the time that the problem is reported?
- Does the problem happen after an environment change, such as upgrading or installing software or hardware?

Responding to these types of questions can give you a frame of reference in which to investigate the problem.

Under which conditions does the problem occur?

Knowing which systems and applications are running at the time that a problem occurs is an important part of troubleshooting. These questions about your environment can help you to identify the root cause of the problem:

- Does the problem always occur when the same task is being performed?
- Does a certain sequence of events need to happen for the problem to occur?
- Do any other applications fail at the same time?

Answering these types of questions can help you explain the environment in which the problem occurs and correlate any dependencies. Remember that just because multiple problems might have occurred around the same time, the problems are not necessarily related.

Can the problem be reproduced?

From a troubleshooting standpoint, the ideal problem is one that can be reproduced. Typically, when a problem can be reproduced you have a larger set of tools or procedures at your disposal to help you investigate. Consequently, problems that you can reproduce are often easier to debug and solve.

However, problems that you can reproduce can have a disadvantage: If the problem is of significant business impact, you do not want it to recur. If possible, re-create
the problem in a test or development environment, which typically offers you more flexibility and control during your investigation.

- Can the problem be re-created on a test system?
- Are multiple users or applications encountering the same type of problem?
- Can the problem be re-created by running a single command, a set of commands, or a particular application?

### Troubleshooting checklist for Linux on System z

When you open a problem, provide as much information as possible about the circumstances.

Answering the following questions can help you or IBM® support to determine the cause for problems that occur with Linux on System z:

1. **How does the problem manifest itself? What are the symptoms?**
   - When this problem occurs, is a specific error message or error code issued?
   - Is trace output of the operation available?

2. **How long has the problem been occurring?**
   - Is it a first time occurrence? When did it happen? (Date and time help to analyze the logs.)
   - How frequently does it occur?
   - Is there any pattern?

3. **If the problem occurred subsequent to some period of normal operation, did anything change in the environment?**
   - Was an operating system patch applied?
   - Did the network environment change? For example, was a server moved or a domain migrated?
   - Did the system recently fail or abnormally terminate?

4. **If you know (for example, based on message prefixes or error codes), where does the problem occur? On one or more systems, production or test environment?**

5. **Can you reproduce the problem on a test system (so that you do not negatively affect the production environment)? What steps are required to reproduce the problem?**

6. **How many users are impacted?**
   - Is this problem affecting one, some, or all users?
   - Is the problem occurring only for a user who was recently added to the environment, such as a new employee?
   - Do differences exist between the users who are affected and the users who are not affected?

7. **How many applications or business processes are impacted?**
   - Is this problem affecting one, some, or all applications or business processes?
   - Is the problem occurring only for a new application or business process?
   - Do differences exist between the applications or business processes that are affected and the applications or business processes that are not affected by the problem?

In your report, describe the server and storage infrastructure in as much detail as possible:
• Machine setup (for example, z9®, or z10™?)
• Storage server (for example, DS8000®)
• Storage attachment (for example FICON®, ESCON®, or FCP?)
• Disk configuration
• Network (OSA (type, mode), HiperSockets™)
• Network topologies
• Middleware setup (databases, web servers, SAP, or TSM. Include version information, if relevant)

You can now collect additional diagnostic data that is required for an IBM technical-support representative to effectively troubleshoot the problem.

## Collecting data for general Linux on System z problems

Collect diagnostic data when a problem occurs. Then submit the diagnostic data to IBM Support. Whatever the problem, start with this general collection of data.

### About this task

Collecting data before opening a problem management record (PMR) can help you to answer the following questions:

- Do the symptoms match any known problems? If so, has a fix or workaround been published?
- Can the problem be identified and resolved without a code fix?
- When does the problem occur?

The diagnostic data that you collect, and the sources from which you collect that data, are dependent on the type of problem that you are investigating. A base set of information is typically always required. For specific symptoms, you might need to collect additional problem-specific data.

When you submit a problem to IBM Support, you must provide a base set of information.

### Procedure

To collect general diagnostic data:

1. Collect the base set of diagnostic information by using the `dbginfo` command.
2. Depending on your distribution, also collect distribution-specific information:
   - On SUSE Linux Enterprise Server, run `supportconfig`.
   - On Red Hat Enterprise Linux, run `sosreport`.

## Collecting data for performance problems

If performance is a problem, collect diagnostic data that you can use to diagnose and resolve the problem.

