

# Data Collection Guideline for MTAS MTAS

---

## OPERATING INSTRUCTIONS

**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

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Prerequisites	1
<b>2</b>	<b>Workflow</b>	<b>3</b>
<b>3</b>	<b>Mandatory Data</b>	<b>5</b>
3.1	General Data to Be Collected	5
3.2	Data Collected Based on Specific Problems	5
<b>4</b>	<b>Data Collection Use</b>	<b>11</b>
4.1	MTAS Data Collection Feature	11
4.2	Start Data Collection	12
4.3	Housekeeping of the Results of Data Collection	14
4.4	Profiles and Steps in Data Collection	15
<b>5</b>	<b>Other Useful Information</b>	<b>19</b>





# 1 Introduction

The purpose of this document is to instruct what troubleshooting data is to be collected and enclosed in a Customer Service Request (CSR) or Trouble Report (TR) in case a problem is experienced with MTAS.

This document also describes the procedure to collect the needed information.

Consider the recommended printouts and tracings listed in the document as requirements for meaningful CSR or TR analysis. If necessary data, descriptions, or enclosures are missing, it can result in more data requests from the Ericsson Customer Support.

## 1.1 Prerequisites

This section describes the possible documents, tools, and required conditions before starting the data collection procedure.

It is expected that the reader has prior knowledge about telecommunication, including knowledge about the virtualized environment and the MTAS. It is assumed that the reader is familiar with concepts, terminology, and abbreviations within these areas.

### 1.1.1 Parameters

Necessary parameters when performing data collection of the MTAS are shown in Table 1.

Table 1 Installation and Configuration Parameters

Name	Value	Description
oam-vip	____.____.____. ____	The external VIP address of the O&M network.  For an explanation of the VIP address concept, refer to <i>Virtual IP Address Management</i> .
linux_user_name		The platform administrator name on the SC processors.
linux_user_password		The platform administrator password on the SC processors.

### 1.1.2 Tools

A workstation with an SSH client must be available before performing any procedure in this document.





## 2 Workflow

The workflow for collecting data from the MTAS node is as follows:

1. Collect mandatory data that is needed in connection to any problems experienced. Go to Section 3 on page 5.
2. Collect specific data based on the type of problem that is experienced. Go to Section 3.2 Data Collected Based on Specific Problems on page 5 and the collecting procedures specified in Section 4.1 MTAS Data Collection Feature on page 11.
3. Collect other useful information if it is available within an acceptable amount of time and effort. Go to Section 5 on page 19.







## 3 Mandatory Data

The data described in this section is always to be included in a CSR or TR.

It is important that data is collected as soon as possible after the problem has occurred as the relevant data can be lost if this activity is postponed.

### 3.1 General Data to Be Collected

The following types of data are to be collected:

- Version information of the MTAS product and other relevant nodes.
- A detailed description of the problem and for which scenarios the problem has been observed.
- If known, a detailed step by step description of how the fault can be reproduced
- Information about alarms and notifications that can be related to the observed problem.
- References to other CSR or TR problems, which have been observed when the fault occurred.
- Complete node configuration, exported from the Configuration Management browser.
- Information about recent configuration changes, software upgrades, and similar activities that have been performed.
- Values of MTAS counters *\*Error*, *\*Failed*, *\*NoKE*, and *\*NoKI*; Refer to section *Checking Counters* in *MTAS Health Check*.
- Any `crashcollector` files that can be related to the observed problem.

### 3.2 Data Collected Based on Specific Problems

This section describes different types of data based on specific problem types that can be collected. The data types to be collected and included in a CSR or TR depends on the problem experienced.

This section describes different types of data that can be collected. The problem experienced decides which of the following data types is to be included in a CSR or TR.



### 3.2.1 Software Versions

The version information for the MTAS software components is to be collected.

### 3.2.2 MTAS Log Files

The logging information that is to be collected is described in Table 2.

