MBMS for MME (eMBMS)

Released as Deploy Quality in Release 20.0.

This chapter deals with the implementation of the LTE version of Multimedia Broadcast/Multicast Service (eMBMS) on the Cisco Mobility Management Entity (MME).

- Feature Description, on page 1
- How It Works, on page 4
- Configuring MME-eMBMS Service, on page 14
- Managing/Troubleshooting the eMBMS on the MME, on page 15

Feature Description

Multimedia Broadcast/Multicast Service (MBMS) is available on a number of network elements and is variously and well described on the Internet. Before looking at the implementation of eMBMS on the Cisco MME, we start with a quick overview of the 3GPP standard concepts to confirm the MME's position.

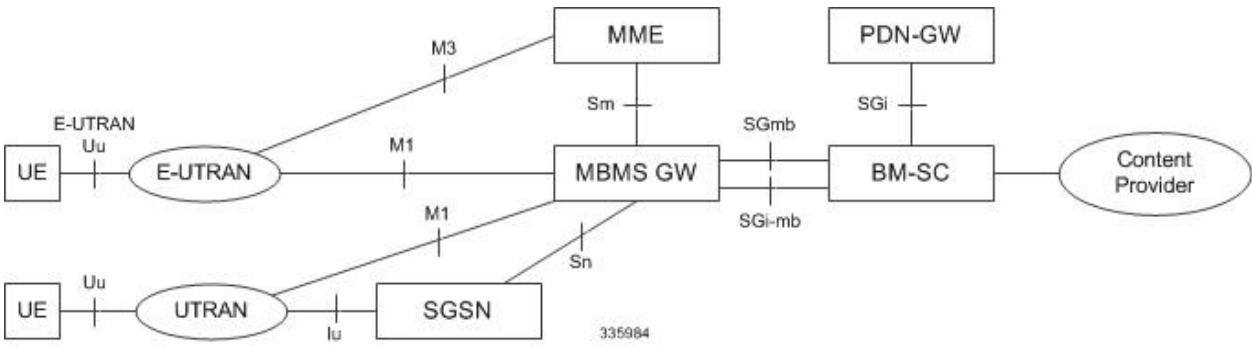
Overview per 3GPP TS 23.246

As defined by 3GPP TS 23.246:

MBMS is a point-to-multipoint service in which data is transmitted from a single source entity to multiple recipients. Transmitting the same data to multiple recipients allows network resources to be shared.

The MBMS bearer service offers two modes: Broadcast & Multicast mode. Broadcast Mode is supported for EPS and GPRS and Multicast Mode is supported for GPRS.

Figure 1: eMBMS Network Diagram



Use Cases for eMBMS on the MME

Transmitting one set of data to many, many eMBMS-capable end-users has a range of possible operator use cases:

- Mobile TV
- Digital Radio
- Video Kiosk or Video on Demand
- Connected Car
- Fixed LTE Quadruple Play
- Local Information such as Coupons
- Wireless Emergency Alerts
- Stadium App
- Data Feeds & Notifications
- e-Newspapers and e-Magazines
- Firmware/OS Updates
- Pushed Video Ads
- Last Mile CDN
- Internet of Things (Smart Meters)

MME Support for MBMS

In an LTE network, the operator using a Cisco MME can provide an MBMS data service using the e-MBMS solution proposed in 3GPP TS 23.246. eMBMS in the LTE network involves the following nodes and reference points:

- Broadcast Multicast Service Centre (BM-SC) - Supports various MBMS user-service specific services such as provisioning and delivery. The BM-SC sets up the e-MBMS session, initiates delivery of the content by pulling it from the content server, uses appropriate CODEC on the content, and collects the reception receipt from the UEs for certain kinds of content.
- MBMS-GW - Creates the MBMS bearer and receives the user-plane MBMS traffic from the BM-SC. Once received, the MBMS-GW allocates a multicast transport address and performs the GTP-U encapsulation of the MBMS data.
- **MME** - Running the Cisco MME-eMBMS service on the MME, the MME communicates with the MBMS GW and the MCE using Sm and M3 interfaces, respectively, for all eMBMS communications and functions. MME-eMBMS facilitates sessions scheduled by the BM-SC. The MME-eMBMS service identifies service areas to be served by a particular MBMS session, so that the MME handles session start, update, and stop. The MME also handles setup and configuration requests from the MCEs.
- E-UTRAN (eNodeB/MCE) - Handles session setup and broadcasting of MBMS data on the broadcast channel on the air. The Multicell/Multicast Coordination Entity (MCE) manages the MBMS content and resources.

- M1 - Is the reference point between MBMS GW and E-UTRAN/UTRAN for MBMS data delivery. IP Multicast is used on this interface to forward data.
- M3 - Is the reference point for the control plane between MME and E-UTRAN.
- Sm - Is the reference point for the control plane between MME and MBMS-GW.
- Sn - Is the reference point between MBMS GW and SGSN (S4 based) for the control plane and for MBMS data delivery. Point-to-point mode is used on this interface to forward data.
- SGi-mb - Is the reference point between BM-SC and MBMS-GW function for MBMS data delivery.
- SGmb - Is the reference point for the control plane between BM-SC and MBMS-GW.

With MBMS functionality, the MME now supports additional interfaces :

- the Sm interface, between the MME and the MBMS-GW, receives MBMS service control messages and the IP Multicast address for MBMS data reception from the MBMS-GW. It also carries the EPS GTPv2-C messages:
  - MBMS Session Start messages
  - MBMS Session Update messages
  - MBMS Session Stop messages
- the M3 interface provides the reference point for the control plane between the MME and the MCE (E-UTRAN). The M3 Application Protocol (M3AP) supports the functions of the M3 interface by providing:
  - Support for both IPV4 and IPV6 addresses at MME endpoint.
  - Session Management - This overall functionality is responsible for starting, updating, and stopping MBMS sessions via the session control signaling on the SAE bearer level.
  - M3 Setup functionality for initial M3 interface setup for providing configuration information.
  - Reset functionality to ensure a well-defined re-initialization on the M3 interface.
  - Error Indication functionality to allow a proper error reporting.
  - MCE Configuration Update function to update the application level configuration data needed for the MCE.

