

# UE Trace Tool

Ericsson Service-Aware Policy Controller

User Guide

## **Copyright**

© Ericsson AB 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>UE Trace Tool Introduction</b>	<b>1</b>
<b>2</b>	<b>UE Trace Tool Function</b>	<b>2</b>
2.1	Subscriber Identities for Tracing	3
<b>3</b>	<b>Configuring the UE Trace Tool</b>	<b>4</b>
<b>4</b>	<b>UE Trace Procedures</b>	<b>5</b>
4.1	View the Trace Results	5
4.2	Start a Trace	5
4.3	List Activated and Scheduled Traces	7
4.4	See a Trace in Real-Time	8
4.5	Stop a Trace	9
<b>5</b>	<b>UE Trace Output Files</b>	<b>11</b>
5.1	Event Record Examples	11
5.1.1	Message Traced on the Gx Interface	12
5.1.2	Message Traced on the Rx Interface	12
<b>6</b>	<b>Troubleshooting</b>	<b>14</b>





# 1 UE Trace Tool Introduction

The purpose of this document is to provide a description of the UE Trace tool and how to use it.

## 2 UE Trace Tool Function

The UE Trace tool is used to troubleshoot problems affecting specific users in the network. It aims to reduce the time on identifying problems in current complex networks.

The UE Trace tool traces the signaling messages sent over the Gx and Rx interfaces based on the Diameter protocol. This tool enables the operator to trace a set of subscribers by their subscriber ID in IMSI, MSISDN, or SIP-URI format.

**Note:** SIP-URI is used for tracing over the Rx interface only.

The operator can define a trace session which consists of a unique identifier, one or several subscribers to be traced, and, optionally, the period of the session.

All UE Trace session data are stored in XML format according to 3GPP TS 32.423. These files contain records from all tracing sessions. After stopping the UE Trace session, PCAP files are generated automatically. XML and PCAP files can be downloaded by using the SFTP protocol.

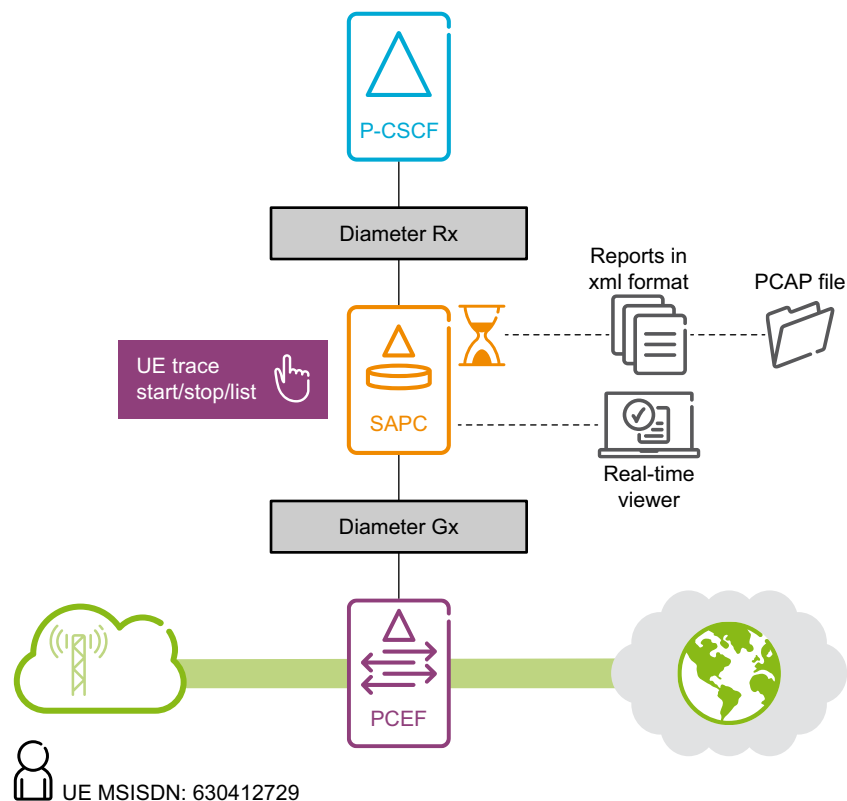


Figure 1 UE Trace Tool Overview



The UE Trace tool is enabled using a license. For detailed information on license management, refer to [License Management](#).