### Procedure

To collect diagnostic data for performance diagnostics:

1. Start `sadc` (System Activity Data Collection) and provide `sar` files.
2. If running as guest under `z/VM®`, collect `z/VM MONWRITE` data.
3. Attach the data files to the opened problem report.

**Collecting data for network problems**

If the network has a problem, collect diagnostic data that you can use to diagnose and resolve the problem.

**Procedure**

To collect diagnostic data for network diagnostics:
1. Provide a diagram of your network setup.
2. Use `netstat` to collect diagnostic data.
3. Attach the data files to the opened problem report.

**Collecting data for hung system problems**

If the system hangs, collect diagnostic data that you can use to diagnose and resolve the problem.

**Procedure**

To collect diagnostic data for hung system diagnostics:
1. Create a kernel dump.
2. Include `system.map`, `kerntypes` (if available) and the `vmlinux` file.

**Collecting data for middleware problems**

If middleware is the problem, collect diagnostic data that you can use to diagnose and resolve the problem.

**Procedure**

To collect data for problems with any middleware product (for example, databases):
1. Contact the product support organization.
2. Collect the appropriate debug data as instructed.
3. Attach the data files to the opened problem report.
Chapter 2. Tools for troubleshooting

A variety of troubleshooting tools are available to help you diagnose and resolve product problems for Linux on System z.

General tools

Tools that can be used in most cases when debugging Linux on System z problems.

dbginfo - Collect information for debugging

The dbginfo.sh script collects various system-related files for debugging purposes. It captures the current system environment and generates a tar file.

If the Linux system runs as z/VM guest operating system, dbginfo also collects information about the z/VM guest setup.

The dbginfo.sh script is part of the s390-tools package in Novell SUSE distributions and the s390-utils package in Red Hat distributions.

The service and development team continuously improve dbginfo.sh. You can download the latest version from the developerWorks® website at http://www.ibm.com/developerworks/linux/linux390/s390-tools.html The dbginfo.sh script is included in the s390-tools tar ball.

Authorization

- Running the script requires root authority.
- For z/VM guest operating systems you require privilege class B.

Syntax

```
/dbginfo.sh
```

Example

To generate a diagnostic report with dbginfo, issue the command:

```
[root@system]# 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
[...]```

supportconfig - SUSE Linux Enterprise Server troubleshooting

The supportconfig script gathers system troubleshooting information on SUSE Linux Enterprise Server systems. It captures the current system environment and generates a tar-archive.

The script file collects complementary info to the dbginfo.sh script. The supportconfig script is part of the Supportutils package.
Authorization

Running the script requires root authority.

Syntax

See the supportconfig man page for more details.

Example

To run supportconfig, issue:

```
root@system:~ # supportconfig
```

Output

The script produces a tar ball. The location of the tar ball is given in the script output:

```
==================================================================
Support Utilities - Supportconfig
Script Version: 2.25-96
Script Date: 2009 02 24
==================================================================
Gathering system information
Basic Server Health Check... Done
[...]
Creating Tar Ball
[ DONE ]==================================================================
Log file tar ball: /var/log/nts_h42lp42_100719_1431.tbz
Log file size: 572K
Log file md5sum: 1dfc98f3a3192771ad970ecc31b6e9d9
```

sosreport - Generate debugging information for Red Hat Enterprise Linux systems

The sosreport script gathers system troubleshooting information. It captures the current system environment and generates a tar file.

The script file collects complementary info to the dbginfo.sh script. The sosreport script is part of the support-utils package.

Authorization

Running the script requires root authority.

Syntax

See the sosreport man page for details.
Example

To run sosreport, issue the command:

```
[root@system]# sosreport
```

Output

The script produces a tar ball. The location of the tar ball is given in the script output:

```
[root@system]# sosreport
sosreport (version 1.7)
[...]
This process may take a while to complete.
No changes will be made to your system.
Press ENTER to continue, or CTRL-C to quit.
Please enter your first initial and last name [h42lp27]: ABC
Please enter the case number that you are generating this report for: DEF
Creating compressed archive...
Your sosreport has been generated and saved in:
/tmp/sosreport-ABC-427338-6e8879.tar.bz2
[...]
```

Performance tools

Tools that can be used when debugging Linux on System z performance problems.

sadc - System activity data collector

The `sadc` command samples system data a specified number of times at a specified interval measured in seconds. It writes to the specified output file or the standard output in binary format. The `sadc` command is a backend to the `sar` command.

For example, data about the following areas is captured:

- CPU utilization
- Disk I/O overview and on device level
- Network I/O and errors on device level
- Memory usage and swapping

The tools report statistics data over time and create average values for each item.

Starting sadc/sar as a service

Start `sadc`/`sar` by using the `sysstat` service. When started as a service, the data files are written to the `/var/log/sa` directory. The files are named `sa<dd>` and `sar<dd>` respectively, where `<dd>` is the current day's two-digit date. Both files are constantly updated during the day.

Procedure

To start the `sadc` command as a service:

Start the `sysstat` service.
To start the sysstat service with Red Hat distributions as a permanent service that persists across reboots, issue:

```
  service sysstat start
```

To check the status of the service, issue:

```
  chkconfig –list |grep sysstat
```

To start the sysstat service using SLES 10 either configure the service using YaST or use the following command:

```
  chkconfig -s sysstat on|12345
```

To start the sysstat service only for the current session:

```
  service sysstat start
```

On SLES 10 this is not persistent across reboots.

To start the sysstat service with SLES 11 you have to configure the service using YaST in order to have data collection persistent across reboots.

To start the sysstat service directly, issue:

```
  /etc/init.d/boot.sysstat start
```

This is not persistent across reboots.

To check the status of the sysstat service, issue:

```
  /etc/init.d/boot.sysstat status
```

### Results

To report performance data, include both the sadc and the sar data files with the problem report.

### What to do next

After you collect the appropriate diagnostic data, you can complete the following tasks, as appropriate:

- Chapter 3, “Contacting IBM Support,” on page 21
- Chapter 4, “Exchanging information with IBM,” on page 23

### Starting sadc/sar directly

If your problem requires data collection that is not covered by the sar/sadc defaults, you can start the tools manually. This may be the case when you need smaller sampling interval than the default.

### About this task

Depending on the time period during which performance problems are seen, either use a default sampling interval of 10 minutes, or, if performance problems occur for
a couple of minutes once in a while, shorten the sampling interval. An interval of less than a minute is recommended.

Procedure

1. To start the **sadc** command directly, issue a command of the following form:

/usr/lib64/sa/sadc [options] [interval [count]] > <sadc_outfile>

See the **sadc** man page for details.

For example:

[root@system:]# /usr/lib64/sa/sadc 1 5 > sadc_outfile
[root@system:]# /usr/lib64/sa/sadc -d 1 5 > sadc_outfile

Omit the **count** parameter to let sadc sample data until it is stopped.
Use the -d option to collect disk statistics. By default sadc does not report disks activity to prevent data files from growing too large.

2. Extract data and write records using the **sar** command. Use a command of the following form:

**sar** -A -f <sadc outfile> > <sar outfile>

For example:

[root@system:]# sar -A -f sadc_outfile > sar_outfile

where:

- **-A** reports all the collected statistics.
- **-f** specifies the binary input file.

The **sar** command creates a collection of performance reports from the collected **sadc** data and writes these reports to an output file.

Results

To report performance data, include both the sadc and the sar data files with the problem report.

What to do next

After you collect the appropriate diagnostic data, you can complete the following tasks, as appropriate:

- **Chapter 3, “Contacting IBM Support,” on page 21**
- **Chapter 4, “Exchanging information with IBM,” on page 23**

**iostat - Monitor input/output device load**

The **iostat** command monitors system input/output device load by observing the time the devices are active in relation to their average transfer rates.

The **iostat** report shows:

- Throughput
• Device queue information
• Service time

Authorization

Root access is required on Linux operating systems.

Syntax

See the iostat man page for the complete syntax and all options.

```
iostat [options] interval count
```

Parameters

- `d` Collects disk statistics.
- `t` Prints a time stamp for each report
- `k` Displays statistics in kilobytes per second instead of blocks per second.
- `x` Displays extended statistics, if available.

Examples

To generate a report with a sampling interval of 10 seconds, collecting disk statistics in KB per second, including a timestamp, and extended statistics, issue the command:

```
[root@system]# iostat -dtkx 10
```

Usage notes

For disk I/O problems on SUSE Linux Enterprise Server 9 and Red Hat Enterprise Linux 4, iostat is preferred over sadc/sar, because the sadc/sar version on these distributions do not include proper disk I/O statistics.

z/VM MONWRITE - Collect CP *MONITOR data

If your Linux system runs as a guest operating system under z/VM and encounters performance problems, use the MONWRITE utility and include CP *MONITOR data in the problem report.

The z/VM monitor records are in binary format. Make sure that:
• The records are packed and tersed correctly.
• The record size settings are correct.
• The binary to ASCII conversion is made correctly.

For more details about how to collect and upload z/VM MONWRITE data, see www.ibm.com/vm/perf/tips/collect.html

Usage notes

• The sadc and sar files must cover the same time interval as the z/VM MONWRITE data.
• Use the default sampling time interval of 1 minute.

Collecting data using DASD statistics

The DASD statistics kernel function monitors the activities of the DASD device driver and the storage subsystem. It mainly records processing time of I/O operations within a given time interval.

Procedure

To collect diagnostic data using DASD statistics:

1. Start DASD statistics with the following command:
   
   ```bash
   # echo set on > /proc/dasd/statistics
   ```

2. Summarized histogram information is available in `/proc/dasd/statistics`, and can be extracted with the following command:
   
   ```bash
   # cat /proc/dasd/statistics
   ```

3. Stop DASD statistics with the following command:
   
   ```bash
   # echo set off > /proc/dasd/statistics
   ```

Results

DASD statistics creates a summary for all devices. An IOCTL interface is available to collect the statistics for individual devices. To get DASD statistics for an individual DASD, use the `tunedasd` command:

   ```bash
   # tunedasd -P /dev/dasd
   ```

What to do next

After you collect the appropriate diagnostic data, you can complete the following tasks:

- [Chapter 3, “Contacting IBM Support,” on page 21](#)
- [Chapter 4, “Exchanging information with IBM,” on page 23](#)

Collecting data using SCSI statistics

The SCSI statistics collect statistics of I/O operations on FCP devices on a request base, separately for read and write requests. It also gives detailed information about the latency.

About this task

Statistical data on FCP devices can be collected using:

- SUSE Linux Enterprise Server 9 SP3 + maintenance (kernel version 2.6.5-7.283 and higher)
- SUSE Linux Enterprise Server 10 GA (kernel version 2.6.16.21-0.8 and higher)
**Procedure**

By default, data gathering is turned off.

1. To switch on data gathering for the devices, enter:
   ```
   echo on=1 > definition
   ```

2. To switch off data gathering for the devices, enter:
   ```
   echo on=0 > definition
   ```

3. To reset the collected data to 0, enter:
   ```
   echo data=reset > definition
   ```

**Results**

Depending on your distribution, the files for zfcp statistics can be found as follows:

- For SUSE Linux Enterprise Server 10 and later, depending on where debugfs is mounted: `<mount_point_debugfs>/statistics`. For example, if debugfs is mounted at directory `/sys/kernel/debug/`, all the collected statistics data can be found at `/sys/kernel/debug/statistics/`.
- For SUSE Linux Enterprise Server 9, depending on where `/proc` is mounted (SUSE Linux Enterprise Server 9 does not use debugfs): `<mount_point_proc>/statistics`.
- For each device (adapter as well as LUN) a subdirectory is created when mounting the device. The subdirectory is named:
  - `zfcp-<device-bus-id>` for an adapter
  - `zfcp-<device-bus-id>-<WWPN>-<LUN>` for a LUN

  Each subdirectory contains two files, a data file and a definition file.

**What to do next**

After you collect the appropriate diagnostic data, you can complete the following tasks:

- Chapter 3, “Contacting IBM Support,” on page 21
- Chapter 4, “Exchanging information with IBM,” on page 23

**ziomon - Collect FCP performance data**

For SUSE Linux Enterprise Server 11, use the `ziomon` tool is used to gather FCP performance data.

The monitor tool `ziomon` collects information and details about:

- The FCP configuration
- The I/O workload
- The utilization of the FCP resources

**Authorization**

Root access is required on Linux operating systems.
Syntax

See the ziomon man page for the complete syntax and all options.