Relationships

The MME-eMBMS service is not associated with the MME service or any of the other major services available on the MME, such as the SBC, SLS, or SGS services.

License Information

A valid license key for the M3 and Sm interfaces is required to enable the controlling CLI and functionality of eMBMS on the MME. Contact your Cisco Account or Support representative for information on how to obtain a license.

How It Works

MBMS Broadcast Service - the Basic Phases

Pre-requisites - the UE, MME, and eNodeB must all be eMBMS capable.

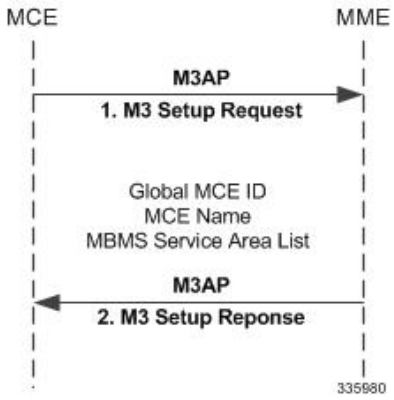
The basic phases of the MBMS broadcast service are:

1. Service Announcement - Informs UEs with media descriptions specifying the media to be delivered as part of an MBMS user service. An MBMS user service announcement can use any one of many mechanisms, for example: SMS Cell broadcast, PUSH mechanism like WAP, MMS, HTTP.
2. Session Start - The BM-SC is ready to send data. Session Start is the trigger for bearer resource establishment for MBMS data transfer.
3. MBMS Notification - Informs the UEs about forthcoming/ongoing MBMS data transfer.
4. Data Transfer - Data transferred to the UE.
5. Session Stop - BM-SC determines that there will be no more data. All the bearer resources are released at session stop.

M3 Setup Procedure

M3 Setup procedure exchanges application level data needed for the MCE and MME to correctly interoperate on the M3 interface.

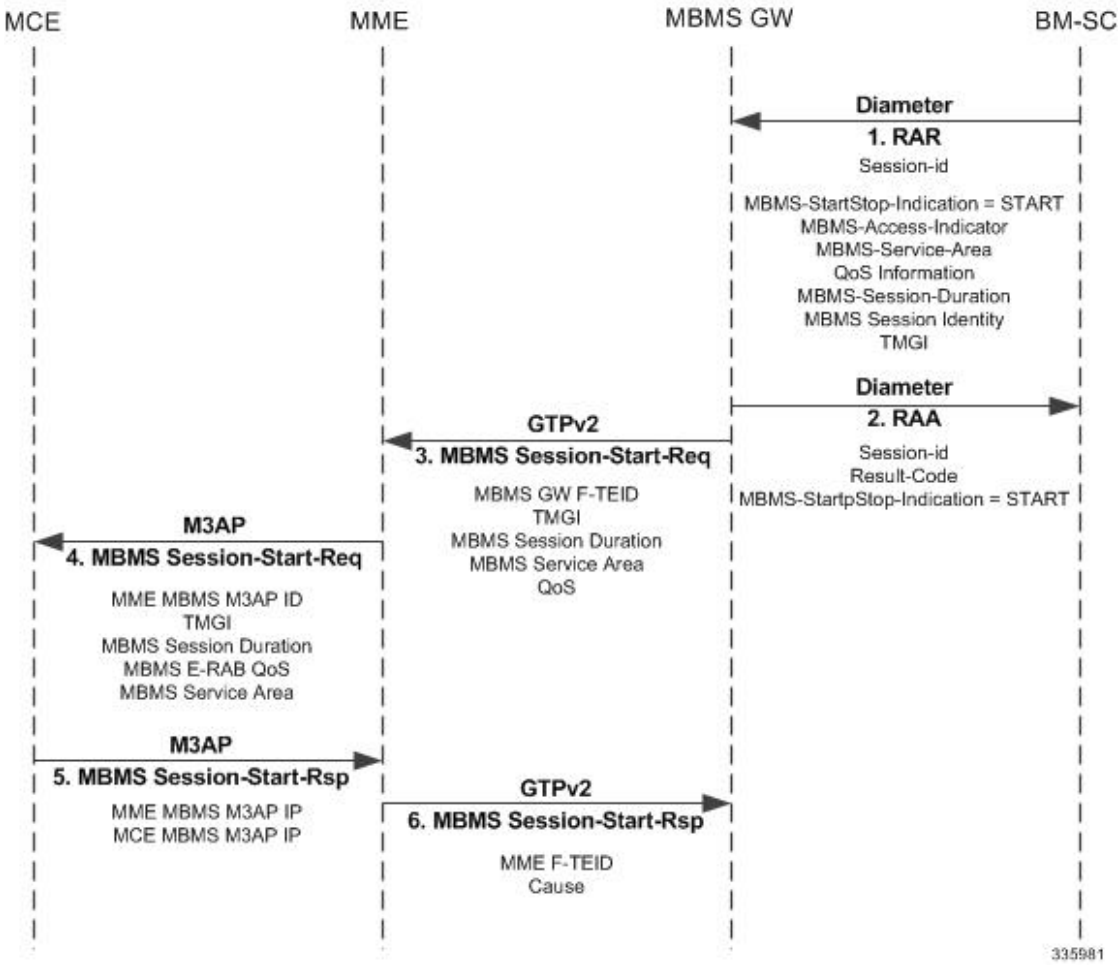
Figure 2: M3 Setup Procedure



1. The MCE sends an M3 Setup Request containing the Global MCE ID, MCE Name & Service Area List.
2. The MME Responds with an M3 Setup Response.

Session Start Procedure

Figure 3: Session Start Procedure



1. The BM-SC sends an RAR (Diameter Re-Authorization Request) message to indicate start of the transmission and to provide the session attributes to the MBMS GWs. The session attributes sent includes but is not limited to: Temporary Mobile Group Identity (TMGI), Flow Identifier, quality of service (QoS), MBMS Service Area, Session Identifier, Estimated Session Duration, List of MBMS control plane nodes (MMEs, SGSNs) for MBMS-GW, access indicator.
2. The MBMS-GW creates an MBMS bearer context, stores the session attributes in the MBMS bearer context, and sends an RAA (Diameter Re-Authorization Response) message to the BM-SC.
3. The MBMS-GW sends a Session Start Request message including the session attributes to MMEs (identified from the "List of MBMS control plane nodes" attribute).
4. The MME creates an MBMS bearer context and initiates an MBMS SESSION START REQUEST message to the MCE. This also sets up the MBMS service-associated logical M3 connection with the MCE.
5. The MCE creates an MBMS bearer context, stores the session attributes and sets the state attribute of its MBMS bearer context to 'Active'. The MCE reports the result of the requested MBMS E-RAB in the MBMS SESSION START RESPONSE message.