## 2.1 Subscriber Identities for Tracing

In the tracing process, the SAPC matches the subscriber identities being traced to the identities in the Subscription-Id AVPs received, rather than the administrative identities provisioned for the subscribers. The messages are traced if any subscriber identity is matched, even if the Diameter result is not successful.

If there is any identity match, at least the subscriber identities for filtering are shown in the trace output, but other traffic identities are not shown in the output. For example, if an operator filters the MSISDN or IMSI type only, while both the MSISDN and IMSI identities are received in the message, only the identity of the filtered type is shown in the XML output.

For tracing over the Rx interface, if no identity in the Subscription-Id AVPs from the Rx side matches the identity being traced or no Subscription-Id AVP is received from the Rx side, the SAPC takes a subscriber identity from the Gx session when the Gx session and Rx session are bound successfully. If the session binding fails in the situation, no trace output from the Rx side is generated.



## 3 Configuring the UE Trace Tool

For the UE Trace tool, it is possible to configure the report period. The SAPC provides the following parameter for the configuration:

- `reportPeriod`: This parameter specifies the time interval between storing two XML files on the network element.
  - Default value: 15 minutes
  - Possible values: `FIVE_MIN`, `FIFTEEN_MIN`, `THIRTY_MIN`, `ONE_HOUR`

To change the time interval between storing two XML files, set the `reportPeriod` attribute in the corresponding `UeTrace` MO class.





## 4 UE Trace Procedures

To perform UE Trace procedures, use Ericsson Command-Line Interface (ECLI) commands in the UeTrace class.

### 4.1 View the Trace Results

To view the trace results, follow the instructions below:

#### Steps

1. Start a trace.  
For detailed instructions, see [Start a Trace](#) on page 5.
2. List activated and scheduled traces.  
For detailed instructions, see [List Activated and Scheduled Traces](#) on page 7.
3. See a trace in real-time.  
For detailed instructions, see [See a Trace in Real-Time](#) on page 8.
4. Stop the trace.  
For detailed instructions, see [Stop a Trace](#) on page 9.

### 4.2 Start a Trace

---

---

#### Caution!

To avoid any performance impact, Ericsson does not recommend tracing more than 10 subscribers simultaneously.

---

---

To start or schedule a trace session, follow the instructions below:

1. Start an ECLI session.  
For detailed information on how to access the Common Operation and Maintenance (COM) CLI, also known as ECLI, for administration node operations, refer to [System Administrator Guide](#).
2. Navigate to the UeTrace class.

```
>dn ManagedElement=1,PolicyControlFunction=1,UeTrace=1
```



### 3. Execute the start command:

```
(ManagedElement=1,PolicyControlFunction=1,UeTrace=1)> start [- →
-traceSessionRef <sessionRef>] [--interfaces <interfaces>] [-- →
imsis <subscriberIds>] [--msisdns <subscriberIds>] [--sipuris →
<subscriberIds>] [--timeStart <YYYYMMDDTHHMMSS>] [--timeEnd <Y →
YYYYMMDDTHHMMSS>] [--duration <minutes>]
```

Where:

<b>traceSessionRef</b>	The unique identifier of a trace session. It is <b>mandatory</b> in this command line.  The trace session identifier can be used to stop a session.
<b>imsis/msisdns/sipuris</b>	The types of identities used for tracing subscribers: IMSI, MSISDN, or SIP-URI. At least one subscriber ID is <b>mandatory</b> . Multiple IDs are supported.  <b>Note:</b> If a SIP-URI format subscriber ID is given for filtering, only messages over the Rx interface are shown in the output.
<b>interfaces</b>	The interface over which the messages are traced. The allowed values are Gx and Rx. If no interface is specified, the SAPC uses the Gx and Rx interfaces together as the default value. That is, the messages over the Gx and Rx interfaces are traced.
<b>timeStart</b>	This is a <b>mandatory</b> parameter for scheduled tracing to specify the beginning of the scheduled UE Trace session.  Format: YYYYMMDDTHHMMSS
<b>timeEnd</b>	Either this or the duration parameter is <b>mandatory</b> for scheduled tracing. This parameter allows the operator to specify the end of the scheduled UE Trace session.  Format: YYYYMMDDTHHMMSS
<b>duration</b>	Either this or the timeEnd parameter is <b>mandatory</b> for scheduled tracing. This parameter allows the operator to specify the duration of the UE Trace session.  Format: integer value in minutes