```
ziomon -l <size limit of output file> -i <interval>
```

```
- d <duration> - o <output file> <device node>
```

Parameters

- `i <interval>`
  Specifies the elapsed time between writing data to disk in seconds. Defaults to 60 seconds.

- `d <duration>`
  Specifies the monitoring duration in minutes. Must be a multiple of the interval length.

- `l <size limit of output file>`
  Defines the upper limit of the output files. Must include one of the suffixes M (megabytes), G (gigabytes) or T (terabytes). Note that this is only a tentative value that can be slightly exceeded.

- `o <output file>`
  Specifies the prefix for the log file, configuration file and aggregation file.

- `<device>`
  Denotes one or more device names separated by blanks.

Examples

To generate a diagnostic report for devices /dev/sda and /dev/sdb, issue the command:

```
[root@system]# ziomon -i 20 -d 5 -l 50M -o trace_data /dev/sda /dev/sdb
```

Output

The ziomon tool creates two output files in the directory where it was started:

- `<output file>.cfg` holds various configuration data from the system
- `<output file>.log` holds the raw data samples taken during the data collection phase in a binary format

Usage notes

- Needs vmalloc space for each device node and CPU.
- The ziomon tool can be stopped with CTRL+C before the time period expires.

Special tools

Tools for special circumstances that can be used when debugging Linux on System z problems.
s390dbf traces - Use the kernel debug feature

All device drivers and other kernel components write debug log records. These records are available after a system crash. You can also read and save them on a running system.

To look at these debug logs use the Linux file system debugfs, which must be mounted. For example, to mount debugfs to /sys/kernel/debug, issue the following command:

```
[root@system]# mount -t debugfs /sys/debug /sys/kernel/debug
```

Below the s390dbf directory each registered component is represented by a subdirectory with the name of that component. The subdirectories contain files that represent different views of the debug log. Available views are: hex_ascii, sprintf, flush, pages, and level.

The debug information written to the logs depends on the debug level set for that log. The debug level ranges from 0 for the least detail to 6 for the most detail. The default level is 2. Only debug entries with a level that is lower or equal to the actual level are written to the log.

To set or change a debug level, from the s390dbf subdirectory for the component you want to work with, issue:

```
echo <value> > level
```

Examples

- To collect the maximum amount of debug information, issue:
  
  ```
echo 6 > level
  ```

- To flush the debug log buffer for the component, issue:
  
  ```
echo - > flush
  ```

- The kernel debug feature uses wraparound memory buffers. To increase the buffer size, read it first and then enter a higher value with the following command:
  
  ```
echo 10 > pages
  ```

**top - See resource usage**

The top command provides a dynamic real-time view of a running system and shows resource usage on a thread level. It can display, for example, CPU utilization and detailed memory usage.
Syntax

See the `top` man page for the complete syntax and all options.

```
top [ -d delay ] [ -n iterations ] [ -p pid ]
```

Parameters

- `-b` Writes the output for each interval into a file.
- `-d` Specifies the delay time interval in seconds.
- `-n` Indicates the maximum number of iterations `top` should produce before ending.
- `-p` Limits the output to the specified processes.

In the running `top` program, use the F key to configure displayed columns. Use the W key to write the current configuration to `~/.toprc` This is the default directory.

Example

To write 180 iterations 1 second apart into a file, issue:

```
[root@system]# top -b -d 1 -n 180 >top.log 2>&1
```

ps - Report a snapshot of the current processes

The `ps` command gives very comprehensive statistics data on process level and reports a snapshot of the current processes.

See the `top` man page for the complete syntax and all options.

Example

The following sample command shows every process in an easily readable format:

```
[root@system]# ( DELAY=10; while [ true ]; do echo "\"date\"; ps -elO pid,user,%cpu,%mem,wchan,stat,time,flags,etime,command:50; sleep $DELAY; done ) | tee psinfo.out
```

netstat - Show information about the Linux networking subsystem

The `netstat` command shows information about the Linux networking subsystem.

In particular `netstat` shows:

- Summary information of each protocol
- Number of incoming and outgoing packages
- Various error states, for example TCP segments retransmitted

See the `netstat` man page for the complete syntax and all options.
**Example**

The following sample command displays summary statistics:

```
[root@system]# netstat -s
```

Where:

- `-s` displays summary statistics for each protocol.

**tcpdump - Collect traffic information for a network interface**

The tcpdump network analysis tool dumps traffic collected for a given network interface.

**Syntax**

See the `tcpdump` man page for the complete syntax and all options.

