

CUDB Node Preventive Maintenance

APPLICATION INFORMATION

Copyright

© Ericsson AB 2016-2018. All rights reserved. No part of this document may be reproduced in any form without the written permission of the copyright owner.

Disclaimer

The contents of this document are subject to revision without notice due to continued progress in methodology, design and manufacturing. Ericsson shall have no liability for any error or damage of any kind resulting from the use of this document.

Trademark List

All trademarks mentioned herein are the property of their respective owners. These are shown in the document Trademark Information.



Contents

1	Introduction	1
1.1	Scope	1
1.2	Revision Information	1
1.3	Target Groups	2
1.4	Typographic Conventions	2
2	Overview	3
3	Continuous Maintenance Procedures	5
3.1	Supervising Alarms	5
3.2	Performance Management	5
3.3	Monitoring Backbone Connectivity	6
4	Daily Maintenance Procedures	7
4.1	Create System Data Backup	7
4.2	Check Log Files	7
4.3	Perform Lightweight Consistency Check	8
4.4	Check Infrastructure Related Components (for CUDB Systems Deployed on Native BSP 8100)	8
4.5	Check Operating System	8
5	Weekly Maintenance Procedures and Tasks with Longer Periodicity	11
5.1	Creating Software and Configuration Backups	11
5.2	Performing Consistency Check	11
	Glossary	13
	Reference List	15





1 Introduction

This document describes the maintenance procedures to prevent breakdown and failures in Ericsson Centralized User Database (CUDB). This document provides the schedule of planned maintenance tasks to be performed to preserve and enhance CUDB system reliability. For more information on troubleshooting, refer to *CUDB Troubleshooting Guide*, Reference [3].

1.1 Scope

This document covers the following maintenance procedures:

- Continuous maintenance procedures.
- Daily maintenance procedures.
- Weekly maintenance procedures and tasks with longer periodicity.

1.2 Revision Information

Rev. A

This document is based on 1/1541-CSH 109 067/9 with the following changes:

- Updated hardware information.
- Virtualization deployment terminology updates throughout the document.
- Section 1.3 on page 2: Updated target groups and list of references.

Rev. B

Other than editorial changes, this document has been revised as follows:

- Virtualization terminology updates throughout the document.

Rev. C

Other than editorial changes, this document has been revised as follows:

- **Section 5.2 User Administration:** Removed subsection.

Rev. D

Other than editorial changes, this document has been revised as follows:



- Updated Network Management System (NMS) terminology.

Rev. E

Other than editorial changes, this document has been revised as follows:

- **Section 4.3 Check Log Faults:** Removed subsection. Content available in the *Troubleshooting cudbAnalyser Results* section of *CUDB Logchecker*, Reference [2].

Rev. F

Editorial changes only.

Rev. G

Other than editorial changes, this document has been revised as follows:

- Section 5.1 on page 11: Added information about data safety and disk space saving recommendation.

1.3 Target Groups

This document is intended for Network Operations Center (NOC) personnel.

Procedures related to CUDB application are intended for users in the `cudbadmin` group. For more details on permissions and how to execute the related commands, refer to the *LDE User Management* section of *CUDB Security and Privacy Management*, Reference [1] and the *CUDB Commands* section of *CUDB Node Commands and Parameters*, Reference [2].

1.4 Typographic Conventions

Typographic conventions can be found in the following document:

- *Typographic Conventions*



2 Overview

This document describes the following periodic checks and maintenance procedures to prevent breakdown and failures in CUDB:

- Continuous tasks and checks. It comprises all the monitoring activities that should be in place during all the time while the CUDB system is running in a production environment. It includes overseeing CUDB fault management and performance management data along with regular supervision of the network interconnecting all the CUDB nodes in the system.
- Daily tasks and checks. It is a comprehensive set of checks and management operations recommended to be executed on a daily basis in a live CUDB system.
- Weekly tasks and sporadic checks. It includes other minor management procedures and checks that can be scheduled weekly or less often.

The periodicity classification of preventive maintenance tasks is recommended to prevent failures in the CUDB system.

