

MTAS AppTrace

MTAS

USER GUIDE

Copyright

© Ericsson AB 2016–2019. 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	Prerequisites	1
1.1.1	Documents	1
2	AppTrace	3
3	AppTrace Use	5
3.1	Trace Levels for Live System	5
3.2	Trace Levels for Single Session	5
3.3	EventHistory	6
3.4	Provisioning Application Logs	6
3.5	Applog	7
3.6	One Subscriber Tracing	7
3.7	Default Trace	8
4	MtasTrace	9
4.1	Modular Trace Profiles	10
4.2	Tailing and Collecting Information	10
4.3	Default Configuration	10
4.4	Examples	11
4.4.1	General Usage	11
4.4.2	Usage Examples	12
5	AppTrace Process Types in MTAS	15
6	AppTrace Domains in MTAS	17
6.1	Domain Parameters	23
7	Trace Profiles	25
7.1	Trace Profiles for Single Session Software Trace	25
7.2	Trace Profiles for Live System Software Trace	31
7.3	Trace Profiles for Single Session Network Trace	35
7.4	Trace Profiles for Live System Network Trace	37
7.5	Trace Profiles for MTAS XDMS	40
8	Example	43





1 Introduction

This document describes how to use and apply the Application Trace (AppTrace) in the MTAS. The document lists the MTAS process types and domains that are applicable for collecting trace information. AppTrace is a service available in the vDicos execution environment. AppTrace is used for tracing vDicos applications selectively and safely.

1.1 Prerequisites

It is assumed that users of this document are familiar with performing operations within the area for Operation and Maintenance (O&M), in general.

1.1.1 Documents

For information on how to use the AppTrace, the different states, levels, and commands, refer to [AppTrace User Guide](#).

For information about how to troubleshoot the MTAS node, refer to [MTAS Troubleshooting Guideline](#).





2 AppTrace

The AppTrace is a service available in the vDicos runtime environment. The main purpose of the AppTrace is to provide practical assistance in troubleshooting vDicos applications on live systems. By using the AppTrace, an AppTrace end user can gain insight into the current behavior of an application. An AppTrace end user is an Ericsson support personnel performing operation and service provisioning tasks in a live network. Operators are not considered AppTrace end users.

Use the AppTrace with caution, since the inherent problem with observing the behavior of a system by tracing is the consumed capacity of the tracing itself. If the cost is too high, it can interfere with the primary function of the system and at worst even cause system failure.

The traces end up in the console log. The console log is a two-file wrapping log writer that writes one log per processor.

Each AppTrace statement includes the following:

- Date
- Time
- Trace level
- Filename
- Line
- Forlop identity
- Processor name
- Process identity
- Process name

Abstract domains cannot be used in the AppTrace definition because they mask underlying concrete domains and generate empty lines in console logs.

For more information about how to configure the AppTrace, refer to [vDicos AppTrace User Guide](#).

For more information about the Console Logs, refer to [MTAS Logs](#).





3 AppTrace Use

This section describes how to use the AppTrace and the defined trace levels in the MTAS. The trace level must be set to the defined trace level+1 to include the trace level in the log, for example, for DEBUG (55), the trace level is set to 56.

It is recommended to use MtasTrace profiles instead of manually assembled domain sets. For more information on handling of trace profiles, see Section 4 on page 9.

3.1 Trace Levels for Live System

Trace levels used in the MTAS for live systems are listed in ascending order in the following table:

Note: It is possible to use FORLOP (56) to trace a single user in a live system providing the load does not exceed 50%.

Trace Level	Description
CRITICAL (7)	Applied to report when a service or an interface is not working, for example, because of a faulty configuration, DNS resolution problems, or unsupported protocol versions.
CONFIG (20)	All changes of configuration data are reported in this trace level. Normally, only one trace per configuration change is reported.
MAJOR (23)	Used to report network tracing of SIP messages for a specific subscriber in the live network.
WARNING (36)	Traces to be generated whenever an unexpected event internally in the node is detected and no assert or error is to be called. This scenario occurs when source of the fault is unclear. It could be because of node internal errors or external misbehavior (transport errors, and so on), therefore assert is not appropriate to use.
PROTOCOL_ERROR (38)	Traces to be generated whenever a message received on an interface is erroneous or received in unexpected state.
MINOR (39) or MINOR_LOW (40)	Two levels used to report an unexpected event which not necessarily indicates an error. Any important information can also be logged with these trace levels.
PROTOCOL_TIMEOUT (41)	Traces to be generated whenever there is a protocol time-out.

3.2 Trace Levels for Single Session

The recommended trace level for a single session is DEBUG (55) or FORLOP (56). Trace levels 56 and 57 are incremented values to be set correspondingly. Other trace levels used in the MTAS are listed in ascending order in the following table:

Trace Level	Description
PROGRESS (43)	Function progress (call setup), interface signaling (SIP, Diameter, H248 signaling), and state transitions are reported using this trace level. Applying this trace level, it is possible to follow the progress of a call, with state transitions, through all protocol functions and services.
PROCESS (51)	Applied when report creation or termination of a dynamic and static process and creation or deletion of vDicos dialogues.



Trace Level	Description
DEBUG (55)	Applied to generate debug traces.
FORLOP (56)	Applied to generate traces when forlop has a value higher than zero.
SIP_STACK (61)	Applied for traces generated by the SIP stack. Must not be used unless a SIP problem is identified.

3.3 EventHistory

In a live system, it is not possible to use full tracing because of the number of logs generated. To help analyze problems occurring under these circumstances, important trace strings can be written to the EventHistory.

These traces are output to the console logs if there is an unexpected process termination in the following Capsule Abortion (CA) cases:

- Divide by zero
- General Protection
- Illegal opcode
- Page fault exception
- User Heap exhausted
- User Heap management error

The type of information saved in the EventHistory could, for example, be the incoming and outgoing SIP messages, h.248, expiring timers, created events, and code execution.

The MTAS EventHistory is always on.

EventHistory is serialized to record all events right from the startup of the Application Process.

It is possible to disable EventHistory serialization, resulting in EventHistory comprising only information since the last signalling event. It is not recommended to use disablement as it restricts the chance for effective troubleshooting. For more details, refer to the description of `mtasTraceDisableEhSer` CM attribute in Managed Object Model (MOM).

3.4 Provisioning Application Logs

The provisioning application logs contain information related to MTAS XDMS application processes. For more information, see [MTAS Logs](#).



3.5 Applog

The MTAS Applog log files contain information about important events and activities of MTAS processes, such as process start, process termination, process upgrade, indication of failure or success of an activity.

Each applog file has a maximum size, defined in the `MAX_FILE_SIZE` variable in the `MTAS.cfg` file located next to the applog files.

The total number of applog files is limited and are configured in the `MTAS.cfg` file in the `LOG_FULL_ACTION` variable.

The log files are named as follows:

```
MTAS_<from-timestamp1>_<to-timestamp1>.log
MTAS_<from-timestamp2>_<to-timestamp2>.log
...
MTAS_<newest-from-timestamp>.log
```

The applog files are located under the following directory:

```
/cluster/storage/no-backup/coremw/var/log/saflog/MTASAppLogs/vdicos
```

The directory is accessible from both the SCs and the PLs.

An applog entry line in an applog file contains the following:

- A sequence number
- A time stamp
- A PL name on which the logging happened
- The log level
- A descriptive field containing the logged information starting with the key or name of the logger process.

The following is an example of an applog entry:

```
5447823 2017-03-03T16:30:28.144+0100 PL-3 WA: "[PotAudit] Removed
SipCallIdMapping 3590879-24479@fd10:0:1000:106::38 ->
sip:B_mmt_normal_term_12677900@mmtnormalterm.imsas.uab.ericsson.se
```

For more information about the AppLogs, refer to [MTAS Logs](#).

3.6 One Subscriber Tracing

Subscriber tracing can be done using the existing forlop tracing supported by vDicos AppTrace. Forlop tracing makes it possible to inject a forlop identity into the AppTrace at a trace event and then to use that identity in subsequent AppTrace expressions to capture related events. The basic idea is to allow the



trace-user to trace an entire “call”, as it propagates through the system by passing through various processes on several processors. A forlop identity is simply an integer value. The default value of zero is a predefined “anonymous forlop”, that is, forlop tracing is turned off.

For MtasTrace script forlop trace usage, see Section 4.4 Examples on page 11.

For an example of using forlop, see Step 6.

For an example use of a subscriber tracing profile, refer to Section Trace Profiles in MTAS Troubleshooting Guideline.