*Table 2 MTAS Log Files*

Filename	Log Path
vDicos Virtual Machine Log	<b>SC</b> <X>:/cluster/storage/no-backup/cdclsv/log/lpmsv/ Log: vm<N>-PL-<Y>
Processor Log	<b>SC</b> <X>:/var/log/SC-<X>/ <b>SC</b> <X>:/var/log/PL-<Y>/ Log: messages
MTAS Application Log	<b>SC</b> <X>:/storage/no-backup/coremw/var/log/saflog/MTASAppLogs/vdicos/ Log: MTAS_<from-date_to-date>.log
Crash collector Log	<b>SC</b> <X>:/cluster/storage/no-backup/cdclsv/cadump/ Log: vdicosvmca-PL-<Y>-<date>.<time>
MTAS Catalina Log	<b>PL</b> <Y>:/opt/mmas/appserver/traffic_instance<N>/logs/ <b>SC</b> <X>:/var/log/PL-<Y>/mmas/appserver/traffic_instance<N>/ Log: catalina.log
MTAS Catalina Out Log	<b>PL</b> <Y>:/opt/mmas/appserver/traffic_instance<N>/logs/ Log: catalina.out
Access Logs	<b>SC</b> <X>:/storage/no-backup/coremw/var/log/saflog/MMASAccessLogs/ Log: AccessLog<from-date_to-date>.log
CAI3G Log (AuditLog)	<b>SC</b> <X>:/storage/no-backup/coremw/var/log/saflog/MMASAuditLogs/ Log: Auditlog<from-date_to-date>.log

Where:

- <X> is the SC number, 1 or 2.
- <Y> is the PL number, starting with 3 and increases by 1.
- <N> is the instance number, starting with 0 and increases by 1 for every PL; 0 for PL-3, 1 for PL-4, and so on.

### 3.2.3 Routing Information

The following logging information is to be collected:

- Routing Information on the SC and PL processors by running the **route-n** command on each of the PLs and SCs.



- Configuration of Evolved Virtual IP (eVIP), verified from the `/storage/system/config/evip-apr9010467/evip.xml` file.

### 3.2.4 Alarms, Notifications, and Events

The MTAS triggers alarms for the most critical events that require operator intervention. The alarm information is accessible through the command `lde-alarm`. Alerts are also triggered to report relevant events for the operator.

Alarms and notifications information can be found in the `FaultManagementLog/alarm` and `FaultManagementLog/alert` directories that are stored on the SC under:  
`/cluster/storage/no-backup/coremw/var/log/saflog/`.

Operating Instructions (OPIs) for each event describe actions to be taken to cease alarms.

A list of the different alarms generated by the MTAS can be found in *MTAS Alarm List*.

### 3.2.5 Performance Measurement Counters

Performance Measurement (PM) counters are useful problem indicators. In particular, this includes the PMF counter reports from the period when the faulty situation occurred.

These counters are available with COM File Management in `PerformanceManagementReportFiles` file group.

**Note:** For more information on how to transfer counter-files, refer to *File Management*.

Alternatively, it is also possible to collect only counter-information for the service to be reported.

### 3.2.6 Clients Used

The following must be provided:

- A list of the clients and versions used.

### 3.2.7 Surrounding Nodes

The following must be provided:

- A list of the surrounding node versions, both Ericsson and other vendors.



### 3.2.8 Traffic Scenarios

The following must be provided:

- A detailed description of the faulty traffic scenario.
- Network trace (PCAP) including the faulty sequence (if the problem can be repeated) together with information on what packet in the trace being evidence of the incorrect MTAS behavior.
- Specify what the expected traffic behavior of MTAS is, together with a reference to the standard specification or MTAS Function Specification defining the expected behavior.
- Specify for what MTAS Application Server role the problem is experienced (for example MMTel Telephony AS, SCC-AS)
- MTAS service data (XML) provisioned on the subscriber affected by the faulty scenario.
- Application trace using an appropriate trace profile as recommended by *MTAS AppTrace* (if the problem can be repeated).

### 3.2.9 Network Configuration

The following must be provided:

- A description of the network configuration.

### 3.2.10 Configuration Parameter Values

The following must be provided:

- The configuration data exported from IMM using the following command:

```
cmw-immconfig-export <filename>
```

### 3.2.11 Co-located Applications

The following must be provided:

- A list of all the applications and their versions that are co-located on the same node as the MTAS.

### 3.2.12 AppTrace

Valuable information can be obtained by using the AppTrace function in the MTAS.