If instructions for collecting and enclosing troubleshooting data due to a Customer Service Request (CSR) are needed when experiencing problems with the product, refer to the *Data Collection* section of *Data Collection Guideline for CUDB*, Reference [13] for more information.





3 Continuous Maintenance Procedures

The following sections describe the maintenance procedures to be performed continually.

3.1 Supervising Alarms

CUDB nodes, as network elements, provide a northbound interface, used to communicate with Network Management System (NMS)). CUDB nodes send Simple Network Management Protocol (SNMP) traps to an external NMS for fault management through the Ericsson SNMP Agent (ESA) component. Performing a daily check on unacknowledged active alarms is recommended, even if the alarm status is continuously supervised. Alarms are issued in CUDB as SNMP traps sent from each CUDB node. Based on information displayed in alarms, the system administrator can locate the source of the event and by using the relevant documentation, steps can be taken to resolve or prevent failures.

CUDB node internal components have automatic procedures to detect faults and malfunctions.

To get a complete list of the raised alarms, execute the `fmactivealarms` command.

For more information on managing alarms, refer to the *Alarm Management* section of *CUDB Node Fault Management Configuration Guide*, Reference [4].

The severity field, which is always set in all of the alarms, indicates the importance of the failure and its impact on normal CUDB system operation. Handling alarms with higher severity values before alarms with lower priority values is recommended.

Non auto ceased alarms must be acknowledged once the cause that produced the alarm is completely solved. For more information on how to clear non auto ceased alarm, refer to the *Clearing Alarms* section of *CUDB Node Fault Management Configuration Guide*, Reference [4].

An auto ceased alarm is terminated when the alarm with the cease of the faulty action is received.

3.2 Performance Management

CUDB includes a number of performance measurements to monitor CUDB node resources and system activity. Performance data is continuously collected and reported in eXtensible Markup Language (XML) files.



CUDB infrastructure and software components provide a set of statistic counters with valuable performance information that helps the system administrators to be aware of possible malfunctions in the system. Regularly check the values of counters for any deviation to baseline to detect and avoid inactivity or overload in any component of the system.

For further details on the counters and on how to read them, refer to the *Counter Generation and Publishing* section of *CUDB Performance Guide*, Reference [5].

Besides CUDB specific counters, CUDB allows applications whose data is hosted in CUDB to define their own counters. For more information on creating application counters, refer to *CUDB Application Counters*, Reference [6], and for a list and description on different counters, refer to *CUDB Counters List*, Reference [7].

3.3 Monitoring Backbone Connectivity

CUDB system relies on the Internet Protocol (IP) transport network of the customer to connect CUDB nodes placed in different network site locations. CUDB nodes implement some system monitoring functions and protocols to check availability and the ability to reach remote CUDB nodes in a system. Refer to the *System Level Availability* section of *CUDB High Availability*, Reference [8] for more information on these mechanisms.

Note: The backbone connectivity is essential for a healthy CUDB system, and therefore it is highly recommended to implement strong and exhaustive supervision mechanisms to check the connectivity between network sites hosting CUDB nodes to alleviate the impact of network failures on the system. For more information, refer to *CUDB Node Network Description*, Reference [9].



4 Daily Maintenance Procedures

The following sections describe maintenance procedures to be performed daily.

4.1 Create System Data Backup

The CUDB backup program utility allows the operator to perform system data backups to generate a complete backup of all data stored in a CUDB system.

Although CUDB is a highly resilient and redundant database system, it is strongly recommended to perform a system data backup everyday.

Backups can be scheduled or executed manually. For more information on how to perform CUDB system data backup, refer to the *Performing System Data Backup* section of *CUDB Backup and Restore Procedures*, Reference [10].

Copying the generated backup files to an external equipment for extra data safety and to save disk storage system space is recommended.

4.2 Check Log Files

Check the high severity events from the log files daily. Filter the logs with the following severities:

- EMERG by executing the `grep -i EMERG <syslog file>` command.
- ALERT by executing the `grep -i ALERT <syslog file>` command.
- CRIT by executing the `grep -i CRIT <syslog file>` command.
- ERR by executing the `grep -i ERR <syslog file>` command.

