

# CUDB Logchecker

---

## USER GUIDE

**Copyright**

© Ericsson AB 2016. 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

|          |   |           |
|----------|---|-----------|
| <b>1</b> | <b>Introduction</b>                       | <b>1</b>  |
| 1.1      | Document Purpose and Scope                | 1         |
| 1.2      | Revision Information                      | 1         |
| <b>2</b> | <b>Overview</b>                           | <b>3</b>  |
| <b>3</b> | <b>Usage Information</b>                  | <b>5</b>  |
| 3.1      | General Information                       | 5         |
| 3.2      | Automatic Log Collection and Log Analysis | 5         |
| 3.3      | Manual Log Collection and Analysis        | 7         |
|          | <b>Glossary</b>                           | <b>9</b>  |
|          | <b>Reference List</b>                     | <b>11</b> |





# 1 Introduction

This section contains the purpose and scope of the document, and also the revision information.

## 1.1 Document Purpose and Scope

The purpose of this document is to provide the instructions and tools needed to automate CUDB log collection and log analysis. This document requires knowledge of the product. It is addressed to both Ericsson personnel and System Administrators. If the contents of this document and the *Preventive Maintenance, Logchecker Found Error(s)*, Reference [1] document are not enough to fix a fault, then contact next Level of Maintenance Support.

## 1.2 Revision Information

### **Rev. A**

This document is based on 12/1553-CSH 109 067/9 with the following changes:

- Terminology updates throughout the document because of virtualized deployment support.





## 2 Overview

CUDB Logchecker is a software monitoring component on top of the current monitoring processes that aims to work as a preventive maintenance tool.





## 3 Usage Information

This section provides usage information about CUDB Logchecker.

### 3.1 General Information

CUDB Logchecker consists of the following two scripts:

- `cudbGetLogs`

The purpose of this script is to collect preventive maintenance logs for log analysis. Log collection takes approximately 4-5 minutes, and the resulting log is saved under the `/home/cudb/monitoring/preventiveMaintenance/` directory. The name of the log file contains the Node ID and a timestamp in the following format:

```
<CUDB_node_id>_<YYYYmmddHHMM>.log
```

For example:

```
CUDB_99_201201020025.log
```

**Note:** Logs are deleted automatically from the `/home/cudb/monitoring/preventiveMaintenance/` directory, if they are older than 180 days.

- `cudbAnalyser`

`cudbAnalyser` is a tool to analyze logs gathered and preprocessed by `cudbGetLogs`. It looks for certain patterns that can indicate malfunctions already present in the CUDB system, and also searches for traces of suspicious events that can lead to system failures if proper preventive maintenance actions are not carried out in time. `cudbAnalyser` categorizes its findings, determines the risk levels, and presents the results on the standard output. The automatic log analysis also sends an alarm to the Operation and Support System (OSS), if it finds problems. All these faults are weighted by CUDB Logchecker. The severity of the alarm depends on the accumulated weights of the faults (this means that multiple minor faults can also result in a major alarm). Severity levels are warning, minor, major or critical.

**Note:** These scripts can be executed only on the active System Controller (SC).

### 3.2 Automatic Log Collection and Log Analysis

CUDB log collection starts automatically every day at 00:25 and 12:25.



At 00:50 and 12:50, there is a scheduled CUDB log analysis, which saves the detailed result under the following location:

```
/home/cudb/monitoring/preventiveMaintenance/cron_analys
is.<SC_NAME>.log
```

In the above path, *SC\_NAME* can be *SC\_2\_1* or *SC\_2\_2*.

The automatic log analysis can also send alarms: see Section 3.1 on page 5 for more information.

**Note:** Automatic log collection and analysis is performed only in the active SC.

The alarm severity is calculated by the number and the severity of faults found during the last analysis (that is, multiple minor faults can result in a major alarm). All these faults are weighted by CUDB Logchecker, then the severity of alarm is calculated and set, using the weight as input.

The severity levels of the alarm can be as follows:

- warning
- minor
- major
- critical

### 3.2.1 Defining Custom Monitoring Intervals

It is possible to define custom monitoring intervals for CUDB Logchecker. Specify monitoring for both SCs separately.

Configuration can be checked using the `crontab -l` command, as shown below in Example 1.

```
CUDB81 SC_2_1# crontab -l
# DO NOT EDIT THIS FILE - edit the master and reinstall.
# (/var/spool/cron/tabs/root installed on Wed Oct 17 13:59:03 2012)
# (Cron version V5.0 -- $Id: crontab.c,v 1.12 2004/01/23 18:56:42 vixie Exp $)
0,15,30,45 * * * * /home/cudb/oam/performanceMgmt/appCounters/scripts/appCounters.cron >> /dev/null
25 0,12 * * * /bin/bash /opt/ericsson/cudb/OAM/bin/cudbGetLogs
50 0,12 * * * /bin/bash /opt/ericsson/cudb/OAM/bin/cudbAnalyser --auto-check --send-alarm --save-counter > \
/home/cudb/monitoring/preventiveMaintenance/cron_analysis.SC_2_1.log
37 0 * * * /bin/bash /opt/ericsson/cudb/Monitors/bin/cudbCheckConsistency --locked --alarms >/dev/null 2>&1 || tr
7 0 * * * /bin/bash /opt/ericsson/cudb/Monitors/bin/cudbCheckReplication --locked --alarms >/dev/null 2>&1 || tr
*/1 * * * * /opt/ericsson/cudb/Monitors/keepAlive/bin/keepAlive_monitor.sh 2>&1
```

#### Example 1 Configuration Check

Define the same configuration files on both of the SCs. The paths of the files are as follows:

```
/home/cudb/monitoring/preventiveMaintenance/logchecker.SC
_2_1.conf
```

```
/home/cudb/monitoring/preventiveMaintenance/logchecker.SC_2_2.conf
```



The following options are accepted:

- `getlogs_schedule`: To define the schedule for log collection.
- `analyser_schedule`: To define the schedule for log analysis.

An example configuration file is shown below:

```
CUDB45 SC_2_1# cat logchecker.SC_2_1.conf
getlogs_schedule=25 0,4,8,12,16,20 * * *
analyser_schedule=50 0,4,8,12,16,20 * * *
```

**Note:** Follow the standard `cron` expression format when defining the value of options. Configuration files must take effect after the SC is rebooted. Use the `crontab -e` command to activate `cron` changes immediately, without rebooting the SC and defining configuration files.

If the configuration files do not exist, the following default options take effect:

- `getlogs_schedule=25 0,12 * * *`
- `analyser_schedule=50 0,12 * * *`

The absence of a configuration file means that the default values are used.

**Note:** Always leave at least 25 minutes between log collection and log analysis, in order to let the log collection finish before the log analysis starts.

## 3.3 Manual Log Collection and Analysis

It is possible to use the CUDB Logchecker to help troubleshooting. For more details, refer to *CUDB Troubleshooting Guide*, Reference [2].





## Glossary

For the terms, definitions, acronyms and abbreviations used in this document, refer to *CUDB Glossary of Terms and Acronyms*, Reference [3].





## Reference List

### **CUDB Documents**

- [1] *Preventive Maintenance, Logchecker Found Error(s)*
- [2] *CUDB Troubleshooting Guide*
- [3] *CUDB Glossary of Terms and Acronyms*