6. The MME responds with MBMS Session Start Response to the MBMS-GW as soon as the session request is accepted by one E-UTRAN node.

In some cases, the session start procedure can involve multiple MCEs. The following briefly outlines the procedures for three possible scenarios:

Scenario 1: Some MCEs (remember, that a single Start Request can go to multiple MCEs in parallel) return failure for the Start Request:

1. The MBMS-GW sends an MBMS Session Start Request to the MME (perhaps in an MBMS Service Area served by multiple MCEs).
2. The MME sends MBMS Session Start Request to multiple MCEs simultaneously.

   Some MCEs respond with MBMS Session Start Failure.
3. An MCE sends the MME an MBMS Session Start Response indicating a successful outcome.
4. The MME responds with cause "Request Accepted".

Scenario 2: All MCEs return failure for the Start Request:

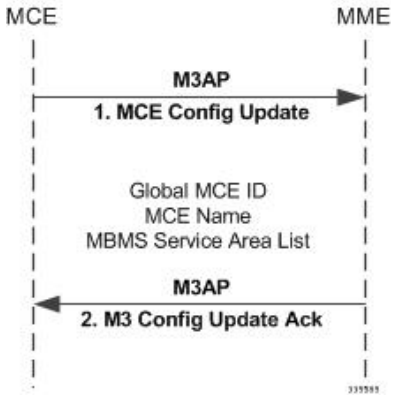
1. The MBMS-GW sends an MBMS Session Start Request to the MME (perhaps in an MBMS Service Area served by multiple MCEs).
2. The MME sends MBMS Session Start Request to the MCEs
3. All MCEs respond with MBMS Session Start Failure.
4. The MME responds with failure cause "Invalid Peer".

Scenario 3: Delayed success response from some MCEs:

1. The MBMS-GW sends an MBMS Session Start Request to the MME (perhaps in an MBMS Service Area served by multiple MCEs).
2. The MME sends MBMS Session Start Request to the MCEs
3. An MCE sends the MME an MBMS Session Start Response indicating a successful outcome.
4. The MME responds with cause "Request Accepted".
5. Further Start Session Responses will be ignored and they will not have any effect on the MBMS bearer context state.

MCE Configuration Update Procedure

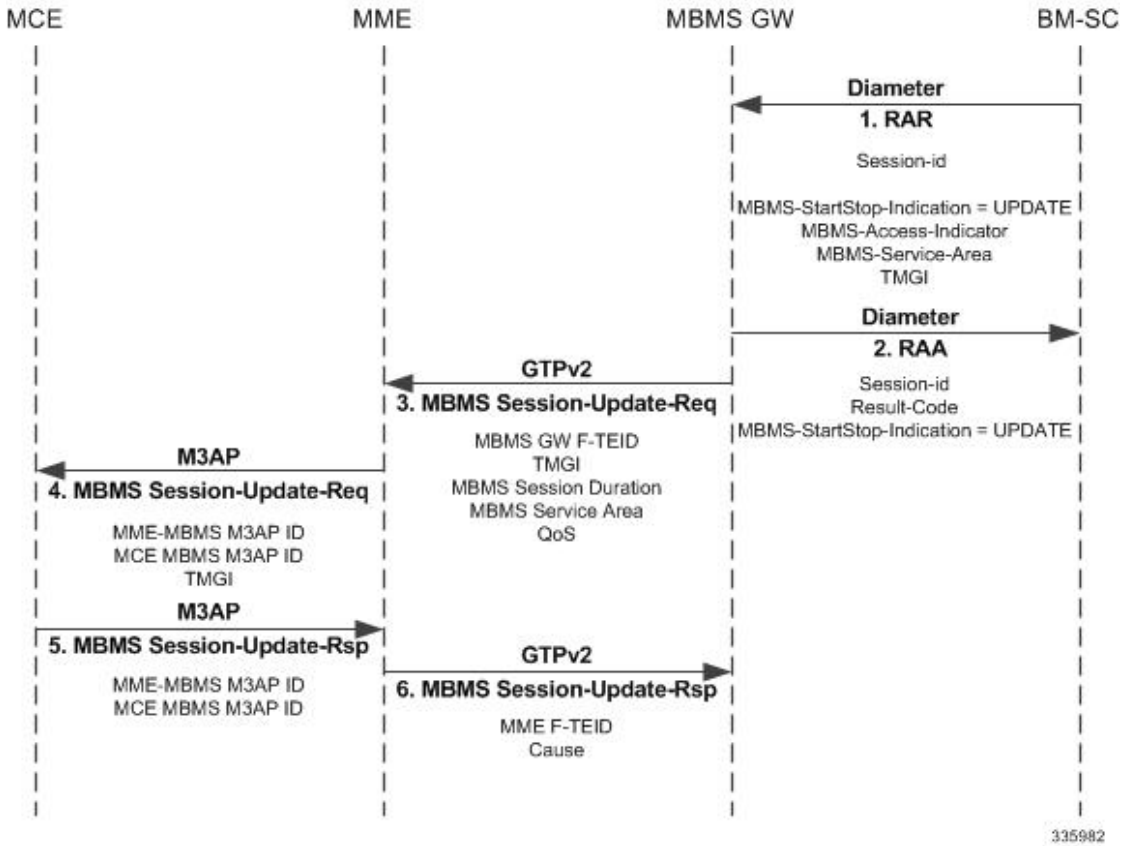
Figure 4: MCE Configuration Update



1. MCE Sends a MCE Configuration Update containing the Global MCE ID, MCE Name & Service Area List.
2. MME Responds with MCE Configuration Acknowledge.

