LTE Core Network

Version: 2018-07-10

Table of Contents

1 Introduction

LTEMME is a LTE MME (Mobility Management Entity) implementation. It has a built-in SGW (Serving Gateway), PGW (Packet Data Network Gateway), PCRF (Policy and Charging Rule Function), HSS (Home Subscriber Server) and EIR (Equipment Identity Register). It can easily be used with the Amarisoft LTE eNodeB to build a highly configurable LTE test network.

2 Features

- LTE release 14 compliant.
- Implements one MME with built-in SGW, PGW, PCRF, HSS and EIR.
- Supports several eNodeBs with standard S1 interface (S1AP and GTP-U protocols).
- NAS integrity check and encryption using the AES, Snow3G and ZUC algorithms. Ciphering support is now subject to export rules for your country.
- Support of USIM cards using the XOR, Milenage or TUAK authentication algorithm.
- Handling of UE procedures: attach, authentication, security configuration, detach, tracking area update, service access, radio bearer establishment, paging.
- Multi-PDN support and built-in dynamic ERAB setup for easy VoLTE/IMS testing.
- Transparent access to the IP network (no external Serving Gateway or PDN Gateway is necessary).
- Configurable access point name, IP range, DNS and E-RAB QoS.
- Support sending of Public Warning System messages (ETWS/CMAS).
- IPv6 support.
- Configurable logging system for all channels with built-in text decoders.
- Remote API using WebSocket.
- Command line monitor.
- PSM and eDRX support.
- Supports several IMS servers with Rx interface.
- Support of NB-IoT RAT and control plane CIoT optimization.
- Non-IP data delivery CIoT feature.
- Attach without PDN connectivity CIoT feature.
- User management via internal database without any external HSS.
- Support of optional S6a interface with external HSS.
- Support of optional S13 interface with external EIR.
- Support of optional SGsAP interface with external VLR/MSC.
- Support of broadcast and multicast PDN options.

3 Requirements

3.1 Hardware requirements

- LTEMME can run on the same PC as the Amarisoft eNodeB if a simple and compact solution is needed. Otherwise, any reasonnably recent PC with at least one Gigabit Ethernet port is acceptable.
- A test USIM card should be plugged into the UE. Test USIM cards from Anritsu are supported by the default configuration. Other test USIM cards should work as well provided they implement the dummy XOR authentication algorithm and that their IMSI and secret key are known. USIM cards using the Milenage or TUAK algorithm are also supported.

3.2 Software requirements

- A 64 bit Linux distribution. Fedora 26 is the officially supported distribution.
  The following distributions are known as compatible:
  - Fedora 17 to 27
  - Cent OS 7
  - Ubuntu 12 to 16

4 Installation

[Quick installation instructions are also given in the Amarisoft eNodeB documentation in case LTEMME is installed on the same PC as the eNodeB].

The network access thru the Gigabit Ethernet port must be correctly configured.

LTEMME can be run directly from the directory when it was unpacked. No need for explicit installation.

4.1 Local network configuration

LTEMME will set up a new virtual network interface `tun0` where each UE has a specific IP address. If you want them to connect to your local network (and Internet), you need to set up IP forwarding and masquerading.

An example is found in the `lte_init.sh`:
Syntax:

```
sudo ./lte_init.sh [-6] <ifname>

sudo ./lte_init.sh default
sudo ./lte_init.sh -6 eth1
```

4.2 Linux setup

LTEMME uses the SCTP protocol for which the necessary packages are not usually installed. In order to install them, do as root user:

- Fedora

  ```
  yum install lksctp-tools kernel-modules-extra
  ```

  As openssl libraries have no standard naming in Fedora and Cent OS, you need to create the following symbolic links on those distributions:

  ```
  ln -s /usr/lib64/libcrypto.so.10 /usr/lib64/libcrypto.so.1.0.0

  ln -s /usr/lib64/libssl.so.10 /usr/lib64/libssl.so.1.0.0
  ```

  On recent Fedora distributions you might have to install the compat-openssl10 package:

  ```
  sudo dnf install compat-openssl10
  ```

- Ubuntu

  ```
  sudo apt-get install lksctp-tools linux-image-extra-3.13.0-24-generic
  ```

  and reboot the PC in case the Linux kernel was upgraded too.

4.3 License key installation

LTEMME needs a license key file to run. *It is associated to your PC, so if you replace it or change its hardware configuration you must contact Amarisoft to get a new license key.*

The following steps are needed to get this license file:

- Run LTEMME:

  ```
  ./ltemme config/mme.cfg
  ```

  It says that the license key is not present and prints a 16 digit hexadecimal code.

- Send by mail to `delivery@amarisoft.com` this hexadecimal code to your contact at Amarisoft. You will get back the `ltemme.key` license key file.

- Copy the `ltemme.key` file to the `${HOME}/.amarisoft/` directory (`${HOME}` is the home directory of the `root` user). You can use the shell variable `AMARISOFT_PATH` to change this path.

Once the license key is installed, `ltemme` should start normally.

4.4 Initial testing

- Edit the file `config/mme.cfg` to set the bind address of the GTP-U interface. Normally it is the address of the default Ethernet of the PC (you can see it with `ifconfig`). You can also set the address of the DNS (`dns_addr` property). You don't need to change the other parameters for an initial test.
- LTEMME creates one virtual network interface where the UE traffic is redirected. A modification of the default routing rules and iptables is usually needed if you want to redirect the UE traffic to the local network and Internet. The script `lte_init.sh` in the Amarisoft LTEMME package gives an example of setup to configure a NAT to access the Internet.
- Start the program as root with:

  ```
  ./ltemme config/mme.cfg
  ```

  [The root access is only needed to set up the Linux virtual interface.]
- The command line interface is used to monitor the operation of LTEMME and to change the logging options. Use `help` to get the list of commands and `quit` to stop the program.
- Use `enb` to list the connected eNodeBs.
- In addition to using the log file, you can monitor the traffic between LTEMME and the eNodeBs with Wireshark. The LTE specific traffic is filtered by putting `s1ap || gtp` in the `filter` input area.
- For optimal performance, it is better to avoid fragmenting the GTP-U packets. So the Ethernet interfaces used between the eNodeBs and LTEMME should be configured to have a MTU of at least 1564 (assuming the UEs use the standard MTU of 1500). You can verify with Wireshark whether the GTP-U packets are fragmented.

5 Configuration reference

5.1 Configuration file syntax

The main configuration file uses a syntax very similar to the Javascript Object Notation (JSON) with few extensions.

1. Supported types:
   - Numbers (64 bit floating point). Notation: `13.4`
   - Complex numbers. Notation: `1.2+3*I`
   - Strings. Notation: `"string"`
   - Booleans. Notation: `true` or `false`.
   - Objects. Notation: `{ field1: value1, field2: value2, .... }`
   - Arrays. Notation: `[ value1, value2, .... ]`
2. The basic operations `+`, `-`, `*` and `/` are supported with numbers and complex numbers. `+` also concatenates strings. The operators `!`, `||`, `&&`, `==`, `!=`, `<`, `<=`, `>=`, `>` are supported too.
3. The numbers `0` and `1` are accepted as synonyms for the boolean values `false` and `true`.
4. {} at top level are optional.
5. " for property names are optional.
6. Properties can be duplicated.
   Merge will be done by recursively overriding values considering reading direction.

```
{
    value: "foo",
    value: "bar",
    sub: {
        value: "foo"
    },
    sub: {
        value: "bar"
    }
}
```

   Will be equivalent to:

```
{
    value: "bar",
    sub: {
        value: "bar"
    }
}
```

7. Files can be included using *include* keyword (must not be quoted) followed by a string (without :) representing the file to include (path is relative to current file) and terminating by a comma.
   Arrays can't be included.
   Merge will be done as for duplicate properties.
   If *file1.cfg* is:

```
    value: "foo",
    include "file2.cfg",
    foo: "foo"
```

   And *file2.cfg* is:

```
    value: "bar",
```

```
    foo: "bar"
```

Final config will be:

```
{
    value: "bar",
    foo: "foo"
}
```

8. A C like preprocessor is supported. The following preprocessor commands are available:

   `#define` *var expr*
   : Define a new variable with value *expr*. *expr* must be a valid JSON expression. Note that unlike the standard C preprocessor, *expr* is evaluated by the preprocessor.

   `#undef` *var*
   : Undefine the variable *var*.

   `#include` *expr*
   : Include the file whose filename is the evaluation of the string expression *expr*.

   `#if` *expr* Consider the following text if *expr* is true.

   `#else` Alternative of `#if` block.

   `#elif` Composition of `#else` and `#if`.

   `#endif` End of `#if` block.

   `#ifdef` *var*
   : Shortcut for `#if defined(var)`

   `#ifndef` *var*
   : Shortcut for `#if !defined(var)`

   In the JSON source, every occurrence of a defined preprocessor variable is replaced by its value.

9. Backquote strings: JSON expression can be inserted in backquote delimited strings with the `${expr}` syntax. Example: `` `abc${1+2}d` `` is evaluated as the string `"abc3d"`. Preprocessor variables can be used inside the expression.

The System Information Blocks use the ASN.1 GSER syntax defined in RFC 3641 (Generic String Encoding Rules for ASN.1 Types). The description of the exact content of the System Information Blocks can be found in 3GPP TS 36.331 (RRC).

5.2 Properties

`log_filename`
: String. Set the log filename. If no leading `/`, it is relative to the configuration file path. See [Log file format], page 43.

`log_options`
: String. Set the logging options as a comma separated list of assignments.

  - *layer*.level=*verbosity*. For each layer, the log verbosity can be set to `none`, `error`, `info` or `debug`. In debug level, the content of the transmitted data is logged.
  - *layer*.max_size=*n*. When dumping data content, at most `n` bytes are shown in hexa. For ASN.1, NAS or Diameter content, show the full content of the message if `n > 0`.