[Example 1](#), [Example 2](#), [Example 3](#), and [Example 4](#) show examples of starting a UE Trace.

[Example 5](#) and [Example 6](#) show examples of scheduling a UE Trace.

**Example 1** Trace two subscribers by IMSI and two subscribers by MSISDN over the Gx interface

```
start --traceSessionRef 1234 --interfaces Gx --imsis "460020292059900,460020292059901" --msisdns "34 →
1591010201,341591010202"
```



### Example 2 Trace one subscriber by MSISDN

```
start --traceSessionRef tc159010101 --msisdns 3461591010101
New UE Trace Session started
```

### Example 3 Trace one subscriber by SIP-URI, specifying the end of the trace session

```
(UeTrace=1)>start --traceSessionRef 6321221 --sipuris sip:alice@atlanta.com --interfaces Rx --timeEnd 2017-12-28T16:30:00
New UE Trace Session started
```

### Example 4 Trace two subscribers by IMSI

```
start --traceSessionRef 123456 --imsis "460020292059900,460020292059901"
New UE Trace Session started
```

### Example 5 Schedule a trace for a subscriber by IMSI

```
(UeTrace=1)>start --traceSessionRef scheduledA --imsis 085954440505901410 --timeStart 2017-12-31T22:00:00 --timeEnd 2018-01-01T02:00:00
New UE Trace Session scheduled
```

### Example 6 Schedule a trace for a subscriber by MSISDN

```
(UeTrace=1)>start --traceSessionRef scheduledB --msisdns 612345670 --timeStart 2017-12-25T16:30:00 --duration 30
New UE Trace Session scheduled
```

- Note:**
- In a standalone scenario, when the SAPC node stops or is reloaded, the ongoing UE Trace session starts again. The scheduled trace sessions still take effect after the reload is finished.
  - In an active-standby geographical redundancy scenario, if the UE Trace is activated in the active zone and then the standby zone takes over, the UE Trace tool does not work automatically in the new active zone.
  - In an active-active geographical redundancy scenario, the UE Trace must be activated in both zones.

## 4.3 List Activated and Scheduled Traces

To check which sessions are being traced or scheduled to be traced, follow the instructions below:

1. Start an ECLI session.

For detailed information on how to access the COM CLI, also known as ECLI, for administration node operations, refer to *System Administrator Guide*.

2. Navigate to the UeTrace class.

```
>dn ManagedElement=1,PolicyControlFunction=1,UeTrace=1
```

3. Execute the list command:



```
(ManagedElement=1,PolicyControlFunction=1,UeTrace=1)> list
```

**Example 7** shows two scheduled and two normal UE Trace sessions, using IMSI, MSISDN, and SIP-URI IDs.

#### Example 7 Output example for scheduled sessions

traceSessionRef duration	userType	userId	interfaces	startTime	endTime	→
-----→						
scheduledA ---	IMSI	085954440505901410	Gx,Rx	2017-12-31T22:00:00	2018-01-01T02:00:00	→
-----→						
scheduledB 30	MSISDN	612345670	Gx,Rx	2017-12-25T16:30:00	---	→
-----→						
6321221 ---	SIP-URI	alice@atlanta.com	Rx	2017-12-22T07:47:18	2017-12-28T16:30:00	→
-----→						
session1 ---	MSISDN	654321091	Gx,Rx	2017-12-22T07:46:31	---	→
	MSISDN	677916500				→
	IMSI	7160652952503786				→
-----→						
-----→						

## 4.4 See a Trace in Real-Time

To see a trace in real-time, follow the instructions below:

1. Access the SAPC executing the following command:

```
> ssh sapcadmin@<OAM_VIP>
```

2. Execute the uetraceViewer command:

```
sapcadmin@SC-X> uetraceViewer --traceSessionRef <traceSessionRefId> | --help
```

Where:

**traceSessionRef** The unique identifier of a trace session. It is **mandatory** in this command line. Multiple traceSessionRef identifiers are not allowed.