Session Update Procedure

Figure 5: Session update Procedure

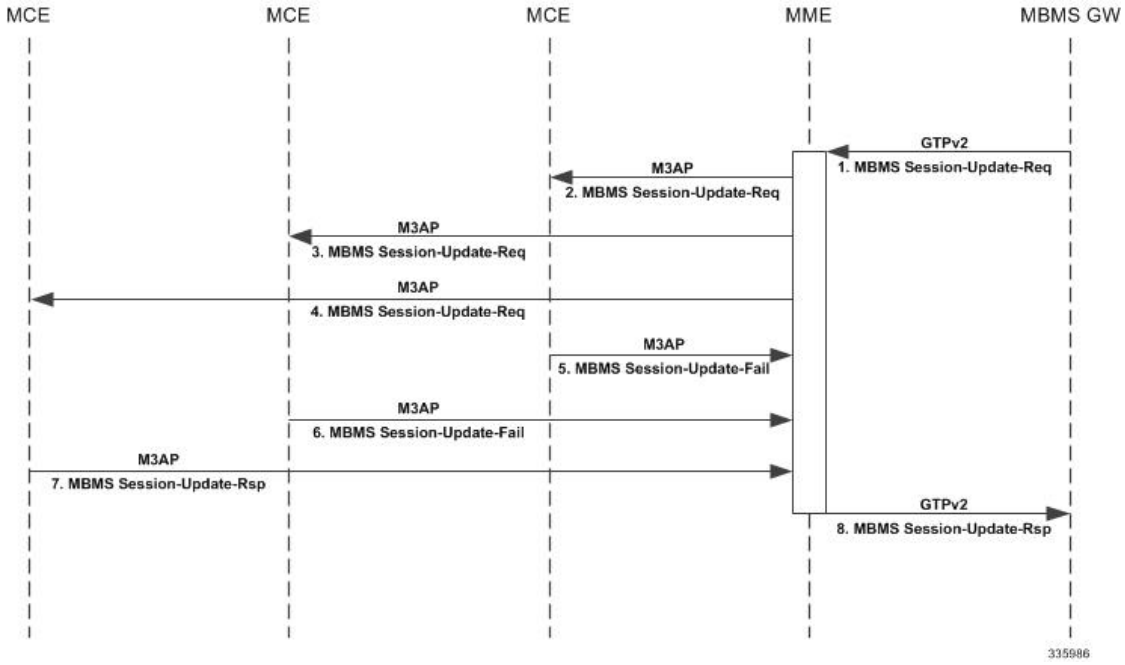


1. The BM-SC sends a RAR message to indicate that the MBMS session is updated. The attributes that can be modified by the RAR message are the MBMS Service Area, the Access indicator and the list of MBMS control plane nodes.
2. The MBMS-GW responds with a RAA message to the BM-SC.
3. The MBMS-GW initiates session start or session update procedure towards the MMEs in its list of MBMS control plane nodes.
4. The MME informs the MCEs, about changed characteristics of an ongoing MBMS service session, based on the MBMS Session Update Request. The MME sends MBMS Session Update Request to all MCEs which have earlier received an MBMS Session Start Request with the same TMGI and GLOW ID.
5. The MCE responds to the MME to confirm the reception of the Session Update Request message.
6. The MME returns a response to the MBMS-GW as soon as the Session Update Request is accepted by any E-UTRAN node.

In some cases, the session update procedure can involve multiple MCEs. The following briefly outlines the procedures for three possible scenarios:

Scenario 1: Some MCEs return failure for the Update Request:

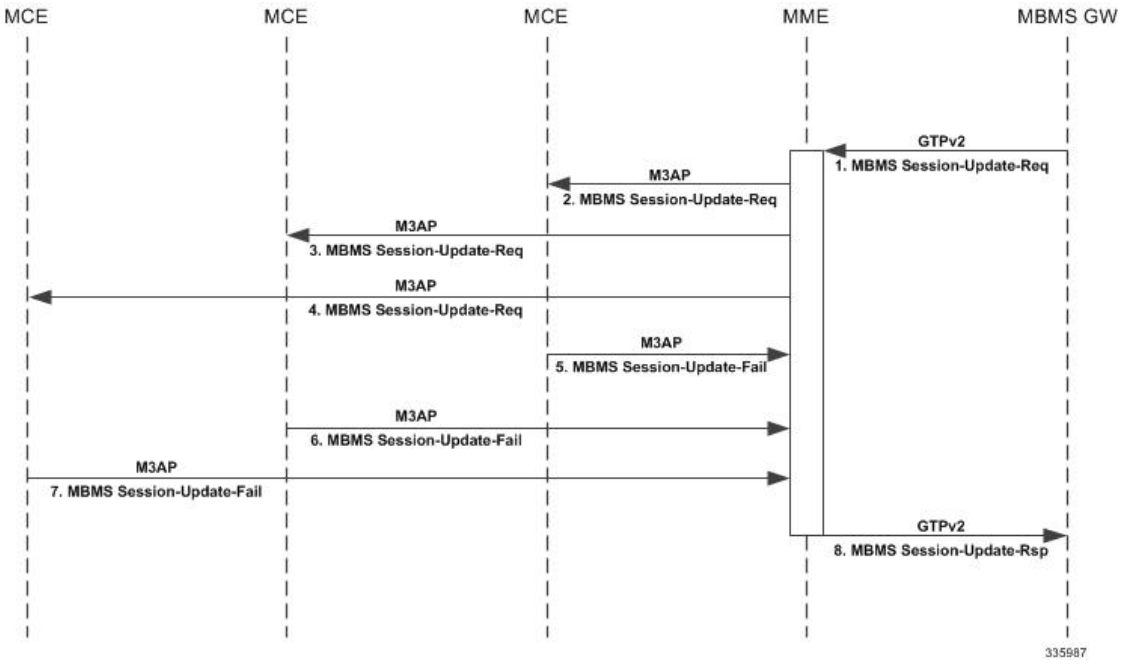
Figure 6: Update Failure from an MCE



1. The MBMS-GW sends an MBMS Session Update Request to the MME.
2. The MME sends an MBMS Session Update Request to MCEs.
3. Some MCEs respond with MBMS Session Update Failure.
4. The MCE sends an MBMS Session Update Response indicating a successful outcome.
5. The MME responds with cause "Request Accepted"

Scenario 2: All MCEs return failure for the Update Request:

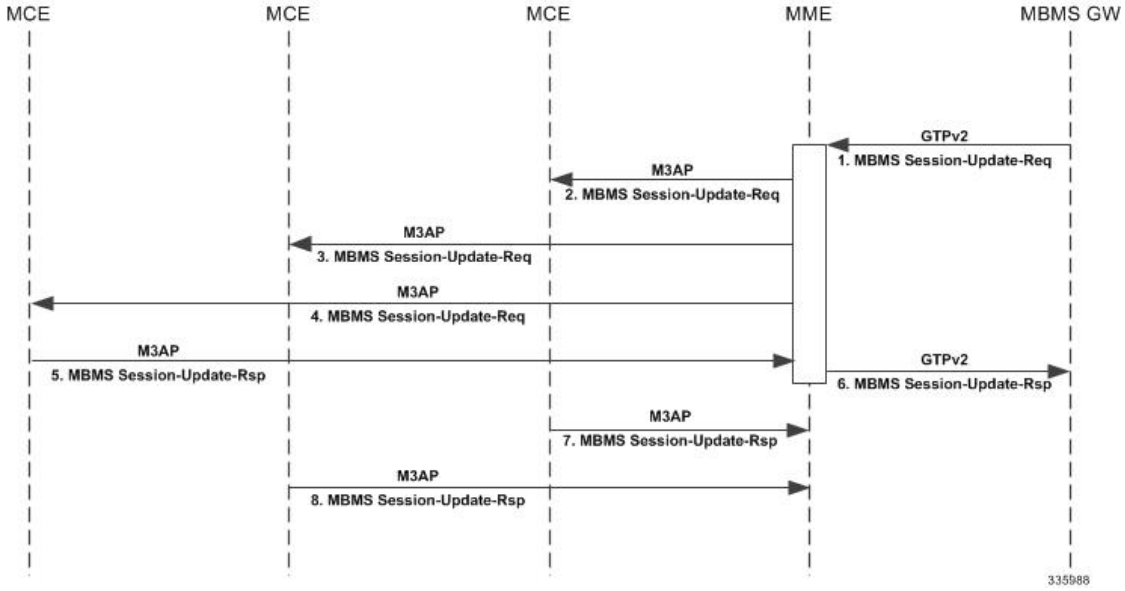
Figure 7: Update Failure from All MCEs



1. The MBMS-GW sends an MBMS Session Update Request to the MME.
2. The MME sends an MBMS Session Update Request to MCEs.
3. All MCEs respond with MBMS Session Update Failure.
4. The MME responds by sending Session Update Response with cause EGTP_CAUSE_INVALID_REPLY_FROM_REMOTE_PEER to the MBMS GW.

Scenario 3: Delayed success responses from all MCEs:

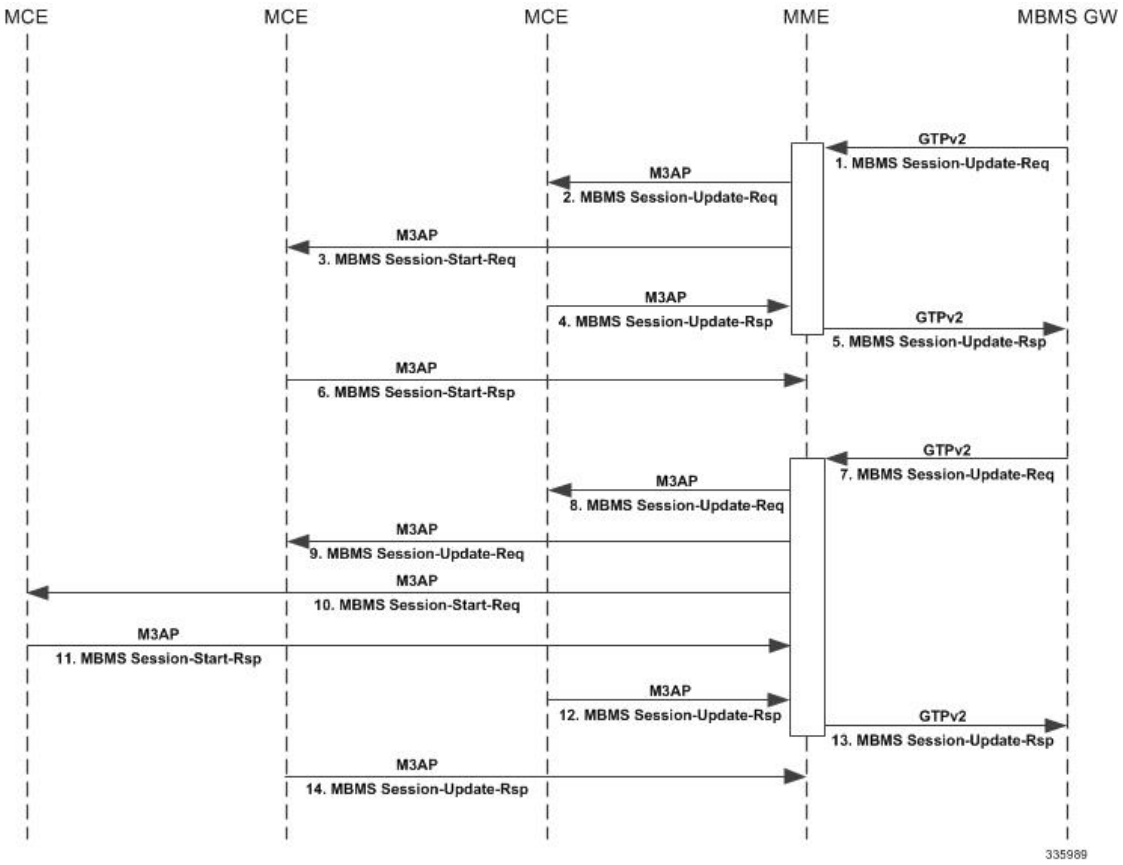
Figure 8: Delayed Update Responses



1. The MBMS-GW sends an MBMS Session Update Request to the MME.
2. The MME sends an MBMS Session Update Request to MCEs.
3. An MCE sends an MBMS Update Response indicating a successful outcome.
4. The MME responds with cause "Request Accepted"
5. Further responses are ignored and will have no effect on the MBMS bearer context state.