- *layer*.payload=[0|1]. Dump ASN.1, NAS, SGsAP or Diameter payload in hexadecimal.
- *layer*.key=[0|1]. Dump security keys (NAS and RRC layers).
- *layer*.crypto=[0|1]. Dump plain and ciphered data (NAS, RRC and PCDP layers).
- time=[sec|short|full]. Display the time as seconds, time only or full date and time (default = time only).
- file=cut. Close current file log and open a new one.
- file.rotate=now. Rename current log with timestamp and open new one.
- file.rotate=*size*. Rename current log every time it reaches *size* bytes open new one. Size is an integer and can be followed by K, M or G.
- file.path=*path*. When log rotation is enabled, move current log to this path instead of initial log path.
- append=[0|1]. (default=0). If 0, truncate the log file when opening it. Otherwise, append to it.

Available layers are: `nas`, `ip`, `s1ap`, `gtpu`, `rx`, `s6`, `cx`, `s13`, `sgsap`

`gtp_addr`

String. Set the IP address (and an optional port) on which the GTP-U packets are received. The default port is 2152. It is normally the IP address of the network interface connected to the core network.

Syntax:

- `"1.2.3.4"` (use default port)
- `"1.2.3.4:5678"` (use explicit port)
- `"2001:db8:0:85a3::ac1f:8001"` (IPv6 address and default port)
- `"[2001:db8:0:85a3::ac1f:8001]:5678"` (IPv6 address and explicit port)

`gtp_ext_addr`

Optional string. Set the IP address on which the eNodeB should transmit the GTP-U packets. It is the same as gtp_addr by default. It can be different if LTEMME is behind a NAT.

`gtp_payload_mtu`

Optional integer (range 68 to 16384, default = 1500). MTU in bytes for the GTP-U payload. Do not forget to update the network interface MTU accordingly for optimal performance. See [Initial testing], page 5.

`s1ap_bind_addr`

Optional string. IP address and optional port on which the S1AP SCTP connection is bound.

`plmn` String. PLMN identity of the MME (5 or 6 digits). It should match one of the PLMN identities broadcasted by the eNodeB.

`mme_group_id`

Range: 0 to 65535. Set the MME group ID.

`mme_code`

Range: 0 to 255. Set the MME code.

`relative_capacity`

Optional integer. Range: 0 to 255. Default : 50. Set the MME relative capacity value used for MME load balancing in S1AP S1 Setup Response and MME Configuration Update messages.

`nas_cipher_algo_pref`
: Array of integers. Set the preferred algorithms for NAS encryption in decreasing order of preference. If none match the UE capabilities, then EEA0 (no encryption) is selected.

  List of supported algorithms:

  `1` EEA1 (Snow 3G)

  `2` EEA2 (128 bit AES)

  `3` EEA3 (ZUC)

  If encryption is necessary, for best performance use AES (EEA2) as first choice if your CPU supports the AES NI Intel instruction set (available starting from Sandy bridge CPUs). Otherwise use Snow3G (EEA1) or ZUC (EEA3).
  Note that ciphering is subject to export rules depending on your country.

`nas_integ_algo_pref`
: Array of integers. Set the preferred algorithms for NAS integrity check in decreasing order of preference. If none match the UE capabilities, then EIA0 (no integrity check) is selected.

  List of supported algorithms:

  `1` EIA1 (Snow 3G)

  `2` EIA2 (128 bit AES)

  `3` EIA3 (ZUC)

  For best performance, use AES (EIA2) as first choice if your CPU supports the AES NI Intel instruction set (available starting from Sandy bridge CPUs). Otherwise use Snow3G (EIA1) or ZUC (EIA3).

`tun_setup_script`
: String. Set the path of the shell script to set up the virtual network interface. Script is called for each PDN with following parameters:

  1. Interface name
  2. PDN index
  3. Access Point Name
  4. IP version: 'ipv4' or 'ipv6'
  5. IP address: first IP address for ipv4 and link local address for IPv6
  6. First IP address
  7. Last IP address

  If no script is given, no virtual network interface is created.

  Note: when using several PDNs, they all share the same virtual network interface. So the IP configuration of this interface must be compatible with the IP address ranges of the PDNs.
  Take a look at *config/mme-ifup* file as an example.

`ue_to_ue_forwarding`
: Optional boolean (default = false). If true, enable UE to UE IP packet forwarding.

`t3402` Optional integer (default = -1). Value in seconds of the T3402 timer. -1 means that the timer value is not transmitted in attach accept or TAU accept so that the UE uses the default value (12 minutes).

`t3412` Optional integer (default = 1800). Value in seconds of the T3412 (TAU update) timer. -1 means that the timer is deactivated. This is the value sent to the UE in NAS signalling, unless UE is requesting the use of a longer timeout with T3412 extended value information element.

`t3412_low_priority`
Optional integer (default = t3412 value). Value in seconds of the T3412 (TAU update) timer if the UE indicates NAS signalling low priority. -1 means that the timer is deactivated. This is the value sent to the UE in NAS signalling, unless UE is requesting the use of a longer timeout with T3412 extended value information element.

`psm` Option boolean (default = true). If set to false, MME will ignore the PSM request sent by the UE.

`t3412_extended_forced`
Optional integer (default = -1). Value in seconds of the T3412 extended timer if UE uses PSM. If different from -1, the MME will ignore the value requested by the UE and will send this one instead.

`t3324_forced`
Optional integer (default = -1). Value in seconds of the T3324 timer if UE uses PSM. If different from -1, the MME will ignore the value requested by the UE and will send this one instead.

`t3346` Optional integer (default = -1). Value in seconds of the T3346 timer. The timer is transmitted in the reject messages if the EMM cause is #22 (congestion) and the value is not -1.

`t3448` Optional integer (default = -1). Value in seconds of the T3448 timer. The timer is transmitted if the value is different from -1 and the UE indicates its support in the UE network capability information element.

`t3460` Optional integer (default = 6). Value in seconds of the T3460 timer.

`edrx` Option boolean (default = true). If set to false, MME will ignore the eDRX request sent by the UE.

`edrx_ptw_wb_s1`
Optional integer (0 to 15, default = 3). 4 bits Paging Time Window length for WB-S1 UEs as defined in 3GPP 24.008 chapter 10.5.5.32.

`edrx_ptw_nb_s1`
Optional integer (0 to 15, default = 3). 4 bits Paging Time Window length for NB-S1 UEs as defined in 3GPP 24.008 chapter 10.5.5.32.

`edrx_cycle_forced`
Optional integer (-1 to 15, default = -1). 4 bits E-UTRAN eDRX cycle length duration as defined in 3GPP 24.008 chapter 10.5.5.32. If different from -1, the MME will ignore the value requested by the UE and will send this one instead.

`ims_list` Optional array. Each entry is an object defining connection to Amarisoft IMS server. This is useful for SMS over SG or 3GPP mode of IMS server when Rx interface is not used.
Each entry has following members:

`ims_addr` IP address of Amarisoft IMS server.

`bind_addr`
IP address of network interface to use for IMS connection.

`ims_vops` Optional boolean (default = false). Set the IMS voice over PS session in S1 mode supported bit of the EPS network feature support field in the NAS attach accept message (VoLTE).

`emc_bs` Optional boolean (default = false). Set the emergency bearer services in S1 mode supported bit of the EPS network feature support field in the NAS attach accept message (VoLTE, Release 9).

`emergency_number_list`
Optional array or objects. Defines a list of emergency numbers to be sent to the UE in the NAS Attach Accept and Tracking Area Update Accept messages.
Each object must contain the following parameters:

- `category` Integer. Bitmask of the category bits as defined in 3GPP TS 24.008 table 10.5.135d (bit 1: police, bit 2: ambulance, bit 3: fire brigade, bit 4: marine guard, bit 5: mountain rescue).
- `digits` String. Emergency number.

`cp_ciot_opt`
Optional boolean (default = false). If true, enable control plane CIoT optimization (if supported by the UE).

`attach_without_pdn`
Optional boolean (default = false). If true, enable attach without PDN functionality (if supported by the UE).

`network_name`
Optional string (default = empty). Set the network name in the EMM information message.

`network_short_name`
Optional string (default = empty). Set the network short name in the EMM information message.

`emm_information_time_enable`
Optional boolean (default = true). Include the time and time zone in the EMM information message.

`emm_information_enable`
Optional boolean. Default = true if `network_name` or `network_short_name` are not empty. If true, send the EMM information message after the NAS attach complete message.

`attach_reject_error`
Optional integer (default depending on scenario). Force value of EMM reject cause in NAS attach reject message.

`tracking_area_update_reject_error`
Optional integer (default depending on scenario). Force value of EMM reject cause in NAS tracking area update reject message.

`service_reject_error`
Optional integer (default depending on scenario). Force value of EMM reject cause in NAS service reject message.

`pdn_connect_reject_error`
Optional integer (default depending on scenario). Force value of ESM reject cause in NAS pdn connectivity reject message.

`pdn_disconnect_reject_error`
: Optional integer (default depending on scenario). Force value of ESM reject cause in NAS pdn disconnect reject message.

`bearer_resource_allocation_reject_error`
: Optional integer (default depending on scenario). Force value of ESM reject cause in NAS bearer resource allocation reject message.

`bearer_resource_modification_reject_error`
: Optional integer (default depending on scenario). Force value of ESM reject cause in NAS bearer resource modification reject message.

`force_eps_only`
: Optional boolean (default = false). Deprecated, use attach_result_mode instead.

`attach_result_mode`
: Optional string (default = auto). Set attach result of attach accept message.
Can be:

  `auto` This is standard LTE behavior.