3.7 Default Trace

The MTAS automatically writes certain number of important trace events into console log files without any manual action needed from the end user. The default trace printouts provide useful information about potential problems on a live system even if the logs do not contain more detailed and specific traces that can be produced by starting an AppTrace session.

The trace events that are printed by default are randomly selected from WARNING, CONFIG, and CRITICAL level trace events. The number of printed trace events is under control: 10 WARNING and another 10 CONFIG, and CRITICAL events are logged by default on average each minute into each console log file of each processor (if so many such events happen at all).

The following is an example of a default trace string:

```
DEF-TRACE: WARNI 7220 sip:B_barring_dndcb_
10202668@barringdndcb.imsas.uab.ericsson.se
ApplicationProcess
SubscriberLanguageCdcsHandler.cc:127
Found no language mapping entry for sv
```

The default trace feature can be controlled through CM attributes. The feature can be turned off and also an intensive default tracing can be achieved for debugging purposes.

For more details about configuring default trace, refer to the description of the `mtasTraceDefaultWarningPerMinute`, `mtasTraceDefaultCriticalConfigPerMinute`, and `mtasTraceDefaultTracePrintSubscriberKey` CM attributes in Managed Object Model (MOM).



4 MtasTrace

MTAS provides a tracing tool for easier handling of trace session and other frequently used functions. One single-line command to check, start, and stop trace sessions with the help of predefined profiles. The tool also provides utilities that are useful from a troubleshooting point of view. For instance, real-time watching and tailing of log files for an on-going trace session. Another built-in tool is the collecting and filtering feature, which helps to analyze and clean noise from the logs.

Note: The man page of the tool provides more information about the used switches and flags: `MtasTrace.sh --man`

Main features:

- Single command to start Software Traces.
- Single command to start Subscriber Traces
- CPU load monitoring
- Modular trace profiles to be able to stack services and subsystems
- Timeout parameter for terminating the trace session after the specified time
- Output of the tracing session can be tailed into a dedicated file
- Existing Appttrace log collecting, filtering, sorting

In order to successfully trace the system, important information must be supplied at the beginning of a tracing session. This information specifies which trace events are to be kept and logged, and which are to be discarded.

Processors	Processor1, Processor2 ...
Process Types	include processes defined on application level, see AppTrace Process Types
Domains	include areas defined on application level, see AppTrace Domains
Trace level	level of detail, see Trace Levels for Live System and Trace Levels for Single Session
Output direction	logs type, location and format
Forlop (optional, for subscriber trace)	trace expression: id linked to subscriber context



4.1 Modular Trace Profiles

Profiles are used to define the tracing use cases most frequently employed for troubleshooting, software debugging and subscriber tracing. The legacy tracing script also used profiles for faster and easier handling of these situations.

Using predefined tracing expressions is beneficial for both operators and system professionals, since it provides a convenient common interface, consistency between traces, and other features. The table below shows the preconfigured tracing profiles available for MtasTrace. These profiles can be stacked, meaning they can be merged together and then communicated to the system.

General	NetTrace	Subsystems
<ul style="list-style-type: none">• All• Troubleshooting	<ul style="list-style-type: none">• NetTrace-Max• NetTrace-Max-XML• NetTrace-Min• NetTrace-Min-XML	<ul style="list-style-type: none">• charging• mediafw• sip• subscriberdata• xdms

4.2 Tailing and Collecting Information

Tailing, in this context, is the name given to the real-time process of monitoring and copying lines from log files to new dedicated files. The purpose of tailing logs while tracing is to obtain a sample of unrotated, unfragmented formatted information which can, in turn, be processed by filters and other functions.

The collect command does not start a new tracing session. Instead it parses the existing logs for appttrace information, based on the profile and input type (console or applog) supplied. Collect has filtering and sorting capabilities. Sorting can be performed according to the timestamps of the merged log files.

Note: Delete generated files manually to avoid running out of disk space.
Default output directory:

```
/cluster/storage/no-backup/MtasTrace/MTAS_<CurrentTimestamp>
```

4.3 Default Configuration

Default MtasTrace values:

CPU_LOAD	70%
STOP_TIMEOUT	20 minutes
ROUTE_OUTPUT	console



CPU_MONITOR_FREQ	2 seconds
STORAGE	/cluster/storage/no-backup/MtasTrace
TRACE_LEVEL	DEBUG
APPLOG_DIR	/cluster/storage/no-backup/coremw/var/log/saflog/MTASAppLogs/vdicos
CONSOLE_DIR	/cluster/storage/no-backup/cdclsv/log/lpmsv

4.4 Examples

4.4.1 General Usage

— Software trace

```
MtasTrace profiles ... -ro <ROUTE_OUTPUT> -l <TRACE_LEVEL>
```

Starts Appttrace for ProcessTypes and Domains in TraceProfile

Most notable default values:

- ROUTE_OUTPUT: console;
- TRACE_LEVEL: DEBUG(55);
- MAX_CPU: 70%;
- STOP_TIMEOUT: 20 MINUTES

— Subscriber trace:

```
MtasTrace profiles ... -l FORLOP -user <PUB_ID>
```

Sets up a forlop appttrace for the specified originating/terminating public id.

Note: Currently, the forlop number/id is always 1234.

— Live-Tailing the Appttrace output:

```
MtasTrace profiles ... --tail
```

After starting Appttrace, tails the generated output to a separate file in real-time.

— Collect Appttrace output

```
MtasTrace profiles ... --collect --input_type console|applog  
-user <PUB_ID>
```



Does not start a new appttrace session. Instead, it parses the existing logs for AppTrace information, and creates a new file for it.

4.4.2 Usage Examples

— `MtasTrace Troubleshooting --tail`

- Starts appttrace with Troubleshooting on (default) DEBUG level
- CPU monitoring and 20 min timeout enabled
- Tails every new line from console logs
- Location of the output files: `/cluster/storage/no-backup/MtasTrace/MTAS_<CurrentTimestamp>`

— `MtasTrace Troubleshooting sip charging -sc`

- Starts appttrace with Troubleshooting, charging and sip stacked profiles on (default) DEBUG level
- CPU monitoring disabled, trace session shall be stopped manually (`MtasTrace stop`)

— `MtasTrace All -level=PROTOCOL_TIMEOUT --stop_timeout=5`

Executes a 5-minute long trace session with all domains and process types enabled on PROTOCOL_TIMEOUT level.

— `MtasTrace -user sip:<user>@<domain>`

Starts a subscriber trace with minimal configuration of domains and process types on FORLOP level.

— `MtasTrace NetTrace-Max -user sip:<user>@<domain>`

Runs a Maximum Level NetTrace in human-readable format for a single user.

— `MtasTrace sip charging --collect --input_type console --group_by_pl --sort`

Note: This example assumes that a trace session with sip and charging domains and process types has been run previously.

- Collects vm logs
- Filters sip and charging trace entries
- Merges vm logs for each PL
- Sorts the generated files by timestamps



— `MtasTrace -user sip:user1@domain1.com --collect --input_type console`

Collects trace entries of specified user from vm logs and puts to a single file in format of `subscriber_forlop_<forlop_id>`.

Note: This example assumes that subscriber trace session has been run previously.





5 AppTrace Process Types in MTAS

MTAS static processes and most dynamic processes start directly after system startup. Certain dynamic processes can however be started or restarted a long time after the system startup. The following table lists the available MTAS processes, each with a short description or comment.