```
tcpdump -s <length> -X -i <interface>
```

**Parameters**

- `-s` Writes `<length>` of data from each packet rather than the default 65535 bytes.
- `-X` Writes each packet in hexadecimal and in ASCII format.
- `-i` Identifies the network interface.

**Example**

To dump network traffic for interface eth0, issue the command:

```
[root@system]# tcpdump -s 65000 -X -i eth0
```

**oprofile - profiling of all running code on Linux systems**

The oprofile tool offers profiling of all running code on Linux systems, providing a variety of statistics.

For more details, see


**Dump tools**

When the system hangs, create a kernel dump.

The following dump tools are available:

- The DASD dump tool writes the dump directly to a DASD partition. It supports both ECKD™ and FBA DASDs.
- The tape dump tool writes the dump directly to an ESCON/FICON tape device.
- The SCSI dump tool writes the dump into file system. It is supported for LPAR and as of z/VM 5.4.
VMDUMP (for z/VM guest operating systems) writes the dump to z/VM spool space (VM reader). VMDUMP uses a dump format specific to z/VM, the dump must be converted. Do not use VMDUMP to dump large VM guests; the dump process is very slow.

For more details, see

Chapter 3. Contacting IBM Support

IBM Support provides assistance with product defects, answers FAQs, and helps users resolve problems with the product.

Before you begin

After trying to find your answer or solution by using other self-help options such as technotes, you can contact IBM Support. Before contacting IBM Support, your company or organization must have an active IBM software maintenance agreement (SWMA), and you must be authorized to submit problems to IBM. For information about the types of available support, see the “Support portfolio” topic in the “Software Support Handbook”.

Procedure

To contact IBM Support about a problem:

1. Define the problem, gather background information, and determine the severity of the problem. For more information, see the “Getting IBM support” topic in the Software Support Handbook.
2. Gather diagnostic information.
3. Submit the problem to IBM Support in one of the following ways:
   - Using IBM Support Assistant (ISA):
   - Online through the [IBM Support Portal]. You can open, update, and view all of your service requests from the Service Request portlet on the Service Request page.
   - By phone: For the phone number to call in your region, see the Directory of worldwide contacts web page.

Results

If the problem that you submit is for a software defect, IBM Support creates a software patch. Missing or inaccurate documentation is normally corrected in the next documentation update. The patch is sent to the Linux distributor for inclusion. Whenever possible, IBM Support provides a workaround that you can implement until the patch is available. For a subscription service for Linux operating system software updates, see the Linux support site available at [http://www.ibm.com/systems/z/os/linux/support/](http://www.ibm.com/systems/z/os/linux/support/) offers.
Chapter 4. Exchanging information with IBM

To diagnose or identify a problem, you might need to provide IBM Support with data and information from your system. In other cases, IBM Support might provide you with tools or utilities to use for problem determination.

Sending information to IBM Support

To reduce the time that is required to resolve your problem, you can send trace and diagnostic information to IBM Support.

Procedure

To submit diagnostic information to IBM Support:
1. Open a problem management record (PMR).
2. Collect the diagnostic data that you need. Diagnostic data helps reduce the time that it takes to resolve your PMR. See the following topics:
   - "Collecting data for general Linux on System z problems" on page 4
   - "Collecting data for performance problems" on page 4
   - "Collecting data for network problems" on page 5
   - "Collecting data for hung system problems" on page 5
   - "Collecting data for middleware problems" on page 5.
3. Compress the files by using the .zip or .tar file format.
4. Transfer the files to IBM. You can use one of the following methods to transfer the files to IBM:
   - Standard data upload methods: FTP, HTTP
     There are two servers available for uploading data:
     - testcase.boulder.ibm.com (US only)
     - ecurep.ibm.com (international)
     For upload instructions, see http://www.ibm.com/de/support/ecurep/index.html
   - Secure data upload methods: FTPS, SFTP, HTTPS
   - IBM Support Assistant
   - The Service Request tool

All of these data exchange methods are explained on the IBM Support website.

Receiving information from IBM Support

Occasionally an IBM technical-support representative might ask you to download diagnostic tools or other files. You can use FTP to download these files.

Before you begin

Ensure that your IBM technical-support representative provided you with the preferred server to use for downloading the files and the exact directory and file names to access.

Procedure

To download files from IBM Support:
1. Use FTP to connect to the site that your IBM technical-support representative provided and log in as anonymous. Use your email address as the password.

2. Change to the appropriate directory:
   a. Change to the /fromibm directory.
      `cd fromibm`
   b. Change to the directory that your IBM technical-support representative provided.
      `cd nameofdirectory`

3. Enable binary mode for your session.
   `binary`

4. Use the `get` command to download the file that your IBM technical-support representative specified.