  `eps_only` If set and UE is sending combined EPS/IMSI attach, the MME will answer with EPS only in attach accept message (EMM cause will be CS domain not available).

  `combined` If set and UE is sending EPS only attach, the MME will answer with combined in attach accept message.

`additional_update_result`
: Optional integer (default = 2). Set the value of additional update result in NAS attach accept message.
If set to -1, the additional update result won't be set.

`imeisv_request_in_smc`
: Optional boolean (default = true). Ask for the UE IMEISV in the NAS security mode command message. Must be enabled if `multi_sim` is set to `true`. IMEISV will always be requested if a S13 connection is defined, or if me_db object is defined.

`force_nas_authentication`
: Optional boolean (default = false). If set to true, MME will force a new NAS authentication procedure even if the Attach Request was already successfully authentified. Deprecated, use `authentication_mode` instead.

`authentication_mode`
: Optional string (default = auto). Set NAS authentication procedure behavior.
Can be:

  `auto` The MME performs authentication procedure unless the UE is already successfully authentified.

  `force` MME forces a new NAS authentication procedure even if the Attach Request was already successfully authentified

  `skip` The MME skips the NAS authentication procedure and uses EIA0/EEA0 algorithms. This needs to be supported on UE side also.

`force_guti_in_tau`
: Optional boolean (default = false). If set, GUTI IE will be systematically present in Tracking Area Update Accept message.

`attach_reject_filter`
: Optional object. Represent UE to reject when trying to attach.
Each property name represent IMSI. If set to `"*"`, every UE will be redirected using this filter.
Each property value is an integer defining the redirection type as described in *rrc_redirect* eNB configuration.

Example:

```
attach_reject_filter: {
    "*": 0,
    "0010112345678": 1
}
```

Will reject UE with IMSI *0010112345678* using redirection configuration 1 and all other UEs using redirection configuration 0.

`emm_procedure_filter`
: Optional object. Allows to define the MME behavior for a list of EMM procedures.
Each property name represents an EMM procedure. The ones currently supported are `attach`, `tracking_area_updating`, `detach`, `service_request`, `identity`, `authentication`, `security_mode_control` and `nas_transport`.
Each property value is an enum: `treat` (UE message is processed), `ignore` (UE message is ignored) or `reject` (UE message is rejected). Network initiated procedures cannot be rejected.
By default all procedures are treated.

Example:

```
emm_procedure_filter: {
    attach: "treat",
    service_request: "reject"
}
```

`qci_dscp_mapping`
: Optional array of objects. Allows to define a specific IP differentiated services code point for a given QCI. QCI not explicitly configured use the default DSCP value 0.
Each object must contain the following properties:

  `qci` Integer (range 1 to 254). QCI value.

  `dscp` Integer (range 0 to 63). DSCP value.

`com_addr` Optional string. Address of the WebSocket server remote API. See [Remote API], page 26.
If set, the WebSocket server for remote API will be enabled and bound to this address.
Default port is 9000.
Setting IP address to 0.0.0.0 will make remote API reachable through all network interfaces.

`com_name` Optional string. Sets server name. MME by default

`com_ssl_certificate`
: Optional string. If set, forces SSL for WebSockets. Defines CA certificate filename.

`com_ssl_key`
: Optional string. Mandatory if *com_ssl_certificate* is set. Defines CA private key filename.

`com_ssl_peer_verify`
: Optional boolean (default is false). If *true*, server will check client certificate.

`license_server`
: Configuration of the Amarisoft license server to use.
Object with following properties:

  `server_addr`
  : String. IP address of the license server.

  `name`
  : Optional string. Text to be displayed inside server monitor or remote API.

  `tag`
  : Optional string. If set, server will only allow license with same tag.

  Example:

```
license_server: {
    server_addr: "192.168.0.20"
}
```

5.2.1 PDN options

`ignore_initial_apn`
: Optional boolean (default = false). If false, UE will be rejected if its desired APN is unknown.

`pdn_list`
: Array of objects. Configure the available Packet Data Networks. The first one is the one to which the UE accesses at the initial attach.

  Each object contains the following properties:

  `access_point_name`
  : String. Set the Access Point Name. Use dots (.) to separate the APN elements.
  Array of string. You can use array to define aliases.

  `pdn_type`
  : Optional enumeration: ipv4, ipv6, ipv4v6, non-ip (default = ipv4). Select the PDN type.

  `first_ip_addr`
  : String. First available IPv4 address.

  `last_ip_addr`
  : String. Last available IPv4 address.

  `gateway`
  : Optional string. If set, forces the address of the gateway used for this PDN and passed to mme-ifup script. With default config, it will be used to provide a IP address to the tun interface.
  If not set, the first IP of the subnet will be used.

  `ip_addr_shift`
  : Optional integer (default = 0). The allocated IPv4 addresses are allocated starting from `first_ip_addr` with a difference of `2^ip_addr_shift`. Hence `last_ip_addr - first_ip_addr` must be a multiple of `2^ip_addr_shift`. This option is useful in case of inter-UE communication to ensure that the IPv4 address of a given UE is the only one in its netmask.

`first_ipv6_prefix`
: String. First available global IPv6 prefix used in Router Advertisement message sent to the UE (only the high 8 bytes of the IPv6 address are meaningful). Note that the selected prefix will also be used as the interface identifier sent in NAS signalling.

`last_ipv6_prefix`
: String. Last available global IPv6 prefix used in Router Advertisement message sent to the UE (only the high 8 bytes of the IPv6 address are meaningful). Note that the selected prefix will also be used as the interface identifier sent in NAS signalling.

`ipv6_interface_identifier`
: String. Interface identifier for the MME network interface of this PDN (only the low 8 bytes of the IPv6 address are meaningful).

`ipv6_router_lifetime`
: Optional integer (range 0 to 65535, default is 65535). IPv6 Router Advertisement router lifetime in seconds.

`ipv6_valid_lifetime`
: Optional integer (default is infinity - 0xffffffff). IPv6 Router Advertisement valid lifetime in seconds.

`ipv6_pref_lifetime`
: Optional integer (default is `ipv6_valid_lifetime` value). IPv6 Router Advertisement preferred lifetime in seconds.
Must not be greater than `ipv6_valid_lifetime`.

`ipv6_onlink_flag`
: Optional boolean (default is true). Defines IPv6 Router Advertisement on-link flag state.

`ipv6_managed_addr_config_flag`
: Optional boolean (default is false). Defines IPv6 Router Advertisement managed address configuration flag state.

`ipv6_other_config_flag`
: Optional boolean (default is false). Defines IPv6 Router Advertisement other configuration flag state.

`ipv6_ra_transmission_interval`
: Optional integer (range -1 to 1800, default is 0). Time in seconds between 2 periodical multicast Router Advertisement transmission, once the initial 3 transmissions have been performed after opening the PDN. The value -1 means that no multicast transmission is done at all (including the 3 initial ones). The value 0 means that periodical transmission is deactivated.

`ipv6_drop_rs`
: Optional boolean (default is false). Defines whether the incoming Router Solicitation messages should be dropped by the MME or not.

`ipv6_prefix_delegation_count`
: Optional integer (2, 4, 8, 16, 32). Defines the number of prefixes delegated by DHCPv6-PD (including the one allocated by the Router Advertisement message). Only the first IA_PD option in the DHCPv6 Solicit message is considered.

`dns_addr` String or Array of strings. IPv4 or IPv6 addresses of the DNS servers.

`p_cscf_addr`
Optional string or Array of strings. IPv4 or IPv6 addresses of the P-CSCF servers (VoLTE).

`mtu_ipv4` Optional integer. Set MTU size (0 means disabled).

`mtu_non_ip`
Optional integer. Set MTU size fo non-IP PDN (0 means disabled, the minimum valid value is 128).

`operator` Optional array of objects. Each element defines an operator reserved container in protocol configuration.

Properties of each element:

`id` Integer. Container identifier, must be between 0xff00 and 0xffff as defined in TS 24.008.

`plmn` String. PLMN info of container.

`value` String. Value to send in hexadecimal string format.

`force` Optional boolean. If true, container will be sent event without request (false by default).

`authentication`
Optional enumeration: `none`, `pap` or `chap` (default set to `none`). Defines the authentication mechanism used for this APN.

`username` Optional string (up to 100 characters) containing the user name used for `pap` or `chap` authentication.

`password` Optional string (up to 100 characters) containing the password used for `pap` or `chap` authentication.

`apn_aggregate_max_bitrate_dl`
Optional integer (default = -1). APN aggregate maximum bitrate for downlink (in bits/s). If set to -1, no APN-AMBR is configured and UE-AMBR is used instead.

`apn_aggregate_max_bitrate_ul`
Optional integer (default = -1). APN aggregate maximum bitrate for uplink (in bits/s). If set to -1, no APN-AMBR is configured and UE-AMBR is used instead.

`emergency`
Optional boolean (default = false). If set, PDN will be selected for emergency calls.

`serving_plmn_rate_control`
Optional integer (range 0 to 65535, default = 0). Defines the serving PLMN rate control IE content when PDN is used with control plane CIoT optimization only. If the value configured is less than 10, the IE is not transmitted.

`apn_rate_control_params`
Optional object. If defined, and if the UE indicates APN rate control parameters support in its protocol configuration options, the following parameters will be sent in MME protocol configuration options:

`additional_exception_report`
: Boolean. Indicates if exception reports are allowed once the limit is reached.

`ul_time_unit`
: Enumeration: `unrestricted`, `minute`, `hour`, `day` or `week`.

`max_ul_rate`
: Integer (range from 0 to 16777215). Number of messages allowed to be sent per `ul_time_unit`.

`additional_apn_rate_control_exception_data_params`
: Optional object. If defined, and if the UE indicates additional APN rate control for exception data parameters support in its protocol configuration options, the following parameters will be sent in MME protocol configuration options:

`ul_time_unit`
: Enumeration: `unrestricted`, `minute`, `hour`, `day` or `week`.