The table shows the complete syntax used in AppTrace process name. This is for most process types written according to the following format:
<ProcessName> .RTID

Domain	Description or Comment
AcrStorageHandlerProc.1119246	When communication with an offline charging Server is lost, this process is used by the application process to store all offline charging ACR messages on disk.
ApplicationProcess.1060633	Application process
CarrierSelectDataProcess.1101115	Handles the MTAS configuration data for Carrier Select
ChargingBackupHandlerProcNew.1101924	Charging backup handler process
ChargingProcess.1101151	Charging process
CommBarTableMatchingProcess.1126759	Caches the configuration tables related to communication barring services and matches the input URIs against the table entries.
DiameterInstallerProc.1060881	Diameter installer process
DiameterInstallerShProc.1101088	Diameter installer Sh process
H248ProcessNew.1101915	H248 process
H248TransportProcessTypeNew.1101945	H.248 transport process
HeadProcessNew.1101927	Head process
HttpDistributorProcessNew.1101918	HTTP distributor process
ImrnAllocatorProcess.1101646	Imrn allocator process
IpmmNumNormConfigSyncProc.1105066	Handles the Number Normalization function used by the MTAS.
IpmmNumNormOamInstallerProc.1041979	Handles the Number Normalization function used by the MTAS.
Ipmm_NumberNormalization_Proc.1041857	Handles the Number Normalization function used by the MTAS.
NumberTranslationProcess.1119085	Number translation process
OamProcess.1060627	At initial startup, the subsystem instantiates the MIB (MOM) and connects to the license server and periodically checks license status.
PotAuditProcess.1101244	Pot audit process
PbxProcess.1126156	Dynamic process responsible for keeping track of the CAC counters and limits of a PBX and deciding whether to allow or reject a call.
SS7DistributorProcess.1101552	SS7DistributorProcess receives and sends CAP messages.
SS7MapDistributorProcess.1119989	Responsible for handling SS7/MAP communication.
SS7ReporterProcess.1126175	SS7Reporter collects and stores the number of active CSI sessions.
SctpOamProcess.1101098	SCTP OAM process
ShNotifyProcessNew.1101930	Sh notify process
ShRequestProcessNew.1126639	Sh request process



Domain	Description or Comment
SipDistributorProcessNew.1126603	SIP distributor process. This process is handling the UDP traffic and the TCP traffic on MTAS listening ports.
SipSocketManagementProcess.1126729	SIP SocketManagement process. This process is handling the MTAS created TCP connections and the corresponding traffic.
SipWorkerProcess.1126164	SIP distributor worker process
SubscriberDataControlProcess.1126150	Subscriber data process
SubscriberDataOamProcess.1061067	Subscriber data OAM process
TimerProcessNew.1101935	Timer process
XdmsDiameterDistributor_PT.1101094	MTAS XDMS Diameter distribution process
XdmsDiameterInstallerProc.1061368	Process on vDicos that runs the code for initializing the MTAS XDMS instance of the Diameter stack.
XdmsLinuxServer_PT.1101006	MTAS XDMS Linux server process

For available Common Components (CMCOs) processes, refer to [IMS Common Components Troubleshooting Guide](#).



6 AppTrace Domains in MTAS

To collect trace output from a trace domain, the name of the domain is added to the trace session, see Section 8 on page 43. A subset of AppTrace domains and domain groups in the MTAS are listed in the following table:

Note: Select the MTAS trace domains with four levels in the AppTrace session.
Example: `ims.mtas.sip.*`.

Domain	Description or Comment
<code>ims.mtas.charging.common</code>	This trace domain handles traces on commonly used functions for Charging.
<code>ims.mtas.common.common</code>	This trace domain handles the common parts.
<code>ims.mtas.common.headprocess</code>	This trace domain exists in only one instance. It supervises the processors to detect added, removed, or restarted processors, and provides the overall control of small restarts and the overall control of the actions required as a result of changes to the administrative states.
<code>ims.mtas.common.nc</code>	This trace domain handles the traces for Node control.
<code>ims.mtas.common.timers</code>	This trace domain handles the traces for Timers.
<code>ims.mtas.common.trace</code>	This trace domain contains the MTAS trace.
<code>ims.mtas.common.utils</code>	This trace domain handles traces on commonly used functions.
<code>ims.mtas.csi.applicationprocess</code>	This trace domain handles traces in the CSI subsystem for CAMEL interaction steps done in the application process. Typically activated together with <code>ims.mtas.csi.common</code> , <code>ims.mtas.csi.staticprocess</code> , and <code>ims.mtas.services.ncc</code> .
<code>ims.mtas.csi.common</code>	This trace domain handles traces in the CSI subsystem for CAMEL interaction steps done in the SS7DistributorProcess and Application process. Typical activated together with <code>ims.mtas.csi.staticprocess</code> , <code>ims.mtas.csi.applicationprocess</code> , and <code>ims.mtas.services.ncc</code> .
<code>ims.mtas.csi.staticprocess</code>	This trace domain handles traces in the CSI subsystem for CAMEL interaction steps done in the SS7DistributorProcess. Typically activated together with <code>ims.mtas.csi.common</code> , <code>ims.mtas.csi.applicationprocess</code> , and <code>ims.mtas.services.ncc</code> .
<code>ims.mtas.errlog.diam.rf</code>	This trace domain handles traces of the Error responses and Reason phrases for Diameter on the RF interface.
<code>ims.mtas.errlog.diam.ro</code>	This trace domain handles traces of the Error responses and Reason phrases for Diameter on the Ro interface.
<code>ims.mtas.errlog.diam.sh</code>	This trace domain handles traces of the Error responses and Reason phrases for Diameter on the SH interface.
<code>ims.mtas.errlog.h248</code>	This trace domain handles traces of the Error responses and Reason phrases on the H.248 interface.
<code>ims.mtas.errlog.sip</code>	This trace domain handles traces of the Error responses and Reason phrases on the SIP interface.
<code>ims.mtas.if.http</code>	This trace domain handles traces on the HTTP interface.
<code>ims.mtas.if.mr</code>	This trace domain handles traces on the Mr interface.
<code>ims.mtas.if.mrfc</code>	This trace domain handles traces on the MRFC interface.
<code>ims.mtas.if.namedb</code>	This trace domain handles traces on the Name DB interface.
<code>ims.mtas.if.rx</code>	This trace domain handles traces on the online charging interface.
<code>ims.mtas.if.sh</code>	This trace domain handles traces on the SH interface.
<code>ims.mtas.if.sip</code>	This trace domain handles traces on the SIP interface.
<code>ims.mtas.log.start</code>	This trace domain starts forlop tracing.
<code>ims.mtas.memory</code>	This trace domain handles traces about memory use information by Application Process.



Domain	Description or Comment
ims.mtas.mrfc.additionalmediaresource	This trace domain handles the traces on the media framework towards MRFC for additional media resource.
ims.mtas.mrfc.h248stack	This trace domain handles the traces on the H.248 stack.
ims.mtas.mrfc.mediafw	This trace domain handles the traces on the media framework towards MRFC.
ims.mtas.netio.info	This trace domain handles the traces for NetTrace on maximum level in human readable format.
ims.mtas.netio.rx	This trace domain handles the traces for NetTrace on maximum level in human readable format.
ims.mtas.netio.sip	This trace domain handles the traces for NetTrace, SIP signalling trace on maximum level in human readable format.
ims.mtas.netio.tx	This trace domain handles the traces for NetTrace on maximum level in human readable format.
ims.mtas.nettrace.info	This trace domain handles the traces for NetTrace on maximum level in machine-readable format as defined by 3GPP.
ims.mtas.nettrace.init	This trace domain handles the traces for NetTrace on maximum level in machine-readable format as defined by 3GPP.
ims.mtas.nettrace.rx	This trace domain handles the traces for NetTrace on maximum level in machine-readable format as defined by 3GPP.
ims.mtas.nettrace.sip	This trace domain handles the traces for NetTrace, SIP signalling trace on maximum level in machine-readable format as defined by 3GPP.
ims.mtas.nettrace.tx	This trace domain handles the traces for NetTrace on maximum level in machine-readable format as defined by 3GPP.
ims.mtas.nwas.foiw	This trace domain handles traces for NWA Forking Inter Working feature.
ims.mtas.nwas.priw	This trace domain handles traces for NWA Precondition Inter Working feature.
ims.mtas.oam.3pty	This trace domain handles the traces on configuration attributes for three-party.
ims.mtas.oam.abdial	This trace domain handles the traces on configuration attributes for Abbreviated Dialing.
ims.mtas.oam.aoc	This trace domain handles the traces on configuration attributes for Advise Of Charge.
ims.mtas.oam.ap	This trace domain handles the traces on configuration attributes for aggregation proxy.
ims.mtas.oam.asiw	This trace domain handles the traces on configuration attributes for application server interworking.
ims.mtas.oam.cac	This trace domain traces on configuration attributes for Call Admission Control.
ims.mtas.oam.cb	This trace domain traces on configuration attributes for Communication Barring.
ims.mtas.oam.cc	This trace domain handles traces on configuration attributes for Call Completion.
ims.mtas.oam.cdiv	This trace domain handles traces on configuration attributes for call diversion.
ims.mtas.oam.cel	This trace domain handles the traces on configuration attributes for Communication Event Logging.
ims.mtas.oam.charging	This trace domain handles traces on configuration attributes for Charging.
ims.mtas.oam.comdetails	This trace domain handles traces on configuration attributes for Communication details.
ims.mtas.oam.common	This trace domain handles traces on configuration attributes for commonly used parts.
ims.mtas.oam.conf	This trace domain handles traces on configuration attributes for Conference.
ims.mtas.oam.cpc	This trace domain handles traces on configuration attributes for Call party category.
ims.mtas.oam.cps	This trace domain handles traces on configuration attributes for Carrier Pre-Select.
ims.mtas.oam.cr	This trace domain handles the traces on configuration attributes for call return.
ims.mtas.oam.cs	This trace domain handles traces on configuration attributes for Carrier Select.
ims.mtas.oam.csa	This trace domain handles the traces on configuration attributes for the Communication Setup Announcement.