In the UE Trace Viewer, you can see the subscriber who belongs to the tracing message together with the message type and the main AVPs presented in the messages.

**Note:** If the AVPs are not presented in the message, the SAPC prints a "-".

To stop the UE Trace Viewer, press Ctrl + C.



**Note:** If the UE Trace session is stopped, the UE Trace Viewer for that session is interrupted automatically.

**Example 8** shows the `uetraceviewer` command for the `tc159010116` trace session.

**Example 9** shows the output of a trace in the real-time viewer.

#### Example 8 UE Trace Viewer

```
uetraceviewer --traceSessionRef tc159010116
```

#### Example 9 Output example in real-time view

```
Mon Jan 22 11:23:42.268475 TraceSessionRef: "tc159010116", SubsId: "3461591010116,460020292059016",
Event: "CCR", SessionId: "tc_01_01_16_Session_Multiple_SubscriptionIds_UeTrace_Msisdn_Imsi;
ggsn2NodeHostname.nodeHostRealm.com;2;1297280", Protocol: "Gx", SubType: "MSISDN","IMSI", IpAddr:
"153.104.100.156", APN: "APN1", Event-Trigger: "-", RequestType: "Initial"

Mon Jan 22 11:23:42.283440 TraceSessionRef: "tc159010116", SubsId: "3461591010116,460020292059016",
Event: "CCA", SessionId: "tc_01_01_16_Session_Multiple_SubscriptionIds_UeTrace_Msisdn_Imsi;
ggsn2NodeHostname.nodeHostRealm.com;2;1297280", Protocol: "Gx", RequestType: "Initial", Result-Code: →
"2001"

Mon Jan 22 11:23:44.328481 TraceSessionRef: "tc159010116", SubsId: "3461591010116,460020292059016",
Event: "CCR", SessionId: "tc_01_01_16_Session_Multiple_SubscriptionIds_UeTrace_Msisdn_Imsi;
ggsn2NodeHostname.nodeHostRealm.com;2;1297280", Protocol: "Gx", SubType: "MSISDN","IMSI",
Event-Trigger: "IP_CAN_CHANGE", RequestType: "Update"

Mon Jan 22 11:23:44.330523 TraceSessionRef: "tc159010116", SubsId: "3461591010116,460020292059016",
Event: "CCA", SessionId: "tc_01_01_16_Session_Multiple_SubscriptionIds_UeTrace_Msisdn_Imsi;
ggsn2NodeHostname.nodeHostRealm.com;2;1297280", Protocol: "Gx", RequestType: "Update", Result-Code: →
"2001"
```

## 4.5 Stop a Trace

To stop the running sessions or cancel the scheduled sessions, follow the instructions below:

1. Start an ECLI session.

For detailed information on how to access the COM CLI, also known as ECLI, for administration node operations, refer to *System Administrator Guide*.

2. Navigate to the UeTrace class.

```
>dn ManagedElement=1,PolicyControlFunction=1,UeTrace=1
```

3. Execute the stop command:

```
(ManagedElement=1,PolicyControlFunction=1,UeTrace=1)> stop [-- →
traceSessionRef <sessionRef> | --all] [--nonPcapFile]
```

Where:

**all** Stops all the ongoing trace sessions.



<b>traceSessionRef</b>	The unique identifier of the trace session that was defined when activating a trace session.
<b>nonPcapFile</b>	After stopping the UE Trace session, the SAPC does not generate a PCAP file automatically.

**Note:** Either the `all` or the `traceSessionRef` parameter is mandatory.

[Example 10](#) shows the `stop` command for the `tc159010107a` trace session.

[Example 11](#) shows how to stop all the traces.