`max_ul_rate`
: Integer (range from 0 to 65535). Number of messages allowed to be sent per `ul_time_unit`.

`t3396`
: Optional integer (default = -1). Value in seconds of the T3396 timer. The timer is transmitted in the ESM reject messages if the value is not -1.

`esm_procedure_filter`
: Optional object. Allows to define the MME behavior for a list of ESM procedures.
Each property name represents an ESM procedure. The ones currently supported are `pdn_connectivity`, `pdn_disconnect`, `bearer_resource_allocation` and `bearer_resource_modification`.
Each property value is an enum: `treat` (UE message is processed), `ignore` (UE message is ignored) or `reject` (UE message is rejected).
By default all procedures are treated.

Example:

```
esm_procedure_filter: {
    pdn_connectivity: "treat",
    bearer_resource_allocation: "reject"
}
```

`tun_setup_script`
: Overrides [tun_setup_script], page 9, for this PDN.

`tun_ifname`
: Optional string. If set, use this tun device instead of creating it.
Usefull when LTEMME has no root privileges.

`erabs`

: Array of objects. Each element defines an E-RAB (E-UTRAN Radio Access Bearer) associated to the PDN. The first E-RAB is the default radio bearer and must always be present. The additional E-RABs are dedicated radio bearers and must include a Traffic Flow Template (TFT) unless they are defined as UE initiated.

Property of each element:

`qci` Range: 1 to 255. QoS Class Identifier of the E-RAB.

`priority_level`
Range: 1 to 15. Priority level.

`pre_emption_capability`
Enumeration: `shall_not_trigger_pre_emption` or `may_trigger_pre_emption`.

`pre_emption_vulnerability`
Enumeration: `not_pre_emptable` or `pre_emptable`.

`setup_type`
Optional enumeration: automatic, on_demand, ue_initiated (default = automatic).

- If set to automatic, the dedicated bearer is created with the default bearer.
- If set to on_demand, the dedicated bearer is created when there is downlink traffic matching the TFT filters. This option is useful to automatically create a dedicated bearer for IMS RTP voice traffic.
- If set to ue_initiated, the dedicated bearer is created when receiving a ESM bearer resource allocation request message. In that case, the `gbr` object defines the maximum values allowed (MME will use the minimum between configured values and the ones sent by the UE) and `tft` object is not required (MME will use the filters sent by the UE).

`gbr` Optional object. Guaranted Bitrate information. List of properties:

`maximum_bitrate_dl`
Integer.

`maximum_bitrate_ul`
Integer.

`guaranteed_bitrate_dl`
Integer.

`guaranteed_bitrate_ul`
Integer.

`filters` Optional array of objects. List of TFT filters. Required for dedicated bearers with `setup_type` different from ue_initiated. Each filter has the following properties:

`direction`
Enumeration: `dl`, `ul` or `both`. Set the filter direction.

`id` Range: 0 to 15. Set the filter identifier.

`precedence`
Range: 0 to 255. Set the filter precedence. All the filters must have different precedence. 0 is the highest precedence.

`components`
: Array of objects. Each component contains one of the following properties as described in 3GPP 23.060 chapter 15.3.2:

  `ipv4_remote_addr`
  : String. Match a remote (external network entity) IPv4 address with the additional `mask` property.

  `ipv6_remote_addr`
  : String. Match a remote (external network entity) IPv6 address with the additional `mask` property.

  `proto_id`
  : Range: 0 to 255. Match against the IP protocol identifier.

  `local_port`
  : Range: 0 to 65536. Match against the local (UE) port.

  `local_port_range`
  : Array of 2 integers. Match against a local (UE) port range.

  `remote_port`
  : Range: 0 to 65536. Match against the remote (external network entity) port.

  `remote_port_range`
  : Array of 2 integers. Match against a remote (external network entity) port range.

  `security_parameter_index`
  : 32 bit integer. Match the ESP or AH security parameter index.

  `type_of_service`
  : Range: 0 to 255. Match the type of service (IPv4) or the traffic class (IPv6) field. The additional `mask` property is the corresponding mask.

  `mask`
  : Depends on TFT component.
    If ipv4_remote_addr is set, string representing IPv4 address used as a mask to apply on packet remote address.
    If ipv6_remote_addr is set, string representing IPv6 address used as a mask to apply on packet remote address.

If type_of_service is set, integer between 0 and 255 used as a mask to apply on packet tos.

`flow_label`
20 bit integer. Match the IPv6 flow label.

`on_demand_timeout`
Optional integer. When `setup_type` is on_demand, set the duration (in ms) after which the dedicated bearer is released when there is no downlink or uplink traffic.

`on_demand_ul_trigger`
Optional boolean (default = false). When `setup_type` is on_demand, if set to true an UL packet matching one of the TFT filters triggers the dedicated E-RAB establishment.

`transaction_identifier`
Optional integer (range 0 to 127). If present, the transaction identifier IE is put in the EPS bearer activation message.

`llc_sapi` Optional integer (range 0 to 15). If present, the LLC service access point identifier IE is put in the EPS bearer activation message.

`radio_priority`
Optional integer (range 0 to 7). If present, the radio priority IE is put in the EPS bearer activation message.

`packet_flow_identifier`
Optional integer (range 0 to 127). If present, the packet flow identifier IE is put in the EPS bearer activation message.

`sm_qos` Optional string. If present, the quality of service IE is put in the EPS bearer activation message. The string must contain the hexadecimal representation of the IE without its IEI and length.

5.2.2 User database options

`ue_db`

Array of objects. Configure the user database. Each element is an entry for one user. The following properties are available:

`imsi` String. Set the IMSI.

`msisdn` Optional string. Set the MSISDN.

`sim_algo` Optional enumeration. xor, milenage or tuak (default = xor). Set the SIM authentication algorithm. Note: test USIM cards use the XOR algorithm.

`amf` Range: 0 to 65535. Set the Authentication Management Field.

`sqn` Optional String (6 byte hexadecimal string). Default = `"000000000000"`. Set the initial sequence number. For the XOR algorithm, the actual value does not matter. For the Milenage or TUAK algorithm, a sequence number resynchronization is initiated if the sequence number does not match the one stored in the USIM.

`K` String. Set the user secret key (as a 16 byte hexadecimal string).

`op` Optional string. Operator key (as a 16 byte hexadecimal string). When the Milenage authentication algorithm is used, either `op` or `opc` must be set.

`opc` Optional string. Operator key preprocessed with the user secret key (as a 16 byte hexadecimal string). When the Milenage authentication algorithm is used, either `op` or `opc` must be set.

`r` Optional array of 5 integers (range: 0 to 127). Allows to customize the r1 to r5 parameters when Milenage authentication algorithm is used. If the array is not present, the default values (as defined in 3GPP 35.206) are used.

`c` Optional array of 5 strings. Each value contains a 16 byte hexadecimal string. Allows to customize the c1 to c5 parameters when Milenage authentication algorithm is used. If the array is not present, the default values (as defined in 3GPP 35.206) are used.

`top` Optional string. Operator key (as a 32 byte hexadecimal string). When the TUAK authentication algorithm is used, either `top` or `topc` must be set.

`topc` Optional string. Operator key preprocessed with the user secret key (as a 32 byte hexadecimal string). When the TUAK authentication algorithm is used, either `top` or `topc` must be set.

`keccak_iter`
Optional integer (range: 1 to MAX_INT). Allows to customize the number of Keccak permutations performed when using the TUAK authentication algorithm. If the item is not present, the default value 1 (as defined in 3GPP 35.231) is used.

`res_len` Optional integer (default = 8). Defines length of response in bytes during authentication. For TUAK authentication algorithm, the value must be 4, 8 or 16 bytes long.

`multi_sim`
Optional boolean (default = false). If true, allow several UEs to have the same IMSI (useful when using several identifical test SIM cards in different UEs at the same time). They are distinguished with their IMEI. Note: it is only allowed with the XOR authentication algorithm.

`ue_aggregate_max_bitrate_dl`
Optional integer (default = 1e9). UE aggregate maximum bitrate for downlink (in bits/s).

`ue_aggregate_max_bitrate_ul`
Optional integer (default = 1e9). UE aggregate maximum bitrate for uplink (in bits/s).

`count` Optional integer (default = 1). Create `n` user entries by incrementing the IMSI and K.

`pdn_list` Optional array. Each entry will set specific parameters for a PDN as defined below:

`access_point_name`
String. Used to define what PDN to configure.

`default` Optional boolean (default = false). If true and UE does not specify APN during Attach procedure, this PDN will be used.

`ipv4_addr`
Optional string. If set, the UE will always use this IPv4 address.

`ipv6_prefix`
Optional string. If set, the UE will always use this IPv6 prefix.

`imeisv` Optional string. If set, this configuration only applies to UE with matching IMEISV.

`multicast`
Optional boolean (default = false). If set, IPv4 multicast traffic will be forwarded to this PDN.

`broadcast`
Optional boolean (default = false). If set, IPv4 broadcast traffic will be forwarded to this PDN.

`routes` Optional array. Each entry of array represent a list of filters. See [TFT components], page 18, for filters syntax except that remote refers to UE and local to network.
When a packet enters MME, if it matches one of the filter list, it will be sent to associated UE.

Ex:

```
routes: [
          [
            {
              ipv4_remote_addr: "10.0.0.0",
              mask: "255.255.255.0"
            }
          ]
        ]
```

Means that all packet addressed to 10.0.0.0/24 network will be sent to UE.

`ue_db_filename`
Optional string. If present, store the current MME state in a persistent file. The MME state contains in particular the allocated TMSI, the associated security contexts and the allocated IP addresses.