Domain	Description or Comment
ims.mtas.oam.csi	This trace domain handles traces on configuration attributes for CSI subsystem, Circuit Switch Interworking.
ims.mtas.oam.csvfb	This trace domain handles traces on configuration attributes for Carrier Select Video Fallback.
ims.mtas.oam.cug	This trace domain handles the traces on configuration attributes for close user group.
ims.mtas.oam.cw	This trace domain handles traces on configuration attributes for Communication Waiting.
ims.mtas.oam.dialplan	This trace domain handles traces on configuration attributes for Dial Plan.
ims.mtas.oam.dr	This trace domain handles the traces on configuration attributes for Distinctive Ring.
ims.mtas.oam.dtm	This trace domain handles traces on configuration attributes for Dial Tone Management.
ims.mtas.oam.ect	This trace domain handles traces on configuration attributes for Explicit Communication Transfer.
ims.mtas.oam.fcd	This trace domain handles traces on configuration attributes for Flexible Communication Distribution.
ims.mtas.oam.fsfs	This trace domain handles the traces on configuration attributes for file transfer.
ims.mtas.oam.ga	This trace domain handles traces on configuration attributes for Generic Announcement.
ims.mtas.oam.gls	This trace domain handles traces on configuration attributes for Global Licensing Service.
ims.mtas.oam.gm	This trace domain handles traces on configuration attributes for Gateway Model.
ims.mtas.oam.hold	This trace domain handles traces on configuration attributes for hold.
ims.mtas.oam.hotline	This trace domain handles the traces on configuration attributes for hotline.
ims.mtas.oam.idpres	This trace domain handles traces on configuration attributes for Identity Presentation.
ims.mtas.oam.jc	This trace domain handles traces on configuration attributes for Japanese Charging.
ims.mtas.oam.mcld	This trace domain handles traces on configuration attributes for Malicious Communication Identification.
ims.mtas.oam.mmt	This trace domain handles traces on configuration attributes for Multimedia Telephony.
ims.mtas.oam.mrfc	This trace domain handles traces on configuration attributes for the MRFC.
ims.mtas.oam.na	This trace domain handles traces on configuration attributes for network announcement.
ims.mtas.oam.namedb	This trace domain handles traces on configuration attributes for nameDb.
ims.mtas.oam.ncc	This trace domain handles traces on configuration attributes for Northbound Call Control.
ims.mtas.oam.numnorm	This trace domain handles traces on configuration attributes for Number Normalization.
ims.mtas.oam.numport	This trace domain handles the traces on configuration attributes for Number Portability.
ims.mtas.oam.nw	This trace domain handles traces on configuration attributes for NWAS (Network AS).
ims.mtas.oam.oct	This trace domain handles the traces on configuration attributes for Operator Controlled Transfer.
ims.mtas.oam.priocall	This trace domain handles traces on configuration attributes for priority call.
ims.mtas.oam.px	This trace domain handles traces on configuration attributes for Parlay-X.
ims.mtas.oam.rbt	This trace domain handles the traces on configuration attributes for Ring Back Tone.
ims.mtas.oam.scc	This trace domain handles the traces on configuration attributes for Supplementary Service Code.
ims.mtas.oam.sds	This trace domain handles the traces on configuration attributes for Service Domain Selection.
ims.mtas.oam.sh	This trace domain handles traces on configuration attributes for SH.
ims.mtas.oam.sip	This trace domain handles traces on configuration attributes for SIP.
ims.mtas.oam.snd	This trace domain handles traces on configuration attributes for Short Number Dialing.
ims.mtas.oam.ssc	This trace domain handles traces on configuration attributes for supplementary service codes.



Domain	Description or Comment
ims.mtas.oam.st	This trace domain handles traces on configuration attributes for STAS (SIP Trunking AS).
ims.mtas.oam.subscriberdata	This trace domain handles traces on configuration attributes for subscriber data.
ims.mtas.oam.tads	This trace domain handles the traces on configuration attributes for Terminating Access Domain Selection.
ims.mtas.oam.vtp	This trace domain handles traces on configuration attributes for Virtual Telephony Provider.
ims.mtas.oam.xdms	This trace domain handles traces on configuration attributes for XDMS.
ims.mtas.px.dispatcher	This trace domain handles traces on the pxDispatcher.
ims.mtas.services.3pty	This trace domain handles traces on the Three Party service.
ims.mtas.services.abdial	This trace domain handles traces on the Abbreviated Dialing service.
ims.mtas.services.aoc	This trace domain handles traces on the Advice of Charge service.
ims.mtas.services.asiw	This trace domain handles traces on the Application Server InterWorking service.
ims.mtas.services.cac	This trace domain handles traces on the Call Admission Control service.
ims.mtas.services.cat	This trace domain handles traces on the Customized Alerting Tones service.
ims.mtas.services.cb	This trace domain handles traces on the Communication Barring service.
ims.mtas.services.cc	This trace domain handles traces on the Call Completion service.
ims.mtas.services.cd	This trace domain handles traces on the Communication Distribution service.
ims.mtas.services.cel	This trace domain handles traces on the Communication Event Logging.
ims.mtas.services.cddiv	This trace domain handles traces on the Communication Diversion service.
ims.mtas.services.charging	This trace domain handles traces on the Charging service.
ims.mtas.services.common	This trace domain handles traces on the commonly used functions for services.
ims.mtas.services.conf	This trace domain handles traces on the Conference service.
ims.mtas.services.conference	This trace domain handles traces on the Conference service.
ims.mtas.services.cpc	This trace domain handles traces on the Call party category service.
ims.mtas.services.cps	This trace domain handles traces on the Carrier Pre-Select service.
ims.mtas.services.cr	This trace domain handles traces on the Call Return service.
ims.mtas.services.cs	This trace domain handles traces on the Carrier Select service.
ims.mtas.services.csa	This trace domain handles traces on the Communication Setup Announcement service.
ims.mtas.services.csvfb	This trace domain handles traces on the Carrier Select Video Fallback service.
ims.mtas.services.cug	This trace domain handles traces on the Closed User Group service.
ims.mtas.services.cw	This trace domain handles traces on the Communication Waiting service.
ims.mtas.services.dbl	This trace domain handles traces on the Dynamic Black List service.
ims.mtas.services.defaultmessageservice	This trace domain handles traces on the DefaultMessage service.
ims.mtas.services.defaultpsiservice	This trace domain handles traces on the DefaultPsi service.
ims.mtas.services.defaultsubscribeservice	This trace domain handles traces on the DefaultSubscribe service.
ims.mtas.services.dialogeventnotifierservice	This trace domain handles traces on the Dialog Event Notifier service.
ims.mtas.services.dialogstatedialogagent service	This trace domain handles traces on the Dialog State Dialog Agent service.
ims.mtas.services.dr	This trace domain handles traces on the Distinctive Ring service.