Scenario 4: Session update involved the additional / deletion of MBMS service areas:

Figure 9: Session Update Changing MBMS Service Areas

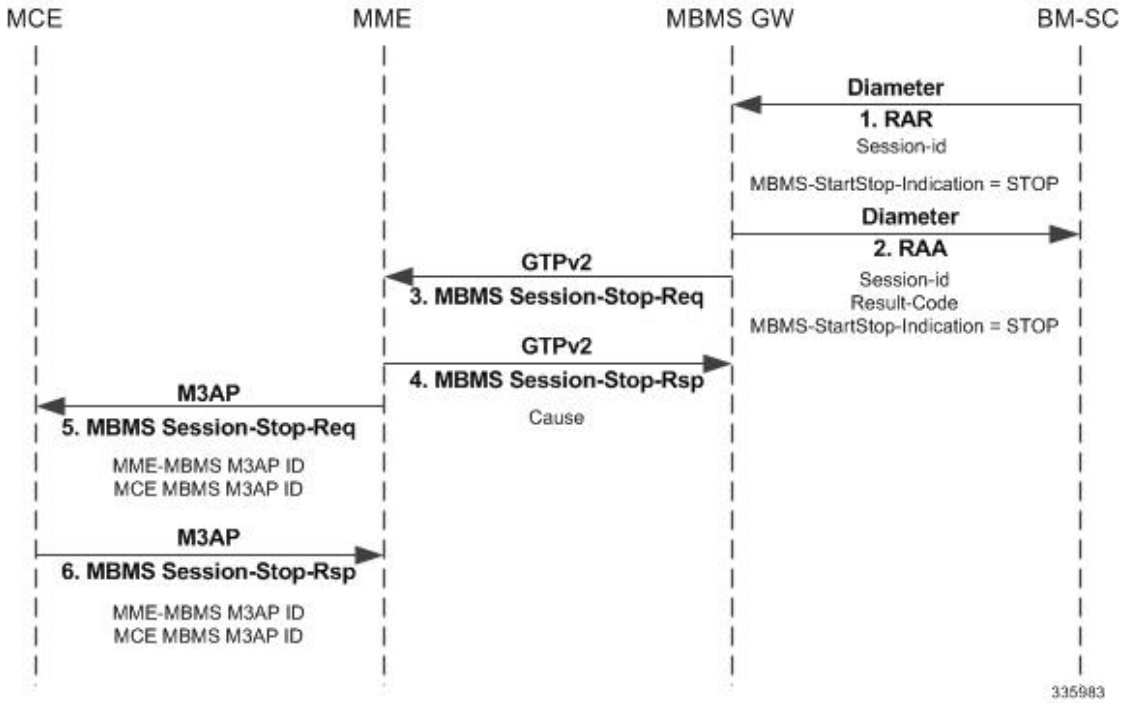


1. The MBMS-GW sends an MBMS Session Start Request to an MME. (For this scenario, consider that the Start Request has been sent to multiple MCEs, in this case MCE1 and MCE2.)
2. The MBMS Session Start Request is sent to MCE1.
3. The MBMS Session Start Request is sent to MCE2.
4. MCE1 responds with successful outcome.
5. The MME responds with cause "Request Accepted" without waiting for response from all MCEs.
6. MCE2 responds with successful outcome.
7. For an existing MBMS bearer context, Update Request is sent from MBMS-GW. (Let us consider there is an MBMS service area deleted and a new service area added.)
8. MME sends MBMS Session Update Request to MCE1. MCE1 has already processed MBMS Session Start Request.
9. MBMS Service Area in MCE2 is deleted in the Session Update Request. Despite this, the MME sends a Session Update Request to MCE2 with service areas received in MBMS Session Update Request from the MBMS GW over GTPv2.
10. For a new Service Area present in the Update Request, the MME sends a Session Start Request to MCE3.

11. The MCE is expected to create an MBMS bearer context and set its state attribute to "Active" and to confirm the Session Start Request.
12. As soon as the MME receives a response with successful outcome, the MME responds with cause "Request Accepted" without waiting for responses from all MCEs.
13. The MME receives a Session Update Response, indicating successful outcome, from the MCE for the Session Update Request which was sent earlier.
14. The MME sends MBMS Session Update Response with cause "Request Accepted" to the MBMS GW.
15. Responses for Update and Start Requests are sent to the MME from other MCEs.

Session Stop Procedure

Figure 10: Session Stop Procedure



1. The BM-SC sends an RAR message to indicate that the MBMS session is terminated and the bearer plane resources can be released.
2. The MBMS-GW responds with Session Stop Response and releases its information regarding the session.
3. The MBMS-GW forwards Session Stop Request message to the MME.
4. The MME releases the MBMS bearer context and responds with MBMS Session Stop Response.
5. The MME initiates MBMS Session Stop Request message to the MCE.
6. The MCE releases the MBMS bearer context associated with the logical M3 connection and responds with MBMS Session Stop Response.

Architecture - MME-eMBMS Service

A new service (mme-embms-service) supports MME's eMBMS functionality. This service is not coupled with the existing mme-service. The maximum number of MME-eMBMS services that can be created is 8. For details about the command in the configuration mode, refer to the *Configuring eMBMS* section in this document.

MCEs can be deployed with eNodeB(s) or they can be standalone. Depending on the deployment model, the number of MCEs supported can vary. Currently, the MME (system) support is limited to 300 MCEs and 100 MBMS sessions. There is no separate limit enforced on the number of MCEs per mme-embms service.

Important The MME supports a maximum total combination of eight (8) MME-specific services, of the types MME + eMBMS + SGs+ SBc + SLs -service, be configured per chassis.

Supported Features and Functions

- Sessions are identified by the combination of the TMGI and the MBMS Flow ID. In the case of no Flow ID, TMGI alone can be used to identify sessions and MBMS Flow ID would be assumed to be 0.
- Session Controller Recovery is provided to fetch MME eMBMS service configuration from the Session Manager in case of session controller failure.
- Manager Recovery support for: MMEdemux, MMEmgr, SessMgr, AAAmgr, EgtpegMgr.
- SMC switchover, PSC card migration, and slot hiding.