The EMERG, CRIT, and ERR log files are stored on the System Controllers (SCs) in `/var/log/SC_2_1: messages, kernel, auth..` and `/var/log/SC_2_2: messages, kernel, auth..`, and on the payload blades or Virtual Machines (VMs) in `/var/log/PL*: messages_mysql, messages, kernel, auth..` while the ALERT log file is found in NDB logs and stored on the payload blade or VM in `/local/cudb/mysql/ndbd/data`.

For more information on extended CUDB logging, refer to *CUDB Node Logging Events*, Reference [11].

These logs include both the manual checks and the logs generated by the CUDB Logchecker. The CUDB Logchecker is an additional software monitoring component on top of the current monitoring processes that aims to work as a



preventive maintenance tool. For more information about CUDB Logchecker, refer to *CUDB Logchecker*, Reference [12].

CUDB Logchecker runs every 12 hours automatically. Following are the CUDB Logchecker scripts:

- `cudbGetLogs` script (to collect preventive maintenance logs for log analysis) runs at 00:25 and 12:25, unless it is configured otherwise.
- `cudbAnalyser` script (to analyze logs gathered and preprocessed by `cudbGetLogs`) runs at 00:50 and 12:50, unless it is configured otherwise.

The execution of CUDB Logchecker may produce alarms. Refer to the *Automatic Log Collection and Log Analysis* section of *CUDB Logchecker*, Reference [12] for more information.

4.3 Perform Lightweight Consistency Check

Performing a Lightweight Consistency Check as described in the *Detecting Inconsistencies between Replicas* section of *CUDB Data Storage Handling*, Reference [14] is recommended.

4.4 Check Infrastructure Related Components (for CUDB Systems Deployed on Native BSP 8100)

Before starting the infrastructure check, perform Lightweight Directory Access Protocol (LDAP) log error checks.

Refer to the “BSP Fault Management” and “BSP System Notifications” documents in the BSP 8100 CPI for detailed information on how to supervise and take preventive actions on native BSP 8100.

4.5 Check Operating System

CUDB nodes use Linux Distribution Extension (LDE) as operating system. For more information on LDE, refer to *LDE Management Guide*, Reference [17].

Perform the following checks in each blade or VM:

- Check the available free memory by executing the `> free` command.
- Check the disk storage system utilization by executing the `> df -k` command.
- Check CPU utilization by executing the `> mpstat -P ALL 1 5` command.



In case of abnormal repeated values on any of these measurements along with performance management data deviations, please contact Ericsson Support.





5 Weekly Maintenance Procedures and Tasks with Longer Periodicity

The following sections describe weekly maintenance procedures and tasks.

5.1 Creating Software and Configuration Backups

Having backups on software packages and configuration files periodically is recommended.

For more information on the backup and restore procedure, refer to the *Software and Configuration Backup and Restore* section of *CUDB Backup and Restore Procedures*, Reference [10].

Copying the generated backup files to an external equipment for extra data safety and to save disk storage system space is also recommended.

5.2 Performing Consistency Check

Performing a Consistency Check as described in the *Consistency Check* section of *CUDB Consistency Check*, Reference [15] is recommended.





Glossary

For the terms, definitions, acronyms, and abbreviations used in this document, refer to *CUDB Glossary of Terms and Acronyms*, Reference [16].





Reference List

CUDB Documents

- [1] *CUDB Security and Privacy Management*
- [2] *CUDB Node Commands and Parameters*
- [3] *CUDB Troubleshooting Guide*
- [4] *CUDB Node Fault Management Configuration Guide*
- [5] *CUDB Performance Guide*
- [6] *CUDB Application Counters*
- [7] *CUDB Counters List*
- [8] *CUDB High Availability*
- [9] *CUDB Node Network Description*
- [10] *CUDB Backup and Restore Procedures*
- [11] *CUDB Node Logging Events*
- [12] *CUDB Logchecker*
- [13] *Data Collection Guideline for CUDB*
- [14] *CUDB Data Storage Handling*
- [15] *CUDB Consistency Check*
- [16] *CUDB Glossary of Terms and Acronyms*

Other Ericsson Documents

- [17] *LDE Management Guide*