Domain	Description or Comment
ims.mtas.services.dtm	This trace domain handles traces on the Dial Tone Management service.
ims.mtas.services.ect	This trace domain handles traces on the Explicit Communication Transfer services.
ims.mtas.services.emergencystate	This trace domain handles traces on the Emergency State service.
ims.mtas.services.gls	This trace domain handles traces on the GLS service.
ims.mtas.services.gm	This trace domain handles traces on the Gateway Model service.
ims.mtas.services.hold	This trace domain handles traces on the Hold service.
ims.mtas.services.hotline	This trace domain handles traces on the hotline service.
ims.mtas.services.idpres	This trace domain handles traces on the Identity Presentation service.
ims.mtas.services.jc	This trace domain handles traces on the Japanese Charging service.
ims.mtas.services.mcid	This trace domain handles traces on the Malicious Communication IDentification service.
ims.mtas.services.mmt	This trace domain handles traces on the Multimedia Telephony service.
ims.mtas.services.multipersona.hh	This trace domain handles traces on the Multi-Persona service.
ims.mtas.services.na	This trace domain handles traces on the Network Announcement service.
ims.mtas.services.ncc	This trace domain handles traces on the Northbound Call Control service.
ims.mtas.services.numbertranslation	This trace domain handles traces on the Number translation service.
ims.mtas.services.numnorm	This trace domain handles traces on the Number Normalization service.
ims.mtas.services.numport	This trace domain handles traces on the Number Portability service.
ims.mtas.services.ocb	This trace domain handles traces on the Outgoing Communication Barring service.
ims.mtas.services.oct	This trace domain handles traces on the Operator Controlled Transfer service.
ims.mtas.services.pemiwf	This trace domain handles traces on the P-Early-Media Interworking Function service.
ims.mtas.services.prenn	This trace domain handles traces on the PreNumberNormalisation Service.
ims.mtas.services.presence	This trace domain handles traces on the Presence Service.
ims.mtas.services.priocall	This trace domain handles traces on the Priority Call service.
ims.mtas.services.px3pcc	This trace domain handles traces on the Parlay-X Third-Party Call Control service.
ims.mtas.services.pxcn	This trace domain handles traces on the Parlay-X Common Name service.
ims.mtas.services.rbt	This trace domain handles traces on the Ring Back Tone service.
ims.mtas.services.scc	This trace domain handles traces on the Service Centralization and Continuity service.
ims.mtas.services.scn	This trace domain handles traces on the Subscriber Credit Notification service.
ims.mtas.services.sds	This trace domain handles traces on the Service Domain Selection service.
ims.mtas.services.servicefw	This trace domain handles traces on the Service Framework service.
ims.mtas.services.snd	This trace domain handles traces on the Short Number Dialing service.
ims.mtas.services.srvcc	This trace domain handles traces on the Single Radio Voice Call Continuity service.
ims.mtas.services.ssc	This trace domain handles traces on the Supplementary Service Codes service including the DICOS part of the Generic SSC feature.
ims.mtas.services.stod	This trace domain handles traces on the Session Transfer to Own Device service.
ims.mtas.services.subdata	This trace domain handles traces on the Subscriber Data service.
ims.mtas.services.ta	This trace domain handles traces on the TestAnnouncement service.
ims.mtas.services.tads	This trace domain handles traces on the Terminating Access Domain Selection service.



Domain	Description or Comment
ims.mtas.services.telsessionincoming	This trace domain handles traces on the Telsession Incoming service.
ims.mtas.services.ucrouting	This trace domain handles traces on the Unified Communication Routing service.
ims.mtas.services.uloc	This trace domain handles traces on the user location service.
ims.mtas.services.unreg	This trace domain handles traces on the unregistered Subscriber service.
ims.mtas.services.vm	This trace domain handles traces on the Voice Mail service.
ims.mtas.sip.dispatcher	This trace domain handles traces on the sipDispatcher.
ims.mtas.sip.distributor	This trace domain handles traces on the SipDistributor.
ims.mtas.sip.sipstack	This trace domain handles traces on the SIP stack.
ims.mtas.sip.socketmanagement	This trace domain handles traces on the SipSocketManagementProcess.
ims.mtas.st.core	This trace domain handles traces in the core parts of the STAS (SIP Trunking AS).
ims.mtas.st.services	This trace domain handles traces for all services in ST AS.
ims.mtas.subscriberdata.conferencedata	This trace domain handles traces on the conference part of the subscriber data.
ims.mtas.subscriberdata.core	This trace domain handles traces on the core of subscriber data.
ims.mtas.subscriberdata.servicedata	This trace domain handles traces on the service data of the subscriber data.
ims.mtas.subscriberdata.servicenumdata	This trace domain handles traces on Number Normalization part of the subscriber data.
ims.mtas.subscriberdata.stasdata	This trace domain handles traces in the STAS (SIP Trunking AS) part of subscriber data.
ims.mtas.subscriberdata.userdata	This trace domain handles traces on the user data of the subscriber data.
ims.mtas.tools.mprof	This trace domain handles traces on the Memory profiler.
ims.mtas.utils.potaudit	This trace domain handles traces on the POT Audit.
ims.mtas.xdms.cai3g	This trace domain handles traces on the CAI3G operator provisioning interface.
ims.mtas.xdms.charging	This trace domain handles traces for Charging for Ut interface updates.
ims.mtas.xdms.dao	This trace domain handles traces for common functions for the XML service data access.
ims.mtas.xdms.dtm	This trace domain handles traces for the XDMS part of the Dial Tone Management feature.
ims.mtas.xdms.genssc	This trace domain handles traces for the XDMS part of the Generic SSC feature.
ims.mtas.xdms.license	This trace domain handles traces for XDMS licenses.
ims.mtas.xdms.misc	This trace domain handles traces for commonly used functions in XDMS.
ims.mtas.xdms.numnorm	This trace domain handles traces for XDMS Number Normalization.
ims.mtas.xdms.oam	This trace domain handles traces for O&M functions (configuration and performance management) within the XDMS.
ims.mtas.xdms.rules	This trace domain handles traces on application rule checks for the XML service data consistency.
ims.mtas.xdms.timings	This trace domain handles traces for timer code within the XDMS.
ims.mtas.xdms.trace	This trace domain handles traces for XDMS trace.
ims.mtas.xdms.triggers	This trace domain handles traces for trigger operation including triggering of Dial Tone Management (DTM).
ims.mtas.xdms.utility	This trace domain handles traces for commonly used utility support functions.
ims.mtas.xdms.xcap	This trace domain handles traces for the Ut XCAP interface for self-administration of XML service data.
ims.mtas.xdms.xconconferenceinfo	This trace domain handles traces for the XDMS part of the conference feature.



Domain	Description or Comment
ims.mtas.xdms.xdp	This trace domain handles traces for inter-process communication within the XDMS.
ims.mtas.xdmsval.validation	This trace domain handles traces for XDMS validation.

For available Common Components (CMCOs) domains, refer to [IMS Common Components Troubleshooting Guide](#).

6.1 Domain Parameters

The Msg parameter is available for all CMCO and most MTAS domains, exceptions are the ones listed.

Domain	Parameters
ims.mtas.errlog.sip	\$FromID, \$Response, \$Reason, \$SessSide
ims.mtas.errlog.h248	\$Response, \$Reason
ims.mtas.errlog.diam.sh	\$Response, FromID, \$Response, \$Reason
ims.mtas.errlog.diam.rf	\$FromID, \$Response, \$Reason, \$SessSide
ims.mtas.errlog.diam.ro	\$Reason, \$SessSide





7 Trace Profiles

Preconfigured trace profiles can be used for either a single session or a live system.

7.1 Trace Profiles for Single Session Software Trace

Table 1 shows the preconfigured trace profiles for a single session software trace:

The following profiles route their output to the console log.

Table 1 Trace Profiles for Single Session Software Trace

Profile Name	MtasProgressTrace43
Description	Dumps information about any user. ⁽¹⁾
Trace level	PROGRESS (43)



Table 1 Trace Profiles for Single Session Software Trace

Processes	ApplicationProcess.1060633 H248ProcessNew.1101915 TimerProcessNew.1101935 DiameterInstallerProc.1060881 ChargingBackupHandlerProcNew.1101924 HeadProcessNew.1101927 SubscriberDataOamProcess.1061067 SipDistributorProcessNew.1126603 SipSocketManagementProcess.1126729 ShRequestProcessNew.1126639 DiameterInstallerShProc.1101088 ShNotifyProcessNew.1101930 ChargingProcess.1101151 PotAuditProcess.1101244 H248TransportProcessTypeNew.1101945 HttpDistributorProcessNew.1101918 MomCMInstallerProcess.1060808 SS7DistributorProcess.1101552 XdmsDiameterDistributor_PT.1101094 XdmsLinuxServer_PT.1101006 OamProcess.1060627 AcrStorageHandlerProc.1119246 XdmsDiameterInstallerProc.1061368 NumberTranslationProcess.1119085 CarrierSelectDataProcess.1101115 CommBarTableMatchingProcess.1126759 ImrnAllocatorProcess.1101646 PbxProcess.1126156 SS7MapDistributorProcess.1119989 SS7ReporterProcess.1126175 SipWorkerProcess.1126164 SubscriberDataControlProcess.1126150 SctpOamProcess.1101098
-----------	---