**Example 10** Stop a trace session whose ID is `tc159010107a`

```
stop --traceSessionRef tc159010107a
```

**Example 11** Stop all the traces

```
stop --all
```

After stopping a UE Trace session, the SAPC generates a PCAP file with all the incoming and outgoing messages for that UE Trace session. If the UE Trace session is stopped using the `all` parameter, the SAPC generates one PCAP file per open UE Trace session.

The PCAP files are available in the SAPC and fetched by an external system using the SFTP protocol.



## 5 UE Trace Output Files

The SAPC includes all filtered messages in an XML file. This file is generated periodically with the format compliant to 3GPP TS 32.423.

**Note:** The time period can be modified using the `reportPeriod` attribute.

If the operator modifies the time period, the SAPC uses the new value in the next period, not in the ongoing one.

**Note:** If the maximum file size of 10 MB is reached, more than one XML files are created within one time period.

The generated XML files contain records from all UE Trace sessions. The XML files are available in the SAPC and fetched periodically by an external system using the SFTP protocol.

The XML files are stored in the `/storage/no-backup/sapc/uetracefiles/uetraceXMLfiles` directory. This directory has a default preventive maintenance policy whose maximum number is 100 and maximum size is 250 MB. If the preventive maintenance limit is exceeded, the oldest file is deleted automatically.

To modify the default limits, change the `PolicyUeTraceXmlFiles` value of the `fileGroupPolicyId` parameter. For more information on file management, refer to [Handling Files](#).

After stopping a UE Trace session, the SAPC generates a PCAP file with all the incoming and outgoing messages for that UE Trace session. The PCAP files are available in the SAPC and fetched by an external system using the SFTP protocol.

**Note:** If the maximum file size of 10 MB is reached, more than one PCAP files are created per UE Trace session.

The PCAP files are stored in the `/storage/no-backup/sapc/uetracefiles/uetracePCAPfiles` directory. This directory has a default preventive maintenance policy whose maximum number is 100 and maximum size is 750 MB. If the preventive maintenance limit is exceeded, the oldest file is deleted automatically.

To modify the default limits, change the `PolicyUeTraceXmlFiles` value of the `fileGroupPolicyId` parameter. For more information on file management, refer to [Handling Files](#).

### 5.1 Event Record Examples

This section gives event record examples for traced messages.

For a description of the XML elements, see 3GPP TS 32.423.



### 5.1.1 Message Traced on the Gx Interface

[Example 12](#) shows a CCR-Initial message recorded for the traced UE.

#### Example 12 CCR-I Message Recorded for the Traced UE

```
<traceRecSession traceSessionRef="1234560A1B16" traceRecSessionRef="1">
  <ue idType="MSISDN" idValue="3461591010106"/>
  <msg function="Gx" name="Credit-Control-Request Initial" changeTime="162.754" vendorSpecific="false">
    <initiator type="PCEF"/>
    <target type="PCRF"/>
    <rawMsg protocol="Diameter" version="1">0100022c80000110010000160002fd0e0000fea900000107400000 →
627463
5f30315f30315f30365f53696e676c65537562736372696265725f4d756c7469706c6553657373696f6e3b6767736e324e6f →
646548
6f73746e616d652e6e6f6465486f73745265616c6d2e636f6d3b323b343835373731300000000012540000027736170634f →
776e48
6f737449642e6f70657261746f725265616c6d2e636f6d000000011b400000196f70657261746f725265616c6d2e636f6d00 →
000000
0001084000002b6767736e324e6f6465486f73746e616d652e6e6f6465486f73745265616c6d2e636f6d0000000128400000 →
196e6f
6465486f73745265616c6d2e636f6d000000000102400000c01000016000001164000000c000000010000019f400000c →
000000
00000001a0400000c000000100000008400000ce431c0a000003e8c0000010000028af0000000000001bb4000002c00 →
0001c2
4000000c00000000000001bc40000015333436313539313031303130360000000000040880000010000028af000003e80000 →
001e40
00000c41504e31000003f8c000003c000028af00000404c0000010000028af000000400000204c0000010000028af000196 →
400000
0203c0000010000028af0001964000000405c0000010000028af0000000100000406c0000010000028af0000000100000274 →
c00000
38000028af0000010a400000c000028af00000275c0000010000028af0000000100000276c0000010000028af0000000b</ →
rawMsg>
  </msg>
</traceRecSession>
```

### 5.1.2 Message Traced on the Rx Interface

[Example 13](#) shows an AA-Request message recorded for the traced UE.