For more information about AppTrace, refer to *MTAS AppTrace*.

### **3.2.13 Trace Pcap**

Traces generated from Wireshark™ or other tracing tools are to be included in a CSR or TR depending on the specific problem type.

### **3.2.14 License Manager Log Files**

License Manager Functionality related events are stored in log files, which are located in the following path of the cluster:

```
/storage/clear/lm-apr9010503/log/lm.SC-<X>.log
```

Where <X> is the number designation of the processor.





## 4 Data Collection Use

### 4.1 MTAS Data Collection Feature

The data collection tool is intended to collect data that is needed to attach to a CSR, and to collect other useful information, if available in an acceptable amount of time and effort.

The data collection tool is composed by several steps. Steps are specific scripts covering different parts of the MTAS software. Steps are reading states, configurations, and produce logs; or, they collect existing logs into a given place.

Most of the produced logs are readable for the user; for example, `EVipAndRouting` collects the status of the eVIP controller and the status of ports into a table.

All steps are grouped in the following four predefined profiles in data collection: Limited, Basic, Full, and SLA.

#### **Limited profile**

Contains mainly MTAS specific logs, information from approximately the last 1 hour. Some other small steps are added as a plus.

#### **Basic profile**

Contains all the steps from Limited profile plus all other logs, except dump files.

#### **Full profile**

Contains everything from Basic profile plus dump files. Archive size can be extremely large.

#### **SLA**

The Service Level Agreement (SLA) profile is for system internally use, only. It contains only one step: Sla.

See Section 4.4 Profiles and Steps in Data Collection on page 15 for a detailed specification of the profiles.

Depending on the selected steps/profile, the execution of Data Collection can take several minutes.

The results of the running data collection are stored in:

```
/cluster/storage/no-backup/dc/
```



## 4.2 Start Data Collection

### 4.2.1 Data Collection Using ECLI

The Data Collection profile packers are listed with the `cdclsv-list` command and executed with the `cdclsv-invoke` command. Execution status can be checked with the `cdclsv-status` command.

#### Steps

1. Log on to ECLI:

```
ssh <username>@<oam-vip> -p 830 -t -s cli
```

2. Check which profiles are available:

```
cdclsv-list
```

The following is an example output:

```
cdclsPk=DcMtasBasic,cdcls=CDCLSVSite
cdclsPk=DcMtasFull,cdcls=CDCLSVSite
cdclsPk=DcMtasLimited,cdcls=CDCLSVSite
cdclsPk=DcMtasSla,cdcls=CDCLSVSite
cdclsPk=HcMtasPostUpgrade,cdcls=CDCLSVSite
cdclsPk=HcMtasPreUpgrade,cdcls=CDCLSVSite
```

3. Select one of the profiles listed in the previous step:

```
DcMtasBasic
DcMtasFull
DcMtasLimited
```

4. Start data collection with the selected profile:

```
cdclsv-invoke cdclsPk=<profile>,cdcls=CDCLSVSite
```

Example with the `DcMtasBasic` profile:

```
cdclsv-invoke cdclsPk=DcMtasBasic,cdcls=CDCLSVSite
```

5. Check the status of the packing:

```
cdclsv-status cdclsPk=<profile>,cdcls=CDCLSVSite
```

Example: `cdclsv-status cdclsPk=DcMtasBasic,cdcls=CDCLSVSite`

The results are found in: `/cluster/storage/no-backup/dc/`





## 4.2.2 Data Collection Using CDCLS

Data collection using the CDCLS (Crash Dump and Console Log Collection Service) is performed by executing packer objects for data collection. The Data Collection profile packers are listed with the `cdclsv-list-packers` command and executed with the `cdclsv-pack` command.