Table 1 Trace Profiles for Single Session Software Trace

Domains	ims.mtas.charging.common ims.mtas.common.* ims.mtas.csi.* ims.mtas.errlog.diam.sh ims.mtas.errlog.diam.rf ims.mtas.errlog.diam.ro ims.mtas.errlog.h248 ims.mtas.errlog.sip ims.mtas.if.* ims.mtas.mrfc.* ims.mtas.oam.* ims.mtas.px.dispatcher ims.mtas.services.* ims.mtas.sip.* ims.mtas.st.* ims.mtas.subscriberdata.* ims.mtas.utils.* ims.mtas.tools.mprof ims.mtas.xdms.* ims.mtas.xdmsval.validation ims.mtas.nwas.*
Profile Name	MtasLoggingOfErrorResponsesDebug55
Description	Dumps information about the Error Responses received on the H.248, Rf, Ro, Sh, and SIP interfaces and debug traces.
Trace level	DEBUG (55)
Processes	ApplicationProcess.1060633 SipDistributorProcessNew.1126603 SipSocketManagementProcess.1126729 ShNotifyProcessNew.1101930 ChargingProcess.1101151 H248ProcessNew.1101915 H248TransportProcessTypeNew.1101945



Table 1 Trace Profiles for Single Session Software Trace

Domains	ims.mtas.charging.common ims.mtas.common.common ims.mtas.common.utils ims.mtas.errlog.diam.rf ims.mtas.errlog.diam.ro ims.mtas.errlog.diam.sh ims.mtas.errlog.h248 ims.mtas.errlog.sip ims.mtas.if.* ims.mtas.sip.dispatcher ims.mtas.sip.distributor ims.mtas.sip.socketmanagement
Profile Name	MtasUserTraceForFromId56, MtasUserTraceToId56, or MtasUserTraceForFromIdAndToId56
Description	Dumps information about a particular user or users.
Trace level	FORLOP (56)
Processes	ApplicationProcess.1060633 CommBarTableMatchingProcess.1126759
Domains	ims.mtas.common.common ims.mtas.sip.dispatcher ims.mtas.services.* ims.mtas.st.*
Profile Name	MtasDebugTrace55
Description	Dumps information about any user. ⁽¹⁾
Trace level	DEBUG (55)



Table 1 Trace Profiles for Single Session Software Trace

Processes	ApplicationProcess.1060633 H248ProcessNew.1101915 TimerProcessNew.1101935 DiameterInstallerProc.1060881 ChargingBackupHandlerProcNew.1101924 HeadProcessNew.1101927 SubscriberDataOamProcess.1061067 SipDistributorProcessNew.1126603 SipSocketManagementProcess.1126729 ShRequestProcessNew.1126639 DiameterInstallerShProc.1101088 ShNotifyProcessNew.1101930 ChargingProcess.1101151 PotAuditProcess.1101244 H248TransportProcessTypeNew.1101945 HttpDistributorProcessNew.1101918 MomCMInstallerProcess.1060808 SS7DistributorProcess.1101552 XdmsDiameterDistributor_PT.OU_XdmsDiameterDistributor XdmsLinuxServer_PT.1101006 OamProcess.1060627 AcrStorageHandlerProc.1119246 XdmsDiameterInstallerProc.1061368 NumberTranslationProcess.1119085 CarrierSelectDataProcess.1101115 CommBarTableMatchingProcess.1126759 ImrnAllocatorProcess.1101646 PbxProcess.1126156 SS7MapDistributorProcess.1119989 SS7ReporterProcess.1126175 SipWorkerProcess.1126164 SubscriberDataControlProcess.1126150 SctpOamProcess.1101098
-----------	--



Table 1 Trace Profiles for Single Session Software Trace

Domains	ims.mtas.errlog.sip ims.mtas.errlog.h248 ims.mtas.errlog.diam.sh ims.mtas.errlog.diam.rf ims.mtas.errlog.diam.ro ims.mtas.common.* ims.mtas.charging.common ims.mtas.if.* ims.mtas.services.* ims.mtas.px.dispatcher ims.mtas.csi.* ims.mtas.mrfc.* ims.mtas.oam.* ims.mtas.sip.* ims.mtas.subscriberdata.* ims.mtas.utils.* ims.mtas.xdms.* ims.mtas.xdmsval.validation ims.mtas.tools.mprof ims.mtas.st.* ims.mtas.nwas.*
Profile Name	MtasSccAsTrace55
Description	Dumps information about any user in SCC AS
Trace level	DEBUG (55)



Table 1 Trace Profiles for Single Session Software Trace

Processes	ApplicationProcess.1060633 CommBarTableMatchingProcess.1126759 SipDistributorProcessNew.1126603 SipSocketManagementProcess.1126729 SS7DistributorProcess.1101552 SS7MapDistributorProcess.1119989
Domains	ims.mtas.charging.common ims.mtas.common.common ims.mtas.common.utils ims.mtas.errlog.sip ims.mtas.errlog.diam.sh ims.mtas.errlog.diam.rf ims.mtas.if.sh ims.mtas.services.common ims.mtas.services.scc ims.mtas.services.sds ims.mtas.services.servicefw ims.mtas.services.srvcc ims.mtas.services.subdata ims.mtas.services.tads ims.mtas.csi.* ims.mtas.sip.* ims.mtas.subscriberdata.*

(1) Only recommended to be used for single session.

7.2 Trace Profiles for Live System Software Trace

Table 2 shows the preconfigured trace profiles for a live system software trace:

The following profiles route their output to the console log.

Table 2 Trace Profiles for Live Stream Software Trace

Profile Name	MtasWarningTrace36
Description	Dumps information about unexpected events and EventHistory.
Trace level	WARNING (36)



Table 2 Trace Profiles for Live Stream Software Trace

Processes	ApplicationProcess.1060633 H248ProcessNew.1101915 TimerProcessNew.1101935 DiameterInstallerProc.1060881 ChargingBackupHandlerProcNew.1101924 HeadProcessNew.1101927 SubscriberDataOamProcess.1061067 SipDistributorProcessNew.1126603 SipSocketManagementProcess.1126729 ShRequestProcessNew.1126639 DiameterInstallerShProc.1101088 ShNotifyProcessNew.1101930 ChargingProcess.1101151 PotAuditProcess.1101244 H248TransportProcessTypeNew.1101945 HttpDistributorProcessNew.1101918 MomCMInstallerProcess.1060808 SS7DistributorProcess.1101552 XdmsDiameterDistributor_PT.1101094 XdmsLinuxServer_PT.1101006 OamProcess.1060627 AcrStorageHandlerProc.1119246 XdmsDiameterInstallerProc.1061368 NumberTranslationProcess.1119085 CarrierSelectDataProcess.1101115 CommBarTableMatchingProcess.1126759 ImrnAllocatorProcess.1101646 PbxProcess.1126156 SS7MapDistributorProcess.1119989 SS7ReporterProcess.1126175 SipWorkerProcess.1126164 SubscriberDataControlProcess.1126150 SctpOamProcess.1101098
-----------	---



Table 2 Trace Profiles for Live Stream Software Trace

Domains	ims.mtas.errlog.sip ims.mtas.errlog.h248 ims.mtas.errlog.diam.sh ims.mtas.errlog.diam.rf ims.mtas.errlog.diam.ro ims.mtas.common.* ims.mtas.charging.common ims.mtas.if.* ' ims.mtas.services.* ims.mtas.px.dispatcher ims.mtas.csi.* ims.mtas.mrfc.* ims.mtas.oam.* ims.mtas.sip.* ims.mtas.subscriberdata.* ims.mtas.utils.* ims.mtas.xdms.* ims.mtas.xdmsval.validation ims.mtas.tools.mprof ims.mtas.st.* ims.mtas.nwas.*
Profile Name	MtasLoggingOfErrorResponses40
Description	Dumps information about the Error Responses received on the H.248, Rf, Ro, Sh, and SIP interfaces.
Trace level	40
Processes	ApplicationProcess.1060633 SipDistributorProcessNew.1126603 ShNotifyProcessNew.1101930 ChargingProcess.1101151 H248ProcessNew.1101915 H248TransportProcessTypeNew.1101945
Domains	ims.mtas.errlog.sip ims.mtas.errlog.h248 ims.mtas.errlog.diam.sh ims.mtas.errlog.diam.rf ims.mtas.errlog.diam.ro
Profile Name	MtasProtocolTimeoutTrace41
Description	Dumps information about unexpected events provided by MTAS.