5.2.3 Public Warning System (ETWS/CMAS) options

`pws_msgs`

Optional array of objects. Define a list of ETWS/CMAS messages which can be sent to the connected eNodeBs with the `pws_write` monitor command. Check TS 23.041 to have the exact definition of each field. Each message contains the following properties:

`local_identifier`
Range: 0 to 65535. Local message identifier. Used as argument to the monitor commands `pws_write` or `pws_kill`.

`message_identifier`
: Range: 0 to 65535. Message Identifier.

`serial_number`
: Range: 0 to 65535. Serial Number.

`repetition_period`
: Optional integer, range: 0 to 4095 (default = 10). Periodicity of the warning message to be broadcast.

`number_of_broadcasts_requested`
: Optional integer, range: 0 to 65535 (default = 65535). Number of times a message is to be broadcast.

`warning_type`
: Optional integer. Range: 0 to 65535. Warning type (ETWS only).

`warning_security_info`
: Optional 50 byte hexadecimal string. Warning security information (ETWS optional).

`warning_message`
: Optional array of string. Message content (ETWS: optional, CMAS: mandatory). Each string is a message page and contains at most 93 characters. At most 15 pages are allowed.

`data_coding_scheme`
: Optional integer. Range 0 to 255. Data coding scheme. Must be present if `warning_message` is present.

`concurrent_warning_message_ind`
: Optional boolean (default = false). Indicates to eNB that the received warning message is a new message to be scheduled for concurrent broadcast with any other ongoing broadcast of warning messages.

5.2.4 NAS special conformance testing options

The MME can automatically activate UE test mode during attachment and configure test loop mode A, B or G (see 3PP 36.509 for details). Once the loop is closed, the user can transmit downlink IP packets to the UE that will be loopbacked in UL.

`nas_test_procedure`
: Optional object allowing to configure the test procedure. It must contain the following objects:

  `test_loop_mode`
  : Enumeration: none, a, b, g. Defines which test loop will be activated.

  `lb_setup_list`
  : Optional array used for test loop mode A if UL PDCP SDU scaling is required. Each element of the array must contain the following 2 objects:

    `ul_pdcp_sdu_size`
    : Integer (range 0 to 1520). UL PDCP SDU size in bytes.

    `drb_id`
    : Integer (range 1 to 32). Data Radio Bearer identity on which the UL PDCP SDU scaling is applied.

  `ip_pdu_delay`
  : Integer (range 0 to 255). Transmission delay in seconds of the UL PDCP SDUs when operating in test loop mode B.

`operation_mode`
: Enumeration (`upper` or `rlc`). `upper` means that data is returned in uplink at the EMM entity. `rlc` means that data is returned in uplink at the RLC AM-SAP of SRB1bis for NB-IoT UE or at the RLC AM-SAP of SRB2 for E-UTRA UE. Used in test loop mode G.

`repetitions`
: Integer (0 to 127). Number of repetitions of received content of received user data in downlink in uplink. Used in test loop mode G.

`ul_data_delay`
: Integer (0 to 255). Uplink data delay in seconds. Used in test loop mode G.

5.2.5 Rx options

`rx`

Optional object allowing to configure the Rx options. It can contain the following objects:

`bind_addr`
: Optional string. IP address and optional port on which the Rx SCTP connection is bound. The default address is the same as the S1AP SCTP connection and the default port is 3368.

`qci`
: Optional object. It can contain five integer properties: audio, video, application, data and control that defines the QCI to use.
Default is 1 for audio, 2 for video and application, 6 for data and control.

`origin_realm`
: Optional string. Defines the string sent in the Origin-Realm AVP for Rx messages. Default is set to `mnc<MNC>.mcc<MCC>.3gppnetwork.org`.

`origin_host`
: Optional string. Defines the string sent in the Origin-Host AVP for Rx messages. Default is set to `epc.mnc<MNC>.mcc<MCC>.3gppnetwork.org`.

`reservation_priority`
: Optional array of 16 elements defining the S1AP ARP (Allocation and Retention Priority) parameters to be used for each Rx reservation priority level. If not present, `priority_level` is set to 15 (no priority), `pre_emption_capability` is set `shall_not_trigger_pre_emption` and `pre_emption_vulnerability` is set to `not_pre_emptable`. If present the array must be ordered by increasing Rx priority level (from 0 to 15) and must contain the following fields:

  `priority_level`
  : Range: 1 to 15.

  `pre_emption_capability`
  : Enumeration: `shall_not_trigger_pre_emption` or `may_trigger_pre_emption`.

  `pre_emption_vulnerability`
  : Enumeration: `not_pre_emptable` or `pre_emptable`.

5.2.6 S6a options

`s6`

Optional object allowing to configure the S6a options. It can contain the following objects:

`server_addr`
String. IP address and optional port of the HSS used for S6a interface. The default port is 3368.

`bind_addr`
Optional string. IP address and optional port on which the S6a SCTP connection is bound. The default address is the same as the S1AP SCTP connection.

`origin_realm`
Optional string. Defines the string sent in the Origin-Realm AVP for S6 messages. Default is set to `mnc<MNC>.mcc<MCC>.3gppnetwork.org`.

`origin_host`
Optional string. Defines the string sent in the Origin-Host AVP for S6 messages. Default is set to `epc.mnc<MNC>.mcc<MCC>.3gppnetwork.org`.

`transaction_timeout`
Optional integer (range 1 to 15000, default = 2000). Defines the timeout in milliseconds for a transaction with the HSS.

5.2.7 EIR/S13 options

`me_db`

Optional object allowing to define a list of IMEI (14 digits without the last Check Digit one) or IMEISV (16 digits), and their status (whitelisted, blacklisted, greylisted). If not present, all devices are considered as whitelisted.
It can contain the following objects:

`default_status`
Enumeration (whitelisted, blacklisted, greylisted). Defines the default status for devices not explicitly defined in the next objects.

`whitelist`
Optional array. It contains a list of IMEI or IMEISV whitelisted.

`blacklist`
Optional array. It contains a list of IMEI or IMEISV blacklisted.

`greylist` Optional array. It contains a list of IMEI or IMEISV greylisted.

Example:

```
me_db: {
  default_status: "blacklisted",
  whitelist: [
    "01234567100000",
    "0123456700000001"
  ]
}
```

`s13`

Optional object allowing to configure the S13 options. It can contain the following objects:

`server_addr`
: String. IP address and optional port of the EIR used for S13 interface. The default port is 3368.

`bind_addr`
: Optional string. IP address and optional port on which the S13 SCTP connection is bound. The default address is the same as the S1AP SCTP connection.

`origin_realm`
: Optional string. Defines the string sent in the Origin-Realm AVP for S13 messages. Default is set to `mnc<MNC>.mcc<MCC>.3gppnetwork.org`.

`origin_host`
: Optional string. Defines the string sent in the Origin-Host AVP for S13 messages. Default is set to `epc.mnc<MNC>.mcc<MCC>.3gppnetwork.org`.

`transaction_timeout`
: Optional integer (range 1 to 15000, default = 2000). Defines the timeout in milliseconds for a transaction with the EIR.

5.2.8 SGs options

`sgs`

Optional object allowing to configure the SGs options. It can contain the following objects:

`csfb_allowed`
: Optional boolean (default = false). If set to true, Circuit Switched Fall back procedures are accepted, otherwise they are rejected.

`lac`
: Optional integer (default = 0x001). Defines the Location Area Identifier of the MSC/VLR to connect to.

`server_addr`
: String. IP address and optional port of the MSC/VLR used for SGs interface. The default port is 29118.

`bind_addr`
: Optional string. IP address and optional port on which the SGs SCTP connection is bound. The default address is the same as the S1AP SCTP connection.

6 Remote API

You can access LTEMME via a remote API.
Protocol used is WebSocket as defined in RFC 6455 (`https://tools.ietf.org/html/rfc6455`).

6.1 Messages

Messages exchanged between client and LTEMME server are in strict JSON format.

Each message is represented by an object. Multiple message can be sent to server using an array of message objects.

Time and delay values are floating number in seconds.

All messages have at least following definition:

`message` String. Represent type of message. This parameter is mandatory and depending on its value, other parameters will apply.
If message is a response from server, response message will have same message member.

`message_id`
Optional any type. If set response sent by the server to this message will have same message_id. This is used to identify response as WebSocket does not provide such a concept.

`start_time`
Optional double. Represent the delay before executing the message.
If not set, the message is executed when received.
Note that some command (*log_get*, *log_reset*, *config_get*, *config_set*, *stats*) can't be executed in future.

`absolute_time`
Optional boolean (default = false). If set, `start_time` is interpreted as absolute.
You can get current clock of system using `time` member of `config_get` command.

6.2 Errors

If a message produces an error, response will have an error string field representing the error.

6.3 Sample nodejs program

You will find in this documentation a sample program: `ws.js`.
This is a nodejs program that allow to send message to `PROG`.
It requires nodejs to be installed:

```
yum install nodejs npm
npm install nodejs-websocket
```

Then simply start it with server name and message you want to send:

```
./ws.js 127.0.0.1:9000 '{"message": "config_get"}'
```

6.4 Common messages

`config_get`
: Retrieve current config.

  Response definition:

  `type` Always "MME"

  `name` String representing server name.

  `time` Number representing time in seconds.
  Usefull to send command with absolute time.

  `logs` Object representing log configuration.
  With following elements:

  `layers` Object. Each member of the object represent a log layer configuration:

  `layer name`
  Object. The member name represent log layer name and parameters are:

  `level` See [log_options], page 7,

  `max_size` See [log_options], page 7,

  `count` Number. Number of bufferizer logs.

`config_set`
: Change current config.
  Each member is optional.
  Message definition:

  `logs` Object. Represent logs configuration. Same structure as config_get (See [config_get logs member], page 28).
  All elements are optional.