To start data collection using CDCLS:

1. Check which profiles are available:

```
cdclsv-list-packers | grep cdclsPk=DcMtas
```

The following results are shown:

```
cdclsPk=DcMtasBasic,cdcls=CDCLSVSite
cdclsPk=DcMtasFull,cdcls=CDCLSVSite
cdclsPk=DcMtasLimited,cdcls=CDCLSVSite
cdclsPk=DcMtasSla,cdcls=CDCLSVSite
```

2. Select one of the following profiles:

```
DcMtasBasic
DcMtasFull
DcMtasLimited
DcMtasSla
```

**Note:** If the profile to use is not known, select `DcMtasFull`.

3. Start data collection with the selected profile:

```
cdclsv-pack cdclsPk=<profile>,cdcls=CDCLSVSite
```

Example with the `DcMtasBasic` profile:

```
cdclsv-pack cdclsPk=DcMtasBasic,cdcls=CDCLSVSite
```

4. Check the status of the packing:

```
cdclsv-pack-status cdclsPk=<profile>,cdcls=CDCLSVSite
```

Example:

```
cdclsv-pack-status cdclsPk=DcMtasBasic,cdcls=CDCLSVSite
```

The results are found in:

```
/cluster/storage/no-backup/dc/
```



### 4.2.3 Data Collection Used by MTAS Health Check

Health check calls the required steps of data collection, and analyzes the produced logs. At the end of the analysis, health check chooses a verdict from the following list:

- PASS
- FAIL
- INFO
- ERROR
- VERIFY

For more information on these verdicts, refer to *MTAS Health Check*.

If the verdict is PASS, the logs from data collection are removed.

If the verdict is FAIL, INFO, ERROR, or VERIFY, the logs remain available for further analysis.

## 4.3 Housekeeping of the Results of Data Collection

Housekeeping is required as the results are collected cumulative (by default). The housekeeping of directory `/storage/no-backup/dc` is configured in ECLI. The configuration can be changed and checked in ECLI, for example:

```
>dn ManagedElement=1,SystemFunctions=1,FileM=1,FileGroupPolicy=dataCollection
```

```
>show -v
```

The following is an output example:

```
fileGroupPolicyId="dataCollection"  
fullFileGroupAction=DISCARD_OLDEST  
maxFileGroupSize=10  
maxNumberFiles=4  
retentionTime=10  
userLabel=[] <empty>
```



## 4.4 Profiles and Steps in Data Collection

*Table 3 Profiles and Steps in Data Collection*

Name	Function	Used in Limited Profile	Used in Basic Profile	Used in Full Profile	Related HC Profile / Step
Ait	Collects AIT (Automatic Installation Tool) logs from <code>/cluster/storage/no-backup/ait</code> <code>-apr9010496_1</code>	NO	YES	YES	None
Alarms	Collects the list of lde-alarms from every host in the cluster. ( <code>/usr/sbin/lde-alarm --status --full --all</code> )	YES	YES	YES	None
AlarmsAndNotifications	Collects notifications logs from <code>/storage/no-backup/coremw/var/log/saflog</code> if they exist. Lists the sums of different types of alarms, and lists the alarms.	YES	YES	YES	All/ AlarmsAndNotifications
AllMtasPortsStatus	Creates a table with the MTAS ports ( <code>netstat -an</code> ). The predefined port numbers are as follows: <ul style="list-style-type: none"> <li>• <code>mtasXdmsXCAPP</code>Port: 8090</li> <li>• <code>mtasXdmsCCMPP</code>Port: 8096</li> <li>• <code>mtasSoap</code>Port: 9080</li> <li>• <code>mtasXdmsCai3GSecure</code>Port: 8443</li> <li>• <code>mtasXdmsCai3G</code>Port: 8095</li> </ul>	YES	YES	YES	Full, PreUpgrade, PostUpgrade / AllMtasPortsStatus
BackupList	Lists the available backups. ( <code>lde-brf</code> )	YES	YES	YES	Full, PreUpgrade, PostUpgrade / BackupList
BasicSysInfo	The following is collected: <ul style="list-style-type: none"> <li>• History of the commands executed by root user (<code>history</code>)</li> <li>• List of current mount points (<code>mount</code>)</li> <li>• Loaded modules (<code>lsmod</code>)</li> <li>• NTP status (<code>ntpq -p</code>)</li> <li>• Active network connections (<code>netstat -n</code>)</li> <li>• Listening sockets (<code>netstat -tulnp</code>)</li> <li>• Process list and tree (<code>ps -Afw ,pstree -p</code>)</li> <li>• List of cached libraries (<code>ldconfig -v</code>)</li> </ul>	NO	YES	YES	None
CapsuleAbortions	Collects CA logs from <code>/opt/cdclsv/storage/cadump/</code>	YES	YES	YES	None
ClusterDumps	Collect dumps from <code>/cluster/dumps/</code>	NO	NO	YES	None
CmData	Collects CM configuration into a <code>CmDataLog.xml</code> file by <code>cmw-immconfig-export</code> .	YES	YES	YES	None
CmwCollectInfo	Collects the information from CMW by <code>cmw-collect-info</code>	NO	YES	YES	None
CoreMWStatus	Returns with the status of CoreMW. ( <code>amf-state</code> )	NO	YES	YES	All / CoreMWStatus