Table 2 Trace Profiles for Live Stream Software Trace

Trace level	PROTOCOL_TIMEOUT (41)
Processes	ApplicationProcess.1060633 H248ProcessNew.1101915 TimerProcessNew.1101935 DiameterInstallerProc.1060881 ChargingBackupHandlerProcNew.1101924 HeadProcessNew.1101927 SubscriberDataOamProcess.1061067 SipDistributorProcessNew.1126603 SipSocketManagementProcess.1126729 ShRequestProcessNew.1126639 DiameterInstallerShProc.1101088 ShNotifyProcessNew.1101930 ChargingProcess.1101151 PotAuditProcess.1101244 H248TransportProcessTypeNew.1101945 HttpDistributorProcessNew.1101918 MomCMInstallerProcess.1060808 SS7DistributorProcess.1101552 XdmsDiameterDistributor_PT.1101094 XdmsLinuxServer_PT.1101006 OamProcess.1060627 AcrStorageHandlerProc.1119246 XdmsDiameterInstallerProc.1061368



Table 2 Trace Profiles for Live Stream Software Trace

	NumberTranslationProcess.1119085 CarrierSelectDataProcess.1101115 CommBarTableMatchingProcess.1126759 ImrnAllocatorProcess.1101646 PbxProcess.1126156 SS7MapDistributorProcess.1119989 SS7ReporterProcess.1126175 SipWorkerProcess.1126164 SubscriberDataControlProcess.1126150 SctpOamProcess.1101098
Domains	ims.mtas.errlog.sip ims.mtas.errlog.h248 ims.mtas.errlog.diam.sh ims.mtas.errlog.diam.rf ims.mtas.errlog.diam.ro ims.mtas.common.* ims.mtas.charging.common ims.mtas.if.* ims.mtas.services.* ims.mtas.px.dispatcher ims.mtas.csi.* ims.mtas.mrfc.* ims.mtas.oam.* ims.mtas.sip.* ims.mtas.subscriberdata.* ims.mtas.utils.* ims.mtas.xdms.* ims.mtas.xdmsval.validation ims.mtas.tools.mprof ims.mtas.st.* ims.mtas.nwas.*

7.3 Trace Profiles for Single Session Network Trace

Table 3 shows the preconfigured trace profiles for a single session network trace:

The following profiles route their output to the applog.



Table 3 Trace Profiles for Single Session Network Trace

Profile Name	MtasNetIoMinLevel0rigUser56
Description	NetTrace, SIP signalling trace on minimum level including debug traces in human readable format.
Trace level	FORLOP (56)
Processes	ApplicationProcess.1060633 ApplicationProcess.1060633 SipDistributorProcessNew.1126603 SipSocketManagementProcess.1126729
Domains	ims.mtas.netio.info ims.mtas.netio.sip ims.mtas.common.utils ims.mtas.sip.dispatcher ims.mtas.sip.distributor ims.mtas.sip.socketmanagement ims.mtas.common.common
Profile Name	MtasNetIoMaxLevel0rigUser56
Description	NetTrace, SIP signalling trace on maximum level including debug traces in human readable format.
Trace level	FORLOP (56)
Processes	ApplicationProcess.1060633 SipDistributorProcessNew.1126603 SipSocketManagementProcess.1126729
Domains	ims.mtas.netio.info ims.mtas.netio.rx ims.mtas.netio.tx ims.mtas.common.utils ims.mtas.sip.dispatcher ims.mtas.sip.distributor ims.mtas.sip.socketmanagement ims.mtas.common.common
Profile Name	MtasNetTraceMinLevel0rigUser56
Description	NetTrace, SIP signalling trace on minimum level including debug traces.
Trace level	FORLOP (56)
Processes	ApplicationProcess.1060633 SipDistributorProcessNew.1126603 SipSocketManagementProcess.1126729



Table 3 Trace Profiles for Single Session Network Trace

Domains	ims.mtas.nettrace.info ims.mtas.nettrace.sip ims.mtas.common.utils ims.mtas.sip.dispatcher ims.mtas.sip.distributor ims.mtas.sip.socketmanagement ims.mtas.common.common
Profile Name	MtasNetTraceMaxLevelOrigUser56
Description	NetTrace, SIP signalling trace on maximum level including debug traces.
Trace level	FORLOP (56)
Processes	ApplicationProcess.1060633 SipDistributorProcessNew.1126603 SipSocketManagementProcess.1126729
Domains	ims.mtas.nettrace.info ims.mtas.nettrace.rx ims.mtas.nettrace.tx ims.mtas.common.utils ims.mtas.sip.dispatcher ims.mtas.sip.distributor ims.mtas.sip.socketmanagement ims.mtas.common.common

7.4 Trace Profiles for Live System Network Trace

Table 4 shows the preconfigured trace profiles for a live system network trace:

The following profiles route their output to the applog.

Table 4 Trace Profiles for Live System Network Trace

Profile Name	MtasNetIoMinLevelOrigUser23
Description	NetTrace, SIP signalling trace on minimum level in human readable format
Trace level	MAJOR (23)
Processes	ApplicationProcess.1060633 SipDistributorProcessNew.1126603
Domains	ims.mtas.netio.info ims.mtas.netio.sip
Profile Name	MtasNetIoMaxLevelOrigUser23



Table 4 Trace Profiles for Live System Network Trace

Description	NetTrace, SIP signalling trace on maximum level in human readable format.
Trace level	MAJOR (23)
Processes	ApplicationProcess.1060633 SipDistributorProcessNew.1126603
Domains	ims.mtas.netio.info ims.mtas.netio.rx ims.mtas.netio.tx
Profile Name	MtasNetIoMinLevelTermUser23
Description	NetTrace, SIP signalling trace on minimum level in human readable format.
Trace level	MAJOR (23)
Processes	ApplicationProcess.1060633 SipDistributorProcessNew.1126603
Domains	ims.mtas.netio.info ims.mtas.netio.sip
Profile Name	MtasNetIoMaxLevelTermUser23
Description	NetTrace, SIP signalling trace on maximum level in human readable format.
Trace level	MAJOR (23)
Processes	ApplicationProcess.1060633 SipDistributorProcessNew.1126603
Domains	ims.mtas.netio.info ims.mtas.netio.rx ims.mtas.netio.tx
Profile Name	MtasNetIoMinLevelOrigUserTermUser23
Description	NetTrace, SIP signalling trace on minimum level in human readable format.
Trace level	MAJOR (23)
Processes	ApplicationProcess.1060633 SipDistributorProcessNew.1126603
Domains	ims.mtas.netio.info ims.mtas.netio.sip
Profile Name	MtasNetIoMaxLevelOrigUserTermUser23
Description	NetTrace, SIP signalling trace on maximum level in human readable format.
Trace level	MAJOR (23)
Processes	ApplicationProcess.1060633 SipDistributorProcessNew.1126603



Table 4 Trace Profiles for Live System Network Trace

Domains	ims.mtas.nettrace.init ims.mtas.netio.info ims.mtas.netio.rx ims.mtas.netio.tx
Profile Name	MtasNetTraceMinLevelOrigUser23
Description	NetTrace, SIP signalling trace on minimum level in machine readable format.
Trace level	MAJOR (23)
Processes	ApplicationProcess.1060633 SipDistributorProcessNew.1126603
Domains	ims.mtas.nettrace.init ims.mtas.nettrace.info ims.mtas.nettrace.sip
Profile Name	MtasNetTraceMaxLevelOrigUser23
Description	NetTrace, SIP signalling trace on maximum level in machine-readable format
Trace level	MAJOR (23)
Processes	ApplicationProcess.1060633 SipDistributorProcessNew.1126603
Domains	ims.mtas.nettrace.info ims.mtas.nettrace.rx ims.mtas.nettrace.tx
Profile Name	MtasNetTraceMinLevelTermUser23
Description	NetTrace, SIP signalling trace on minimum level in machine-readable format as defined by 3GPP.
Trace level	MAJOR (23)
Processes	ApplicationProcess.1060633 SipDistributorProcessNew.1126603
Domains	ims.mtas.nettrace.info ims.mtas.nettrace.sip
Profile Name	MtasNetTraceMaxLevelTermUser23
Description	NetTrace, SIP signalling trace on maximum level in machine-readable format as defined by 3GPP.
Trace level	MAJOR (23)
Processes	ApplicationProcess.1060633 SipDistributorProcessNew.1126603



Table 4 Trace Profiles for Live System Network Trace

Domains	ims.mtas.nettrace.info ims.mtas.nettrace.rx ims.mtas.nettrace.tx
Profile Name	MtasNetTraceMinLevelOrigUserTermUser23
Description	NetTrace, SIP signalling trace on minimum level in machine-readable format as defined by 3GPP. Used for tracing call between specific users A and B.
Trace level	MAJOR (23)
Processes	ApplicationProcess.1060633 SipDistributorProcessNew.1126603
Domains	ims.mtas.nettrace.info ims.mtas.nettrace.sip
Profile Name	MtasNetTraceMaxLevelOrigUserTermUser23
Description	NetTrace, SIP signalling trace on maximum level in machine-readable format as defined by 3GPP. Used for tracing call between specific users A and B.
Trace level	MAJOR (23)
Processes	ApplicationProcess.1060633 SipDistributorProcessNew.1126603
Domains	ims.mtas.nettrace.info ims.mtas.nettrace.rx ims.mtas.nettrace.tx

7.5 Trace Profiles for MTAS XDMS

Table 5 shows the preconfigured trace profiles for MTAS XDMS:

The following profiles route their output to the console log.