#### Example 13 AAR Message Recorded for the Traced UE

```
<traceRecSession traceSessionRef="1234560A1B16" traceRecSessionRef="1">
  <ue idType="SIP-URI" idValue="22444032@phonesystem.3cx.com"/>
  <msg function="Rx" name="AA-Request" changeTime="812.958" vendorSpecific="false">
    <initiator type="AF"/>
    <target type="PCRF"/>
    <rawMsg protocol="Diameter" version="1">010003308000010901000014000000020000000200000107400000 →
5c7463
5f30325f53696e676c65537562736372696265725f4d534953444e5f494d53495f5349505552493b61664e6f6465486f7374 →
6e616d
652e6e6f6465486f73745265616c6d2e636f6d3b323b353239383031300000012540000027736170634f776e486f73744964 →
2e6f70
657261746f725265616c6d2e636f6d000000011b400000196f70657261746f725265616c6d2e636f6d00000000001084000 →
002861
664e6f6465486f73746e616d652e6e6f6465486f73745265616c6d2e636f6d00000128400000196e6f6465486f7374526561 →
6c6d2e
636f6d00000000000102400000c0100001400000116400000c0000000100000008400000c1edb754b000001bb40000038 →
000001
c2400000c00000002000001bc40000024323234343033324070686f6e6573797374656d2e3363782e636f6d000001e40 →
00000c
41504e31000001f8c0000013000028af756e6b6e6f776e0000000205c000014c000028af00000206c0000010000028af0000 →
000100
000207c00000bc000028af000001fdc0000010000028af00000001000001fbc000003f000028af7065726d697420696e2069 →
702066
726f6d2033302e3231392e3131372e373520323020746f203132372e312e312e353120323000000001fbc0000040000028af →
706572
6d6974206f75742069702066726f6d203132372e312e312e353220313020746f2033302e3231392e3131372e373520313000 →
0001ff
```





```
c0000010000028af0000000200000200c0000010000028af00000000000001f8c0000013000028af756e6b6e6f776e000000 →  
0208c0  
000010000028af0000000100000204c0000010000028af0003e80000000203c0000010000028af0003e800000001ffc00000 →  
100000  
28af0000000200000209c0000010000028af0003e8000000020ac0000010000028af0003e800000001f9c0000028000028af →  
41462d  
4368617267696e674964656e746966696572556e6b6e6f776e0000020bc0000010000028af000000000000274c000003800 →  
0028af  
0000010a4000000c000028af00000275c0000010000028af0000000100000276c0000010000028af00000013</rawMsg>  
</msg>  
</traceRecSession>
```



## 6 Troubleshooting

For typical errors and possible solutions see [Table 1](#).

Table 1 Problems and Solutions

Problem	Possible Cause	Possible Solution
ERROR: The traceSessionRef is already in use.	This error is displayed when the operator starts a new UE Trace session and there is another ongoing UE Trace session with the same traceSessionRef ID.	Define another ID for the trace session.
ERROR: The UE Trace session <traceSessionRef> does not exist.	This error is displayed when the operator tries to stop an unavailable UE Trace session, for example by indicating a wrong traceSessionRef ID.	Check the traceSessionRef ID and correct it if necessary.
ERROR: The UE Trace session <traceSessionRef> is stopped. No messages for this UE Trace session.	This error is displayed when the operator stops a UE Trace session, but the SAPC does not collect any messages for the interrupted session. In this case, the SAPC does not generate a PCAP file.	Start a new UE Trace session and check if there are any records in the XML file before stopping the UE Trace session.
The SAPC does not store the filtered message in the uetraceXMLfiles directory.	The license is not granted.	Verify if the license is granted.