Table 3 Profiles and Steps in Data Collection

Name	Function	Used in Limited Profile	Used in Basic Profile	Used in Full Profile	Related HC Profile / Step
CpuLoad	Creates tables for each node in the cluster. Each table contains the current CPU-load of each processor of this node. (mpstat -P ALL).	YES	YES	YES	Full, PreUpgrade, PostUpgrade / CpuLoadOnSCs, CpuLoadOnPLs
DiameterPortsStatus	Lists the Diameter ports with their properties: port number, IP address, and SCTP address (netstat -apn -A inet,inet6).	YES	YES	YES	Full, PreUpgrade, PostUpgrade / DiameterPortsStatus
DiskStatus	Collects status of disks from every SC node by hdparm, dmsetup info, dmsetup status", and iostat.	NO	YES	YES	None
DiskUsage	Creates a table with the current disk use for every SC node by df -x tmpfs -x devtmpfs	YES	YES	YES	Full, PreUpgrade, PostUpgrade / DiskUsageOnSCs
Dmesg	Collect the content of message buffer of the kernel on each node. (dmesg)	NO	YES	YES	None
DrbdStatus	(1) Reads out the DRBD status from /proc/drbd file on every SC node.	NO	YES	YES	All / DrbdStatus
DumpFiles	Collect logdumps from /opt/cdclsv/storage/dumps	NO	NO	YES	None
EVipAndRouting	Collects the routing tables, and the eVIP tables from every (2) ALB. (telnet ^/opt/vip/bin/getactivecontrol` 25190 --> show albs , show agents [name of alb])	YES	YES	YES	All / Evip
LmScLogs	Collects LM-, SC logs from /storage/clear/lm-apr9010503/log/	NO	YES	YES	None
MemoryUsage	Creates a table with the current memory use for every node by free -m	YES	YES	YES	Full, PreUpgrade, PostUpgrade / MemoryUsageOnSCs, MemoryUsageOnPLs
MIMMOMFiles	Collects xml files from /opt/com/etc/model.	NO	YES	YES	None
Mmas	Collects mmas logs and collects logs from /opt/mmas/appserver/ from PL nodes (catalina logs and catalina-out logs).	YES	YES	YES	All / Mmas
MtasLogs	Collects MTAS-logs from /cluster/storage/no-backup/coremw/var/log/saflog/. (MTAS Application Log)	YES	YES	YES	None
MtasTrafficPoolMemoryUsage	Creates a detailed list with the current memory use for every PL node by /proc/meminfo and top -n 1 -b	YES	YES	YES	None
NelsConnectivity	Lists the NeLS configuration and the NeLS status. (from CLI)	NO	YES	YES	All / NelsConnectivity
NETCONFConnection	Collects the network configuration from every SC node by netcat localhost 9977	YES	YES	YES	All / NETCONFConnection
NetworkConnectivity	Checks the communication between SC and PLs one by one (ping).	NO	YES	YES	All / NetworkConnectivity
NodeConfiguration	Lists the installed rpms per node: collects the config-files from /cluster/nodes/[node]/etc/rpm.conf	NO	YES	YES	None
NodeOutage	Collects recovery events from /cluster/storage/no-backup/coremw/var/log/saflog/isplog/*	NO	YES	YES	All / NodeOutage