  `relative_capacity`
  Optional integer. Range: 0 to 255. Default : 50. Set the MME relative capacity value used for MME load balancing in S1AP S1 Setup Response and MME Configuration Update messages.

  `attach_reject_error`
  Optional integer. Forces attach reject EMM cause.

  `tracking_area_update_reject_error`
  Optional integer. Forces tracking area update reject EMM cause.

  `service_reject_error`
  Optional integer. Forces value of EMM reject cause in NAS service reject message.

  `pdn_connect_reject_error`
  Optional integer. Forces pdn connectivity reject ESM cause.

  `pdn_disconnect_reject_error`
  Optional integer. Forces value of ESM reject cause in NAS pdn disconnect reject message.

`bearer_resource_allocation_reject_error`
: Optional integer. Forces value of ESM reject cause in NAS bearer resource allocation reject message.

`bearer_resource_modification_reject_error`
: Optional integer. Forces value of ESM reject cause in NAS bearer resource modification reject message.

`attach_reject_filter`
: Optional Object. Represent UE to reject when trying to attach.
Each property name represent IMSI. If set tp `"*"`, every UE will be redirected using this filter.
Each property value may be:

  `null` Removes redirection matching IMSI

  `integer` Defines redirection type as described in *rrc_redirect* eNB configuration.

  `string` Defines PLMN to redirect to

`t3402` Optional integer. Value in seconds of the T3402 timer. -1 means that the timer value is not transmitted in attach accept or TAU accept so that the UE uses the default value (12 minutes).

`t3412` Optional integer. Value in seconds of the T3412 (TAU update) timer. -1 means that the timer is deactivated.

`t3412_low_priority`
: Optional integer. Value in seconds of the T3412 (TAU update) timer if the UE indicates NAS signalling low priority. -1 means that the timer is deactivated.

`psm` Option boolean (default = true). If set to false, MME will ignore the PSM request sent by the UE.

`t3412_extended_forced`
: Optional integer. Value in seconds of the T3412 extended timer if UE uses PSM. If different from -1, the MME will ignore the value requested by the UE and will send this one instead.

`t3324_forced`
: Optional integer. Value in seconds of the T3324 timer if UE uses PSM. If different from -1, the MME will ignore the value requested by the UE and will send this one instead.

`t3346` Optional integer. Value in seconds of the T3346 timer. The timer is transmitted in the reject messages if the EMM cause is #22 (congestion) and the value is not -1.

`t3448` Optional integer (default = -1). Value in seconds of the T3448 timer. The timer is transmitted if the value is different from -1 and the UE indicates its support in the UE network capability information element.

`t3460` Optional integer (default = 6). Value in seconds of the T3460 timer.

`edrx` Option boolean (default = true). If set to false, MME will ignore the eDRX request sent by the UE.

`edrx_ptw_wb_s1`
: Optional integer. 4 bits Paging Time Window length for WB-S1 UEs as defined in 3GPP 24.008 chapter 10.5.5.32.

`edrx_ptw_nb_s1`
: Optional integer. 4 bits Paging Time Window length for NB-S1 UEs as defined in 3GPP 24.008 chapter 10.5.5.32.

`edrx_cycle_forced`
: Optional integer. 4 bits E-UTRAN eDRX cycle length duration as defined in 3GPP 24.008 chapter 10.5.5.32. If different from -1, the MME will ignore the value requested by the UE and will send this one instead.

`ims_vops`
: Optional boolean. Set the IMS voice over PS session in S1 mode supported bit of the EPS network feature support field in the NAS attach accept message (VoLTE).

`emc_bs`
: Optional boolean. Set the emergency bearer services in S1 mode supported bit of the EPS network feature support field in the NAS attach accept message (VoLTE, Release 9).

`cp_ciot_opt`
: Optional boolean. If true, enable control plane CIoT optimization (if supported by the UE).

`attach_without_pdn`
: Optional boolean. If true, enable attach without PDN functionality (if supported by the UE).

`attach_result_mode`
: Optional string. Set attach result of attach accept message.
Can be:

  `auto`
  : This is standard LTE behavior.

  `eps_only`
  : If set and UE is sending combined EPS/IMSI attach, the MME will answer with EPS only in attach accept message (EMM cause will be CS domain not available).

  `combined`
  : If set and UE is sending EPS only attach, the MME will answer with combined in attach accept message.

`additional_update_result`
: Optional integer. Set the value of additional update result in NAS attach accept message.
If set to -1, the additional update result won't be set.

`force_nas_authentication`
: Optional boolean. If set to true, MME will force a new NAS authentication procedure even if the Attach Request was already successfully authentified. Deprecated, use `authentication_mode` instead.

`authentication_mode`
: Optional string (default = auto). Set NAS authentication procedure behavior.
Can be:

  `auto`
  : The MME performs authentication procedure unless the UE is already successfully authentified.

`force` MME forces a new NAS authentication procedure even if the Attach Request was already successfully authentified

`skip` The MME skips the NAS authentication procedure and uses EIA0/EEA0 algorithms. This needs to be supported on UE side also.

`force_guti_in_tau`
Optional boolean (default = false). If set, GUTI IE will be systematically present in Tracking Area Update Accept message.

`emm_procedure_filter`
Optional object. Allows to define the MME behavior for a list of EMM procedures.
Each property name represents an EMM procedure. The ones currently supported are `attach`, `tracking_area_updating`, `detach`, `service_request`, `identity`, `authentication`, `security_mode_control` and `nas_transport`.
Each property value is an enum `treat` (UE message is processed), `ignore` (UE message is ignored) or `reject` (UE message is rejected). Network initiated procedures cannot be rejected.

Example:

```
emm_procedure_filter: {
    attach: "treat",
    service_request: "reject"
}
```

`pdn_list` Optional array of object. Each object can contain the following properties:

`apn` String. APN allowing to identify the PDN to be modified.

`operator` Optional array of objects. Each element defines an operator reserved container in protocol configuration.

Properties of each element:

`id` Integer. Container identifier, must be between 0xff00 and 0xffff as defined in TS 24.008.

`plmn` String. PLMN info of container.

`value` String. Value to send in hexadecimal string format.

`force` Optional boolean. If true, container will be sent event without request (false by default).

`serving_plmn_rate_control`
Optional integer (range 0 to 65535). Defines the serving PLMN rate control IE content when PDN is used with control plane CIoT optimization only. If the value configured is less than 10, the IE is not transmitted.

`apn_rate_control_params`
Optional object. If defined, and if the UE indicates APN rate control parameters support in its protocol configura-

tion options, the following parameters will be sent in MME protocol configuration options:

`additional_exception_report`
: Boolean. Indicates if exception reports are allowed once the limit is reached.

`ul_time_unit`
: Enumeration: `unrestricted`, `minute`, `hour`, `day` or `week`.

`max_ul_rate`
: Integer (range from 0 to 16777215). Number of messages allowed to be sent per `ul_time_unit`.

`additional_apn_rate_control_exception_data_params`
: Optional object. If defined, and if the UE indicates additional APN rate control for exception data parameters support in its protocol configuration options, the following parameters will be sent in MME protocol configuration options:

`ul_time_unit`
: Enumeration: `unrestricted`, `minute`, `hour`, `day` or `week`.

`max_ul_rate`
: Integer (range from 0 to 65535). Number of messages allowed to be sent per `ul_time_unit`.

`t3396`
: Optional integer. Value in seconds of the T3396 timer. The timer is transmitted in the ESM reject messages if the value is not -1.

`esm_procedure_filter`
: Optional object. Allows to define the MME behavior for a list of ESM procedures.
Each property name represents an ESM procedure. The ones currently supported are `pdn_connectivity`, `pdn_disconnect`, `bearer_resource_allocation` and `bearer_resource_modification`.
Each property value is an enum: `treat` (UE message is processed), `ignore` (UE message is ignored) or `reject` (UE message is rejected).

Example:

```
esm_procedure_filter: {
    pdn_connectivity: "treat",
    bearer_resource_allocation: "reject"
}
```

`ipv6_router_lifetime`
: Optional integer (range 0 to 65535). IPv6 Router Advertisement router lifetime in seconds.

`ipv6_valid_lifetime`
: Optional integer. IPv6 Router Advertisement valid lifetime in seconds.

`ipv6_pref_lifetime`
: Optional integer (default is `ipv6_valid_lifetime` value). IPv6 Router Advertisement preferred lifetime in seconds. Must not be greater than `ipv6_valid_lifetime`.

`ipv6_onlink_flag`
: Optional boolean. Defines IPv6 Router Advertisement on-link flag state.

`ipv6_managed_addr_config_flag`
: Optional boolean (default is false). Defines IPv6 Router Advertisement managed address configuration flag state.

`ipv6_other_config_flag`
: Optional boolean (default is false). Defines IPv6 Router Advertisement other configuration flag state.

`ipv6_ra_transmission_interval`
: Optional integer (range -1 to 1800, default is 0). Time in seconds between 2 periodical multicast Router Advertisement transmission, once the initial 3 transmissions have been performed after opening the PDN. The value -1 means that no multicast transmission is done at all (including the 3 initial ones). The value 0 means that periodical transmission is deactivated.

`ipv6_drop_rs`
: Optional boolean (default is false). Defines whether the incoming Router Solicitation messages should be dropped by the MME or not.

`log_get` Get logs.
Message definition:

`min` Optional number (default = 1). Minimum amount of logs to retrieve. Response won't be sent until this limit is reached (Unless timeout occurs).

`max` Optionnal number (default = 4096). Maximum logs sent in a response.

`timeout` Optional number (default = 1). If at least 1 log is available and no more logs have been geenrated for this time, response will be sent.

`rnti` Optional number. If set, send only logs matching rnti.

`ue_id` Optional number. If set, send only logs with matching ue_id.