Table 5 Trace Profiles for MTAS XDMS

Profile Name	MtasXdmsDebugTrace55
Description	Dumps information about any user. (1)
Trace level	55



Table 5 Trace Profiles for MTAS XDMS

Processes	ApplicationProcess.1060633 SubscriberDataOamProcess.1061067 SubscriberDataControlProcess.1126150 Ipmm_NumberNormalization_Proc.1041857 IpmmNumNormOamInstallerProc.1041979 IpmmNumNormConfigSyncProc.1105066 ShRequestProcessNew.1126639 XdmsDiameterDistributor_PT.1101094 XdmsLinuxServer_PT.1101006
Domains	ims.mtas.if.sh ims.mtas.services.numnorm ims.mtas.services.numbertranslation ims.mtas.services.conf ims.mtas.services.conference ims.mtas.oam.numnorm ims.mtas.oam.xdms ims.mtas.subscriberdata.servicenumdata ims.mtas.xdms.* ims.mtas.xdmsval.validation ims.mtas.errlog.diam.sh
Profile Name	MtasXdmsWarningTrace36
Description	Dumps information about any user. (1)
Trace level	36



Table 5 Trace Profiles for MTAS XDMS

Processes	ApplicationProcess.1060633 SubscriberDataOamProcess.1061067 SubscriberDataControlProcess.1126150 Ipmm_NumberNormalization_Proc.1041857 IpmmNumNormOamInstallerProc.1041979 IpmmNumNormConfigSyncProc.1105066 ShRequestProcessNew.1126639 XdmsDiameterDistributor_PT.1101094 XdmsLinuxServer_PT.1101006
Domains	ims.mtas.if.sh ims.mtas.services.numnorm ims.mtas.services.numbertranslation ims.mtas.services.conf ims.mtas.services.conference ims.mtas.oam.numnorm ims.mtas.oam.xdms ims.mtas.subscriberdata.servicenumdata ims.mtas.xdms.* ims.mtas.xdmsval.validation ims.mtas.errlog.diam.sh



8 Example

The following is an example on how to start AppTrace:

Note: For the different processes and domains in the MTAS, see Section 5 on page 15 and Section 6 on page 17.

1. Log on to any of the payloads in the cluster (for example, PL-3):

```
ssh PL-3
```

2. Go to the AppTrace directory and start a new session:

```
cd /opt/lpmsv/bin/apptrace
```

```
./verify_domains.sh
```

Output example:

```
.....
ims.mtas.sip[abstract|OK]
ims.mtas.sip.dispatcher[concrete|OK]
ims.mtas.sip.distributor[concrete|OK]
ims.mtas.sip.sipstack[concrete|OK]
.....
SUMMARY: All domains verified OK
AppTrace verify_domains done
```

3. Begin the AppTrace session:

```
./begin_session.sh
```

Output example:

```
AppTrace begin_session done sessionId: 1264422676
```

4. Trace on all processors (alternatively, specify the processor to be traced):

```
./include_processors.sh -a
```

Output example:

```
Result from [.lpmsv.director.tutil]:AppTrace include_processors
done
```

5. Add process types:

```
add_process_type <PT1>[<PT2>...]
```

For example:



```
./add_process_type.sh ApplicationProcess.1060633
```

Output example:

```
AppTrace add_process_type done
```

6. Add domains:

For forloop tracing insert:

```
./insert_expression.sh 'ims.mtas.log.start(seek(string(<user@domain>), $FromID)) => $forlop=<forlopID>'
```

For example:

```
./insert_expression.sh 'ims.mtas.log.start(seek(string(7888005a), $FromID)) => $forlop=12345'
```

For all MTAS SIP domains, insert:

```
./insert_expression.sh ims.mtas.sip.*
```

Or with parameters, insert:

```
./insert_expression.sh 'ims.mtas.sip.* => L($id, $forlop, $Msg)'
```

Output example:

```
===Trace Program(simplified)===
```

```
[0]:ims.mtas.log.start
[1]:ims.mtas.log.dispatcher
[2]:ims.mtas.sip.distributor
[3]:ims.mtas.sip.sipstack
[4]:ims.mtas.sip.upstreamoc
AppTrace insert_expression done
```

For subscriber-specific domains, insert:

```
./insert_expression.sh 'ims.mtas.errlog.sip ($forlop != 0) => L($processorname, $pid, $date, $time, $id, $forlop, $FromID, $Response, $Reason, $SessSide)'
```

Output example:

```
===Trace Program(simplified)===
[0]:ims.mtas.log.start
[1]:ims.mtas.sip.dispatcher
[2]:ims.mtas.sip.distributor
[3]:ims.mtas.sip.sipstack
[4]:ims.mtas.sip.upstreamoc
[5]:ims.mtas.errlog.sip
```




```
AppTrace insert_expression done
```

To see all domains, enter:

```
./ls_domains.sh
```

To see all subdomains for a specific domain, enter the following (this is an example of the domain `ims.mtas.sip.*`):

```
./cat_domain.sh ims.mtas.sip
```

Output example:

```
=====
---Domain ims.mtas.sip---
---Condition: [abstract|OK]
---Signatures:
---Output prefix:
---Parameter Types:NONE
---Parameter Names:
NONE
---Parameter Postfixes:
NONE
---Subdomains:
Subdomain:ims.mtas.sip.dispatcher
Subdomain:ims.mtas.sip.distributor
Subdomain:ims.mtas.sip.sipstack
Subdomain:ims.mtas.sip.upstreamoc
---Errors:
NONE
```

7. Execute the AppTrace session:

```
./route_output.sh console
```

Output example:

```
AppTrace route_output command done
```

```
./display_session.sh
```

Output example:

```
---Session 1488566117 is OPEN for editing---general state:Clean
---Included processors:
PL-3
PL-4
---Enable process types:
ApplicationProcess.1060633/1060633 located on:
PL-3
PL-4
---Enable process instances:
```



NONE

---Trace program:

```
[0] ims.mtas.log.start(seek(string(7888005a),$FromID)) =>
$forlop=$forlop_set
Compiled: seek([_7888005a],$[0:0,0])=>F=$2
[1] ims.mtas.sip.dispatcher() => L
Compiled: 1=>L($[0:48,0], $[0:0,0])
[2] ims.mtas.sip.distributor() => L
Compiled: 1=>L($[0:48,0], $[0:0,0])
[3] ims.mtas.sip.sipstack() => L
Compiled: 1=>L($[0:48,0], $[0:0,0])
[4] ims.mtas.sip.upstreamoc() => L
Compiled: 1=>L($[0:48,0], $[0:0,0])
[5] ims.mtas.errlog.sip($forlop!=0) => L
($processorname,$pid,$date,$time,$id,$forlop,$FromID,$Response
,$Reason,$SessSide)
Compiled: $50!=0=>L
($[0:59,0],$54,$a61,$a60,$[0:48,0],$50,$[0:0,0],$1,$[0:2,0],$[0
:3,0])
---Routing of output:
To (buffered) console. Bytes/sec max:25000.
BuffSize/processor: 16384.
AppTrace display_session done
```

8. Upload the AppTrace session:

./upload_session.sh

Output example:

```
====Upload Session====
====Enabling Processes====
AppTrace upload_session done
```

9. Start tracing, for example, level DEBUG (55):

./start_trace.sh 56

Output example:

```
Trace level is 56 [trace events in ranges: debug, minor,
major, and exceptional]
====Starting Trace====
AppTrace start_trace done
```

Output of the trace shows in the console logs.

Output example:

```
2014-10-30 15:02:21.760 APP-TRACE: id:"ims.mtas.sip.dispatcher"
Msg:"2014-10-30 14:02:21.749 DEBUG
```



```
SipDispatcher.cc:773  
SC-1 167 ApplicationProcess 00| mKey = [sip:A-TC_TCP_LMCOIR00  
10@ericsson.com]"
```

10. Stop the AppTrace session:

```
./stop_trace.sh
```

```
./unload_session.sh
```

```
./end_session.sh
```

The Console logs are stored in the /cluster/storage/no-backup/cdclsv/log/lpmsv directory. For more information about console logs, refer to [MTAS Logs](#).