Table 3 Profiles and Steps in Data Collection

Name	Function	Used in Limited Profile	Used in Basic Profile	Used in Full Profile	Related HC Profile / Step
OperationalState	Returns the <code>mtasFunctionAdministrativeState</code> from CM.	YES	YES	YES	All / OperationalState
PMReportFiles	Collects and copies PM report files into the DC logs. (from the folder <code>/cluster/storage/no-backup/com-apr9010443/PerformanceManagementReportFiles/</code> ) The PM jobs must be started before any PM report files are created.	YES	YES	YES	None
ProcessorLogs	Collects processor-logs (messages-log) from <code>/var/log/[node]/messages</code>	YES	YES	YES	None
SecurityStatus	Lists the installed RPMs containing security SW parts.	NO	YES	YES	Full, PreUpgrade, PostUpgrade / SecurityStatus
SignalingStatus	Collects SS7 logs from <code>/opt/sign/</code> .	YES	YES	YES	None
SIPPortsStatus	Lists the IP ports used by SIP and their properties.	YES	YES	YES	Full, PreUpgrade, PostUpgrade / SIPPortsStatus
Sla	<ul style="list-style-type: none"> <li>Verifies the status of Service Level Agreement (SLA) and records the Key Performance Indicator (KPI)(2) for the Virtual Machine (VM), Core and Network Interface under the Sla Directory for the last hour.</li> <li>Moves the MtasSla PM Report files periodically to an internal location: <code>/storage/no-backup/MtasSla_PmReportFiles/</code> for consumption by HealthCheck.</li> <li>Deletes MtasSla files older than a day in the periodical housekeeping.</li> </ul>	YES	YES	YES	All / Sla
SoftwareInventory	Lists the available (3) (4) RPM /SDP files from <code>/cluster/storage/system/software/lpmsv/codearchive/</code> .	YES	YES	YES	Full, PreUpgrade, PostUpgrade / SoftwareInventory
SoftwareVersions	Lists all the installed SW components and their versions by <code>cmw-repository-list</code>	YES	YES	YES	Full, PreUpgrade, PostUpgrade / SoftwareVersions
SS7Connections	Creates a table with the SS7 connections and their properties from <code>/opt/sign/etc/signmgr.cnf</code>	YES	YES	YES	All / SS7Connections
SystemEnvironmentVariables	Lists the vDicos environmental variables.	YES	YES	YES	Full, PreUpgrade, PostUpgrade / SystemEnvironmentVariables

**Table 3** Profiles and Steps in Data Collection

Name	Function	Used in Limited Profile	Used in Basic Profile	Used in Full Profile	Related HC Profile / Step
SystemStatus	Results a verdict of the following commands: <ul style="list-style-type: none"> <li>• <code>cmw-status app</code></li> <li>• <code>cmw-status csiass</code></li> <li>• <code>cmw-status comp</code></li> <li>• <code>cmw-status node</code></li> <li>• <code>cmw-status sg</code></li> <li>• <code>cmw-status si</code></li> <li>• <code>cmw-status siass</code></li> <li>• <code>cmw-status su</code></li> <li>• <code>cmw-status pm</code></li> </ul>	NO	YES	YES	All / SystemStatus
TipcStatus	Lists TIPC status information by <code>/sbin/tipc-config -n -l -p</code> .	NO	YES	YES	None
VDicosLogs	Collects all the vDicos logs from <code>/opt/cdclsv/storage/log/</code> .	NO	YES	YES	None
VirtualDicosProcessOutage	Creates a detailed list of VMs and their status.	NO	YES	YES	All / VirtualDicosProcessOutage
VmLogs	Collects logs from <code>/opt/cdclsv/storage/log/lpmsv</code> .	YES	YES	YES	Full, PreUpgrade, PostUpgrade / VmLogs
XdmsCaiLicence	Lists XDMS Keystore Information.	YES	YES	YES	Full, PreUpgrade, PostUpgrade / XdmsCaiLicence
XdmsInstance	Results a status of the ' <code>cmw-status app</code> '.	YES	YES	YES	All / XdmsInstance
XdmsRpm	Lists the MMAS and XDMS-related packages. <code>(rpm -qa   grep -i '\(mmas\ xdms\)')</code>	YES	YES	YES	Full, PreUpgrade, PostUpgrade / XdmsRpm



## 5 Other Useful Information

Other useful information can be included in a CSR or a TR if it is easily available and there is enough time to collect it. An example of useful information is subscriber data, which is collected through the Business Support System.