`layers` Optional Object. Each member name represents a log layer and values must be string representing maximum level. See [log_options], page 7.
If *layers* is not set, all layers level will be set to *debug*, else it will be set to *none*.
Note also the logs is also limited by general log level. See [log_options], page 7.

Response definition:

`logs` Array. List of logs. Each item is a an object with following members:

`data` Array. Each item is a string representing a line of log.

`timestamp`
: Number. Number of seconds since start of session or start of day.

`layer` String. Log layer.

`level` String. Log level: *error*, *warn*, *info* or *debug*.

`dir` Optional string. Log direction: *UL*, *DL*, *FROM* or *TO*.

`ue_id` Optional number. UE_ID.

`cell` Optional number (only for PHY layer logs). Cell ID.

`rnti` Optional number (only for PHY layer logs). RNTI.

`frame` Optional number (only for PHY layer logs). Frame number (Subframe is decimal part).

`channel` Optional string (only for PHY layer logs). Channel name.

`src` String. Server name.

`idx` Integer. Log index.

`discontinuity`
: Optional number. If set, this means some logs have been discarded due to log buffer overflow.

Note that only one request can be sent by client.
If a request is sent before previous one has returned, previous one will be sent without matchine min/max/timeout conditions.

`log_reset`
: Resets logs buffer.

`quit` Terminates ltemme.

`help` Provides list of available messages in *messages* array of strings and events to register in *events* array of strings.

`stats` Provides statistics.
Every time this message is received by server, statistics are reset.

Response definition:

`time` Time in seconds since LTEMME starting.

`cpu` Object. Each member name defines a type and its value cpu load in % of one core.

`instance_id`
: Number. Constant over process lifetime. Changes on process restart.

`counters` Object. List of counters, with following sub members:

`messages` Object. Each member name is the message name and its value is its occurence.
To get list of message, type *cevent help msg* in LTEMME monitor.

`errors` Object. Each member name is the error name and its value is its occurence.
To get list of message, type *cevent help msg* in LTEMME monitor.

`emm_registered_ue_count`
: Integer. Number of UEs in EMM-REGISTERED state.

`s1_connections`
: Array of objects. List of S1AP connection betweens eNBs and MME. Each object contains the following fields:

  `plmn` String. PLMN of the Global eNB ID.

  `enb_id_type`
  : String (macro, home, short_macro or long_macro). Type of identifier of the Global eNB ID.

  `enb_id` Integer. Identifier of the Global eNB ID.

  `ip_addr` String. IP address and port of the eNB.

  `ta_list` Array of objects. List of the Tracking Areas served by the eNB. Each object contains the following fields:

    `plmn` String. PLMN of Tracking Area.

    `tac` Integer. Tracking Area Code.

  `emm_connected_ue_count`
  : Integer. Number of UEs in EMM-CONNECTED state for this S1AP connection.

`register` Register client to message generated by server. Message definition:

`register` String or array of string. List of message to register to. Can be erab˙end, non˙ip˙data, generic˙nas˙transport

`unregister`
: String or array of string. List of message to unregister. Can be erab˙end, non˙ip˙data, generic˙nas˙transport

6.5 LTE messages

`ue_get` Get UE informations.
Message definition:

`imsi` Optional string. If set, retrieve only information from UE with matching IMSI.

`radio_capabilities`
: Optional boolean. If set, provides `radio_capabilities` in response.

Response definition:

`ue_list` Array of current UEs.
Each element has following definition:

`imsi` String. IMSI.

`imeisv` String. IMEISV.

`m_tmsi` String. M_TMSI.

`tac` Integer. Current tracking area code.

`tac_plmn` String. Current tracking area PLMN.

`ue_aggregate_max_bitrate_dl`
: Number. UE aggregate maximum bitrate for downlink.

`ue_aggregate_max_bitrate_ul`
: Number. UE aggregate maximum bitrate for uplink.

`registered`
: Boolean. True if UE is currently registered to network.

`enb_ue_id`
: Integer. eNB UE id. This field would only be present if the UE is still in connected mode.

`mme_ue_id`
: Integer. MME UE id. This field would only be present if the UE is still in connected mode.

`bearers`
: Array. List of connected default bearers. Each bearer has following definition:

  `erab_id`
  : Integer. Bearer ID.

  `ip`
  : String. IPv4 address.

  `ipv6`
  : String. IPv6 address.

  `ul_total_bytes`
  : Number. Total uplink transfered bytes.

  `dl_total_bytes`
  : Number. Total downlink transfered bytes.

  `apn`
  : String. Access point name.

  `dedicated`
  : Array of object. Each object represent a dedicated bearer defined as follow:

    `erab_id`
    : Integer. Bearer ID.

    `ul_total_bytes`
    : Number. Total uplink transfered bytes.

    `dl_total_bytes`
    : Number. Total downlink transfered bytes.

`radio_capabilities`
: GSER string. UE radio access capabilities. Only present if `radio_capabilities` is set to true in request.

`ue_add`
: Add UE to UE database.
Message definition:

  `ue_db`
  : Array. List of UE configuration. See [ue_db], page 20,

`ue_del`
: Remove UE from the UE database and force disconnect if necessary.
Message definition:

  `imsi`
  : String. IMSI of the UE to delete.

`ue_detach`
: Force a detach from network.
Message definition:

  `imsi`
  : String. IMSI of the UE to detach.

`imeisv` Optional string. UE IMEISV, required if `multi_sim` is set to true.

`type` Optional number (default = 2 / re-attach not required). Set NAS detach request type.

`cause` Optional number (default = 3 / illegal UE). Set EMM cause. The value -1 means that the EMM cause IE is not sent in the NAS Detach Request message.

`me_add` Add or update one or several devices in ME database.
Message definition:

`default_status`
Optional enumeration (whitelisted, blacklisted, greylisted). Defines the default status for devices not explicitly defined in the next objects.

`whitelist`
Optional array. It contains a list of IMEI (14 digits) or IMEISV (16 digits) whitelisted.

`blacklist`
Optional array. It contains a list of IMEI (14 digits) or IMEISV (16 digits) blacklisted.

`greylist` Optional array. It contains a list of IMEI (14 digits) or IMEISV (16 digits) greylisted.

`me_del` Remove one or several devices in ME database.
Message definition:

`list` Array of strings. Each entry must be an IMEI (14 digits) or IMEISV (16 digits).

`pws_write`
Start broadcasting Public Warning System message.
Message definition:

`local_id` Number. ID of the message as defined by local_identifier in MME configuration file

`pws_kill` Stop broadcasting Public Warning System message.
Message definition:

`local_id` Number. ID of the message as defined by local_identifier in MME configuration file

`enb` Get list of eNB connections.
Response definition:

`enb_list` Array of object. Each object represents an eNB connection:

`plmn` String. PLMN.

`eNB_ID_type`
String (macro, home, short_macro or long_macro). eNB type.

`eNB_ID` Integer. eNB ID.

`address` String. eNB IP address and port.

`ue_ctx` Number. Number of UE contexts.

`s6` Get information regarding the S6a connection.
Response definition:

- `state` String. S6a connection state (disconnected, connecting, connected or inactive).
- `address` String. HSS address and port.
- `host` Optional string. HSS Diameter host identifier retrieved during Capabilities Exchange procedure.
- `realm` Optional string. HSS Diameter realm identifier retrieved during Capabilities Exchange procedure.

`s6connect`
Force S6a connection establishment.
Message definition:

- `addr` Optional string. If not set, the MME will try to connect to the previously configured address

`s6disconnect`
Force S6a connection release.

`s13` Get information regarding the S13 connection.
Response definition:

- `state` String. S13 connection state (disconnected, connecting, connected or inactive).
- `address` String. EIR address and port.
- `host` Optional string. EIR Diameter host identifier retrieved during Capabilities Exchange procedure.
- `realm` Optional string. EIR Diameter realm identifier retrieved during Capabilities Exchange procedure.

`s13connect`
Force S13 connection establishment.
Message definition:

- `addr` Optional string. If not set, the MME will try to connect to the previously configured address

`s13disconnect`
Force S13 connection release.

`sgs` Get information regarding the SGs connection.
Response definition:

- `state` String. SGs connection state (disconnected, connecting, connected or inactive).
- `address` String. MSC/VLR address and port.

`sgsconnect`
Force SGs connection establishment.
Message definition:

- `addr` Optional string. If not set, the MME will try to connect to the previously configured address

`sgsdisconnect`
: Force SGs connection release.

`ue_activate_dedicated_bearer`
: Trigger a network initiated dedicated EPS bearer activation.
Message definition:

  `imsi` String. UE IMSI.

  `imeisv` Optional string. UE IMEISV, required if `multi_sim` is set to true.

  `apn` String. APN of the default EPS bearer associated to the dedicated one.

  `qci` Integer (range 1 to 255). QoS Class Identifier of the E-RAB.

  `priority_level`
  : Optional integer (1 to 15, default 15). Priority level.

  `pre_emption_capability`
  : Optional enumeration (`shall_not_trigger_pre_emption` or `may_trigger_pre_emption`, default `shall_not_trigger_pre_emption`).

  `pre_emption_vulnerability`
  : Optional enumeration (`not_pre_emptable` or `pre_emptable`, default `not_pre_emptable`).

  `filters` Array. See [TFT], page 18.

  `gbr` Optional object. See [GBR], page 18.

  `transaction_identifier`
  : Optional integer (range 0 to 127). If present, the transaction identifier IE is put in the EPS bearer activation message.

  `llc_sapi` Optional integer (range 0 to 15). If present, the LLC service access point identifier IE is put in the EPS bearer activation message.

  `radio_priority`
  : Optional integer (range 0 to 7). If present, the radio priority IE is put in the EPS bearer activation message.