Standards Compliance

The Cisco implementation of eMBMS on the Cisco MME is compliant with the following standards:

- 3GPP TS 23.246, Version 12.6.0 - Multimedia Broadcast/Multicast Service (MBMS); Architecture and functional description
- TS 36.444, Version 12.2.0 - M3 Application Protocol
- TS 29.274, Version 12.8.0 - Tunnelling Protocol for Control plane (GTPv2-C)

Limitations

- MBMS flags are supported only for MBMS Session Start Request messages and not for MBMS Session Stop Request messages.
  - Re-establishment IE, which comes from MBMS-GW in Session Start Request, is forwarded to the MCEs.
  - MBMS flags are not supported in MBMS Session Stop Request messages.
- Currently, CLI limitations for the MME eMBMS feature include:
  - the **monitor protocol** command is supported, but without any of the command keywords.
  - the **monitor subscriber** command is not supported at this time for use with eMBMS.

- In the event that all MMEmgrs are restarted at the same time, then MCE Restart Handling will not perform properly.
- If the Session-Start-Response message includes an Absolute Time timestamp value (for the MBMS Data Transfer) that corresponds to a time in the past, then Session Start is rejected with cause "Mandatory IE Incorrect".

Configuring MME-eMBMS Service

Reminder: A valid M3/Sm interface license key is required to use the following commands to create an MME-eMBMS service.

The following configuration commands will setup a single MME-eMBMS Service. The commands in the MME-eMBMS service configuration mode are listed in the order in which they appear. The commands can be entered in a different order, to suit your needs.

```
configure
  context ctxt_name
    mme-embms-service mme_embms_service_name
      associate egtp-service egtp_service_name [ context ctxt_name ]
      associate sctp-param-template sctp_param_template_name
      bind { ipv4-address ipv4_address | ipv6-address ipv6_address  }
      mmemgr-recovery { no-reset | reset-peers }
      plmn-id mcc mcc mnc mnc
      sctp port port_number
      setup-timeout number_seconds
```

Notes:

- The *ctxt_name* identifies the context in which the MME-eMBMS service configuration is to reside. The name must be a string of 1 through 79 alphanumeric characters.
- The *mme_embms_service_name* must be a string of 1 through 63 alphanumeric characters. We recommend that this service name be unique on the chassis. For additional information, refer to the **mme-embms-service** command description in the *Global Configuration Mode Commands* section of the *Command Line Interface Reference*.
- The **associate** command associates either a previously configured eGTP service with the MME-eMBMS service or a previously configured SCTP parameter template. The command should be repeated to associate both with the MME-eMBMS service.
  - **egtp-service** *egtp_service_name* must be a string of 1 through 63 alphanumeric characters.
  - **context** *ctxt_name* in which the eGTP service has been configured; the context name must be a string of 1 through 79 alphanumeric characters.
  - **sctp-param-template** *sctp_param_template_name* must be a string of 1 through 63 alphanumeric characters.
  - For additional information about the eGTP service or SCTP parameter template configurations, refer to the *Command Line Interface Reference*.

- The **bind** command binds the MME-eMBMS service to a logical IP interface serving as the M3 interface. Enter either a standard IPv4 or IPv6 address.
- The **mmemgr-recovery** command sets the action the MME is to take regarding the peers (MCEs) upon recovery after an MME Manager crash/failure:
  - **no-reset** - so peer associations are not reset.

    reset-peers - so peer associations are reset. *NOTE: Currently, this option is not supported.*
- The **plmn-id** command configures the PLMN identifier associated with the eMBMS service area.
- The **sctp** command configures the SCTP port number to be associated with the M3AP interface of the eMBMS service. The *port_number* is an integer from 1 to 65535 and the default is 36412.
- The **setup-timeout** command configures the number of seconds for the guard timer expiry for call setup. The *timeout_value* is an integer from 1 to 10000 and the default is 60.

Important The maximum number of MME-eMBMS services that can be created on a single chassis is 8. However, you need to note that Of the 256 possible services, the MME supports a maximum total combination of eight (8) MME-specific services, of the types MME + MME-eMBMS + SBc + SGs + SLs -service, be configured per chassis.

Verifying the MME-eMBMS Feature Configuration

Use the following command to verify your configuration:

show mme-embms-service [all | name *mme_embms_service_name* **]**

The output will provide a display similar to the following:

```
show mme-embms-service name embms1

 Service name                    : embms1
 Context                         : ingress
 Status                          : STARTED
 SCTP Bind Port                  : 36444
 MME-EMBMS IP Address            : 192.80.80.201
                                   192.80.80.202
 SCTP Param Template Associated  : sctptemp1
 Setup Timeout                   : 60
 PLMN                            : mcc 123 mnc 456
 EGTPC Service                   : egtp_mbms
```

Managing/Troubleshooting the eMBMS on the MME

Managing the eMBMS Service

The following commands can be used to manage an active eMBMS service. They are issued from the Exec mode.

- To reset MCE associations on the M3AP link by sending a RESET message to a designated MCE/eNodeB to reset all UE-associated M3 connections.

mme-embms reset m3-peer *peer_id*

- To disconnect MCE associations on the M3AP link and perform a graceful/ungraceful disconnection of an SCTP peer (MCE) , use the following command in the Exec mode:

  mme-embms disconnect m3-peer *peer_id*

Output from "show" Commands

Numerous counters and information fields provide information helpful for monitoring and/or troubleshooting eMBMS on the MME. The following is a listing of the commands with brief information on their usefulness:

show mme-embms-service { all | { all-session-info [summary] } | { m3ap statistics { all [verbose] | name *mme_embms_service_name* **} } | { mce-association { all [summary] | full { all | name** *mme_embms_service_name* **} | name** *mme_embms_service_name* **[summary] | path-info { all | name** *mme_embms_service_name* **} } } | { mce-session-association { plmn-id mcc** *mcc* **mnc** *mnc* **mce-id** *mce_id* **| tmgi-service-id** *tmgi_service_id* **mbms-flow-id** *mbms_flow_id* **} } | name** *mme_embms_service_name* **| sctp statistics { all | name** *mme_embms_service_name* **} } }**

Notes:

- **all** -- a listing of the names of all created MME-eMBMS services and a display of the overall MBMS service status.
- **all-session-info [summary]** -- a listing of the eMBMS sessions being handled by the MMEmgr or optionally a summary of eMBMS session information.
- **m3ap statistics { all [verbose] | name** *mme_embms_service_name* **} }** -- a display of all M3AP statistics available for the MME or a display of the M3AP statistics for the named "active" MME-eMBMS service.
- **mce-association { all [summary] | full { all | name** *mme_embms_service_name* **} | name** *mme_embms_service_name* **[summary] | path-info { all | name** *mme_embms_service_name* **} }** – displays
  - all MCE peer associations for all or named MME-eMBMS service(s)
  - identifies the number of MCE associations with all or the named MME-eMBMS service(s)
  - displays path information for MCEs associated with all or the named MME-eMBMS service(s); particularly useful for checking multi-homed sessions.
- **mce-session-association { plmn-id mcc** *mcc* **mnc** *mnc* **mce-id** *mce_id* **| tmgi-service-id** *tmgi_service_id* **mbms-flow-id** *mbms_flow_id* **}** – displays
  - MCE session associations for a specific MCE
  - MCE session associations for the TMGI or TMGI and FLOW ID combination.
- **name** *mme_embms_service_name* **[summary]** – displays the configuration for the named eMBMS service.
- **sctp statistics { all | name** *mme_embms_service_name* **}** – displays SCTP statistics for all or named "active" eMBMS service(s).

show mme-embms-service m3ap statistics all [verbose]

Notes:

The command above is used to clarify status of MBMS sessions with the following counters added to the output:

- MBMS Session Start Request
- MBMS Session Start Response
- MBMS Session Start Response Failure

show mme-embms-service all-session-info [summary]

Notes:

The command above displays counters to illustrate session information maintained at all MMEMgrs.

show mme-embms-service mce-session-association tmgi-service-id *tmgi_service_id* **[mbms-flow-id** *mbms_flow_id* **]**

Notes:

The command above displays fields and counters to illustrate configured MCE associations.

show subscribers mme-embms-only [all | full]

Notes:

The command above displays MBMS subscriber information.

Disconnect Reasons

Information for system disconnects specific to eMBMS, can be found in the statistics for the following:

- disc-reason-607 = mme-embms-call-setup-timeout(607) - The number of times an eMBMS call setup has timed out.
- disc-reason-608 = mme-embms-normal-disconnect(608) - The number of times an eMBMS call has disconnected normally.
- disc-reason-609 = mme-embms-sctp-down(609) - The number of times an eMBMS call experienced an SCTP failure.

To generate the disconnect reason statistics, use the command **show session disconnect-reasons verbose** or refer to the system schema bulk statistics.

Logging Support

The following commands identify the logging support provided for the MME eMBMS Service functionality:

```
logging filter active facility mme-embms level {critical | error | warning
 |
unusual | info | trace | debug }

 logging filter active facility m3ap level {critical | error | warning |
 unusual
| info | trace | debug }
```

Logging Events

The range of event IDs supported for eMBMS is 212001 to 212024.

The following configuration disables logging for specified event or event ranges:

```
configure
logging disable eventid event_id [ to event_id ]
```

The following configuration enables logging for specified event or event ranges:

```
configure
no logging disable eventid event_id [ to event_id ]
```

Monitor Protocol Logging

- Monitor protocol option (97-M3AP) is added to display M3AP messages.
- Monitor protocol option (74 - EGTPC) is re-used to display GTPv2 messages on Sm Interface.

Bulk Statistic Support

mme-embms is the schema that has been added to enable the MME to provide statistics specific to eMBMS on the MME. Variables included are:

- mme-embms-m3ap-recdata-m3setup-req
- mme-embms-m3ap-recdata-mce-config-upd
- mme-embms-m3ap-recdata-mbms-sess-start-rsp
- mme-embms-m3ap-recdata-mbms-sess-start-rsp-fail
- mme-embms-m3ap-recdata-mbms-sess-upd-rsp
- mme-embms-m3ap-recdata-mbms-sess-upd-rsp-fail
- mme-embms-m3ap-recdata-mbms-sess-stop-rsp
- mme-embms-m3ap-recdata-reset
- mme-embms-m3ap-recdata-reset-ack
- mme-embms-m3ap-recdata-err-ind
- mme-embms-m3ap-transdata-m3setup-rsp
- mme-embms-m3ap-transdata-m3setup-rsp-fail
- mme-embms-m3ap-transdata-mce-config-upd-ack
- mme-embms-m3ap-transdata-mce-config-upd-ack-fail
- mme-embms-m3ap-transdata-mbms-sess-start-req
- mme-embms-m3ap-transdata-mbms-sess-upd-req
- mme-embms-m3ap-transdata-mbms-sess-stop-req

- mme-embms-m3ap-transdata-reset
- mme-embms-m3ap-transdata-reset-ack
- mme-embms-m3ap-transdata-err-ind
- mme-embms-m3ap-unknown-mme-mbms-m3ap-id
- mme-embms-m3ap-unknown-mce-mbms-m3ap-id
- mme-embms-m3ap-unknown-mbms-m3ap-id-pair
- mme-embms-m3ap-tx-syntax-err
- mme-embms-m3ap-semantic-err
- mme-embms-m3ap-msg-not-compatible
- mme-embms-m3ap-abstract-syntax-err
- mme-embms-m3ap-abstract-syntax-err-reject
- mme-embms-m3ap-abstract-syntax-err-ignore-notify
- mme-embms-m3ap-abstract-syntax-err-false-constr-msg
- mme-embms-m3ap-mce-total-active
- mme-embms-m3ap-mce-total-created
- mme-embms-m3ap-mce-total-closed
- mme-embms-m3ap-mce-total-rejected

SNMP Traps

The following identifies the traps new for the MME eMBMS feature and illustrates a sample display:

```
show snmp trap statistics
```

```
 Trap Name                #Gen   #Disc  Disable  Last Generated
 -----------------------  -----  -----  -------  --------------------
 MMEEMBMSServiceStart       1      0      0      2015:09:08:09:14:08
 MMEEMBMSServiceStop        1      0      0      2015:09:08:09:14:03
 MCEAssocDown               1      0      0      2015:09:08:09:14:19
 MCEAssocUp                 1      0      0      2015:09:08:09:14:16
```