   `get filename.extension`

5. End your FTP session.
   `quit`
Accessibility

Accessibility features help users who have a disability, such as restricted mobility or limited vision, to use information technology products successfully.

Documentation accessibility

The Linux on System z publications are in Adobe Portable Document Format (PDF) and should be compliant with accessibility standards. If you experience difficulties when you use the PDF file and want to request a Web-based format for this publication, use the Reader Comment Form in the back of this publication, send an email to eservdoc@de.ibm.com, or write to:

IBM Deutschland Research & Development GmbH
Information Development
Department 3248
Schoenaicher Strasse 220
71032 Boeblingen
Germany

In the request, be sure to include the publication number and title.

When you send information to IBM, you grant IBM a nonexclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

IBM and accessibility

See the IBM Human Ability and Accessibility Center for more information about the commitment that IBM has to accessibility at

www.ibm.com/able
Notices

This information was developed for products and services offered in the U.S.A. IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

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

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:
INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION “AS IS” WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

The licensed program described in this information and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement, or any equivalent agreement between us.

All statements regarding IBM’s future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.
This information is for planning purposes only. The information herein is subject to change before the products described become available.

**Trademarks**

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at "Copyright and trademark information" at


Adobe is either a registered trademark or trademark of Adobe Systems Incorporated in the United States, and/or other countries.

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

### A
- accessibility 25

### C
- checklists
  - troubleshooting
    - Linux on System z 3
- collecting data for general Linux on System z problems 4

### D
- DASD statistics 13
- dbginfo 7
- diagnostic data
  - collecting for hung system problems 5
  - collecting for middleware problems 5
  - collecting for network problems 5
  - collecting for performance problems 9
  - starting sadc/sar directly 10
- dump tools 18

### G
- general Linux on System z problems
  - collecting data for 4

### H
- hung system
  - troubleshooting
    - collecting diagnostic data 5

### I
- iostats 11

### L
- Linux on System z
  - troubleshooting checklist 3

### M
- middleware
  - troubleshooting
    - collecting diagnostic data 5
- MONWRITE 12

### N
- network
  - troubleshooting
    - collecting diagnostic data 5

### O
- oprofile 18

### P
- performance problems
  - collecting diagnostic data 9
- problem determination
  - exchanging information with IBM Support 23
- ps 17

### S
- s390dbf 16
- sadc
  - starting as service 9
- sar
  - starting as service 9
- SCSI statistics 13
- starting sadc/sar directly
  - troubleshooting
    - collecting diagnostic data 10
- supportconfig 7, 8

### T
- tcpdump 18
- top 17
- troubleshooting checklist
  - Linux on System z 3
  - collecting data for general Linux on System z problems 4
  - collecting data for hung system problems 5
  - collecting data for middleware problems 5
  - collecting data for network problems 5
  - collecting data for performance problems 9
  - starting sadc/sar directly 10
- general tools 7
- identifying problems, techniques for 1
- Linux on System z 1
- performance tools 9
- special tools 16
- tools 7

### Troubleshooting and Support
- contacting IBM Support 21
- exchanging information with IBM Support 23
- troubleshooting techniques 1

### Troubleshooting Tools
- DASD statistics 13
- dbginfo 7
- dump tools 18
- iostats 11
- MONWRITE 12
- oprofile 18
- ps 17
- s390dbf 16

© Copyright IBM Corp. 2012
troubleshooting tools  (continued)

SCSI statistics  13
supportconfig 7, 8
tcpdump 18
top 17
ziomon 14
Readers’ Comments — We'd Like to Hear from You

Linux on System z
Troubleshooting
Version 1 Release 1

We appreciate your comments about this publication. Please comment on specific errors or omissions, accuracy, organization, subject matter, or completeness of this book. The comments you send should pertain to only the information in this manual or product and the way in which the information is presented.

For technical questions and information about products and prices, please contact your IBM branch office, your IBM business partner, or your authorized remarketer.

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you. IBM or any other organizations will only use the personal information that you supply to contact you about the issues that you state on this form.

Comments:

Thank you for your support.
Submit your comments using one of these channels:
• Send your comments to the address on the reverse side of this form.
• Send your comments via email to: eservdoc@de.ibm.com

If you would like a response from IBM, please fill in the following information:

Name
Address
Company or Organization
Phone No.
Email address
Readers’ Comments — We’d Like to Hear from You

IBM Deutschland Research & Development GmbH
Information Development
Department 3248
Schoenaicher Strasse 220
71032 Boeblingen
Germany
IBM