  `packet_flow_identifier`
  : Optional integer (range 0 to 127). If present, the packet flow identifier IE is put in the EPS bearer activation message.

  `sm_qos` Optional string. If present, the quality of service IE is put in the EPS bearer activation message. The string must contain the hexadecimal representation of the IE without its IEI and length.

  Response definition:

  `erab_id` Integer. Allocated ERAB identity for this dedicated EPS bearer.

`ue_modify_bearer`
: Trigger a network initiated EPS bearer modification.
Message definition:

  `imsi` String. UE IMSI.

  `imeisv` Optional string. UE IMEISV, required if `multi_sim` is set to true.

  `erab_id` Integer. ERAB identity of the bearer to be modified.

`qos` Optional objet. If present a QoS modification is done. It should contain the following objects:

`qci` Integer (range 1 to 255). QoS Class Identifier of the E-RAB.

`priority_level`
Optional integer (1 to 15, default 15). Priority level.

`pre_emption_capability`
Optional enumeration (`shall_not_trigger_pre_emption` or `may_trigger_pre_emption`, default `shall_not_trigger_pre_emption`).

`pre_emption_vulnerability`
Optional enumeration (`not_pre_emptable` or `pre_emptable`, default `not_pre_emptable`).

`gbr` Optional object. See [GBR], page 18.

`filters` Array. Contains the new TFT after modification. See [TFT], page 18.

`llc_sapi` Optional integer (range 0 to 15). If present, the LLC service access point identifier IE is put in the EPS bearer activation message.

`radio_priority`
Optional integer (range 0 to 7). If present, the radio priority IE is put in the EPS bearer activation message.

`packet_flow_identifier`
Optional integer (range 0 to 127). If present, the packet flow identifier IE is put in the EPS bearer activation message.

`sm_qos` Optional string. If present, the quality of service IE is put in the EPS bearer activation message. The string must contain the hexadecimal representation of the IE without its IEI and length.

Response definition:

`erab_id` Integer. ERAB identity of the EPS bearer.

`ue_deactivate_bearer`
Trigger a network initiated default or dedicated EPS bearer deactivation.
Message definition:

`imsi` String. UE IMSI.

`imeisv` Optional string. UE IMEISV, required if `multi_sim` is set to true.

`erab_id` Integer. ERAB identity of the bearer to be released.

`esm_cause`
Optional integer (default = 36). ESM cause for the message.

`non_ip_data`
Send data over a non IP PDN.
Message definition:

`imsi` String. UE IMSI.

`imeisv` Optional string. UE IMEISV, required if `multi_sim` is set to true.

`erab_id` Integer. ERAB identity of the non IP default bearer.

`data` String. ASCII representation of the data hexadecimal dump.

`generic_nas_transport`
: Send a downlink generic NAS transport message.
Message definition:

  `imsi` String. UE IMSI.

  `imeisv` Optional string. UE IMEISV, required if `multi_sim` is set to true.

  `type` Integer (range: 0 to 255). Generic message container type information element.

  `payload` String. ASCII representation of the generic message container hexadecimal dump.

  `add_info` Optional string. ASCII representation of the additional information hexadecimal dump.

`reset_ue_pos_stored_info`
: Send a test procedure reset UE positioning stored information message.
Message definition:

  `imsi` String. UE IMSI.

  `imeisv` Optional string. UE IMEISV, required if `multi_sim` is set to true.

  `techno` Integer (range: 0 to 255). UE positioning technology as sepcified in 3GPP 36.509 chapter 6.9.

`mt_cs_paging`
: Trigger a CS paging.
Message definition:

  `imsi` String. UE IMSI.

6.6 LTE events

Following events are sent by MME if they have been registered on WebSocket.

`non_ip_data`
: Generated by data reception over a non IP PDN.

  `imsi` String. UE IMSI.

  `imeisv` Optional string. UE IMEISV, sent if `multi_sim` is set to true.

  `erab_id` Integer. ERAB identity of the non IP default bearer.

  `data` String. ASCII representation of the data hexadecimal dump.

`generic_nas_transport`
: Generated when receiving an uplink generic NAS transport message.
Message definition:

  `imsi` String. UE IMSI.

  `imeisv` Optional string. UE IMEISV, sent if `multi_sim` is set to true.

  `type` Integer. Generic message container type information element.

  `payload` String. ASCII representation of the generic message container hexadecimal dump.

  `add_info` Optional string. ASCII representation of the additional information hexadecimal dump.

6.7 Examples

1. Config
    1. Client sends

```
{
    "message": "config_get",
    "message_id": "foo"
}
```

    2. Server replies

```
{
    "message_id": "foo",
    "message": "config_get",
    "name": "UE",
    "logs": {
        "phy": {
            "level": "error",
            "max_size": 0
        },
        ...
        "rrc": {
            "level": "debug",
            "max_size": 1
        }
    }
}
```

2. Error
    1. Client sends

```
{
    "message": "bar",
    "message_id": "foo"
}
```

    2. Server replies

```
{
    "message_id": "foo",
    "message": "bar",
    "error": "Unknown message: bar"
}
```

7 Command line monitor reference

The following commands are available:

`help` Display the help. Use `help` *`command`* to have a more detailed help about a command.

`log [`*`log_options`*`]`
Display the current log state. If *log_options* are given, change the log options. The syntax is the same as the `log_options` configuration property.

`enb` List the connected eNodeBs.

`ue [`*`reg`*`]` List all the UE contexts (the UEs can be connected or not). If used with parameter *reg*, only registered UEs will be displayed.

`uectx` List all the active S1 UE contexts.

`pws_write` *`local_id`*
Start broadcasting the ETWS/CMAS message identified by *local_id* on all connected eNodeBs.

`pws_kill` *`local_id`*
Stop broadcasting the ETWS/CMAS message identified by *local_id* on all connected eNodeBs.

`quit` Stop the program and exit.

8 Log file format

8.1 NAS layer

When a NAS message is dumped, the format is:

```
time layer - message
```

When a NAS data PDU is dumped (debug level), the format is:

```
time layer dir MME_UE_ID message_type
        long_content
```

`time` Time using the selected format

`layer` Indicate the layer (`[NAS]` here).

`dir` UL (uplink) or DL (downlink).

`MME_UE_ID`
MME S1AP UE identifier (hexadecimal).

`message_type`
NAS message type.

`long_content`
Full content of the NAS message if `nas.max_size > 0`.

8.2 IP layer

When a IP data PDU is dumped (debug level), the format is:

```
time layer dir short_content
        long_content
```

`time` Time using the selected format

`layer` Indicate the layer (`[IP]` here).

`dir` UL (uplink) or DL (downlink).

`short_content`
Single line content (at least the IP protocol and the source and destination address).

`long_content`
Optional hexadecimal dump of the PDU if `ip.max_size > 0`.

8.3 S1AP and GTP-U layers

When a message is dumped, the format is:

```
time layer - message
```

When a data PDU is dumped (debug level), the format is:

```
time layer dir ip_address short_content
        long_content
```

`time` Time using the selected format.

`layer` Indicate the layer (`[S1AP]` or `[GTPU]` here).

`dir` Direction: `TO` or `FROM`.

`ip_address`
source or destination IP address, depending on the `dir` field.

`short_content`
: Single line content.

`long_content`
: - S1AP: full ASN.1 content of the S1AP message if `layer.max_size > 0`.
  - GTPU: hexadecimal dump of the message if `layer.max_size > 0`.

9 FAQ

9.1 Traffic control

I want to generate errors, limit bandwidth, introduce latency...

Easiest and most powerful way is to do this at IP level using the *tc* Linux command.
There are various tutorials on the internet but it is not a piece of cake so here are some common commands to handle simple case.

First, *tc* will operate at Linux interface level, which means that for LTE we will control the *tun0* interface created by MME.
Note that this configuration will be dropped each time you restart the MME so if you want to set it automatically and keep it we recommand to place the commands inside *config/mme-ifup* (See [tun_setup_script], page 9).

- To limit overall bandwidth to 2mbps:

```
tc qdisc add dev tun0 root handle 1:0 htb default 1
tc class add dev tun0 parent 1:0 classid 1:1 htb rate 2000kbit
```

- To simulate 10% packet loss:

```
tc qdisc add dev tun0 root handle 1: netem loss 10%
```

- To change previous packet loss to 20%:

```
tc qdisc change dev tun0 root handle 1: netem loss 10%
```

- To add 100ms latency with more or less 10ms:

```
tc qdisc add dev tun0 root handle 1: netem delay 100ms 10ms
```

- Same as previous but with a normal distribution:

```
tc qdisc add dev tun0 root handle 1: netem delay 100ms 10ms distribution normal
```

tc is very powerful and you may also chain filters (qdisc), apply them on specific traffic...

10 Known limitations

We present here the known limitations of LTEMME:

- A single PLMN is supported.
- No interface with external SGW is implemented.

11 License

Abbreviations

APN	Access Point Name
DL	Downlink
E-RAB	E-UTRAN Radio Access Bearer
E-UTRA	Evolved UMTS Terrestrial Radio Access
E-UTRAN	Evolved UMTS Terrestrial Radio Access Network
EIR	Equipment Identity Register
HSS	Home Subscriber Server
IMEI	International Mobile Equipment Identity
IMSI	International Mobile Subscriber Identity
LTE	Long Term Evolution
MME	Mobility Management Entity
NAS	Non Access Stratum
PCRF	Policy and Charging Enforcement Function
PDN	Packet Data Network
PDU	Protocol Data Unit
PGW	Packet Data Network Gateway
QCI	Quality of Service (QoS) Class Identifier
QoS	Quality of Service
SDU	Service Data Unit
SGW	Serving Gateway
TMSI	Temporary Mobile Subscriber Identity
UE	User Equipment
UL	Uplink
USIM	Universal Subscriber Identity Module
VoLTE	Voice over LTE