

Configuring GPS

This chapter contains the following sections:

- GPS Overview, on page 1
- Configuring LTE GPS, on page 3
- Enabling NMEA Data Streaming, on page 6
- Debug Cellular-Based GPS, on page 8
- NTP Clock Sync with GPS, on page 8
- National Marine Electronics Association (NMEA) IOx Support, on page 11
- Example: Connecting to a Server Hosting a GPS Application, on page 11
- GPS NMEA unique identifier , on page 12

GPS Overview

Satellite navigation is a system of satellites that provide autonomous geo-spatial positioning with global coverage and allow small electronic receivers to determine location (longitude, latitude, and altitude/elevation) using time signals transmitted from satellites. Often times the terms “GNSS” and “GPS” are used interchangeably but there are key differences between the two:

- The United States’ Global Positioning System (GPS) consists of up to 31 medium Earth orbit satellites in six different orbital planes, with the exact number of satellites varying as older satellites are retired and replaced. Operational since 1978 and globally available since 1994, GPS is currently the world’s most utilized satellite navigation system. A GPS receiver can only use signals from the 31 satellites in the Global Positioning System, and if too many of these signals are blocked, the receiver becomes useless until it can find a signal again.

- GNSS stands for Global Navigation Satellite System, and is the standard generic term for satellite navigation systems that provide autonomous geo-spatial positioning with global coverage. This term includes GPS, GLONASS, Galileo, Beidou and other regional systems. GNSS is a term used worldwide The advantage to having access to multiple satellites is accuracy, redundancy and availability at all times. Though satellite systems don’t often fail, if one fails GNSS receivers can pick up signals from other systems. Also, if line of sight is obstructed, having access to multiple satellites is also a benefit. Common GNSS Systems are GPS, GLONASS, Galileo, Beidou and other regional systems. A GNSS receiver can use signals from any positioning satellite, not just the ones in the GPS system. This means that even if all of the GPS signals are blocked, it can pick up signals from any of the other satellite systems worldwide. This flexibility makes GNSS receivers much more accurate and reliable than GPS technology alone. With GNSS, you can be confident that you are getting the best result possible, wherever and whenever you need it.

Note A GNSS antenna is required for proper reception of GNSS constellation signal. Most Cisco antennas are GPS only, your antenna SKU must explicitly state it supports GNSS.

Note For the purposes of this document, we will be referring only to Cellular Modem based GPS.

Cellular GPS Use Cases

The following are descriptions of some of the use cases for GPS:

- Geo-Fencing — Geo-fencing is the concept of controlling a device's operation based on where it is located. One use may be to only allow a device to operate if it is located in an area it is authorized to be. For example, if a router inside a store, kiosk or ATM was taken to a location other than where it should be, the router could disable itself. Or if the device is programmed to send (or polled) for its location, and the cloud/host application finds it outside the authorized area, the application may stop allowing communications to the devices behind the router.
- Asset Tracking — Asset tracking is the concept of knowing where equipment or devices are currently, and their condition/operation. The condition/operation can be gleaned by cloud/host applications, or by network management tools. The location of the specific device/equipment however can be important to an organization's operations. For example, a transportation enterprise may need to know the location of each of its trucks, train cars and ships in order to coordinate the on-time shipment and delivery of cargo. This may allow for the best selection of transport means and specific vehicles for a particular expedited shipment.
- Time Synchronization — Accurate time on a router can be important, for having exact time stamps of logs, or for accurate execution of tasks via automated/scheduled router functions. There may be times that the router requires synchronization of its internal clock with a highly accurate time source. Inability to sync the time may occur with deployments in vehicles or movable installations that are out of signal range for long periods, or where the deployment used a connection to an isolated private network with no NTP server.
- GPS streaming to Apps and IOx — When a modem has GPS enabled, the NMEA stream can be forwarded to IOx from the ngiolite module. This is accomplished by create a tunnel between Linux and IOx and then forward all NMEA messages over the tunnel to IOx.

Cellular GPS Parameters

Refer to the following table:

Parameters	Cellular Modem Based GPS
Type	Cellular Modem based GPS
PIDs Supported	See the Supported Modem Technology section for modules that support GPS.
Configuration Modes	Standalone mode

Parameters	Cellular Modem Based GPS
Number of satellites needed for co-ordinates	Standalone mode – 4
Satellites Supported in show command	Co-ordinates seen in show commands output is based only on GPS.
Initial Calibration Required	No
Co-ordinates in Absence of Satellite	No Co-ordinates will be acquired, and it stays in acquiring status.
Device Name of Controller to use for Configuration	controller cellular <slot>
CLI to enable feature	**lte gps enable** **lte gps mode standalone** **Note** A modem power cycle is required after enabling the configuration.
CLI to configure nmea	**lte gps nmea** **Note** NMEA must be enabled for the router to receive the GPS coordinates.
CLI to configure nmea udp socket	**lte gps nmea ip udp** <*source_ip*> <*destination_ip*> <*destination_port*>
CLI to verify configuration under show running-config	**show run | sec controller cellular**<*slot*>
Show commands to verify gps output	**show cellular** <*slot*> **gps** **show controller cellular** <*slot*> **| inc GPS**
Access to GPS nmea traffic on IOx side	Supported
Debug Command	**debug cellular**<*slot*> **messages gps** **debug cellular**<*slot*> **messages nmea**
Yang Model Support	Yes

Configuring LTE GPS

Follow these steps to configure the LTE cellular modem for GPS in Standalone mode.

Note

- Before configuring GPS on any modem, install the latest available modem firmware and OEM PRI.

  For example, the minimum required versions for the EM9293 modem are:

  - Modem firmware 02.17.08.00 or later.
  - OEM PRI EM9293_001.006.

  To download, see Release Notes for Cisco Catalyst IR1101, IR1800, IR8140, and IR8340 Routers.

- Use the **show cellular** *slot* **hardware** command before and after upgrading, to verify the installed versions.

Note For a 4G LTE-Advanced, the *slot* argument identifies the router slot, module slot, and the port, and is separated by slashes (0/4/0).

Note Starting with IOS XE 17.9.1, GPS will be enabled by default.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Router# configure terminal`	Enters the configuration mode.
Step 2	**controller cellular** *slot* **Example:** `Router(config)# controller cellular 0/4/0`	Enters the controller cellular configuration mode.
Step 3	**lte gps enable** **Example:** `Router(config-controller)# lte gps enable`	(Optional) GPS is enabled by default. Use this command to enable the GPS feature if GPS has been disabled for any reason.
Step 4	**lte gps mode standalone** **Example:** `Router(config-controller)# lte gps mode standalone`	Enables the standalone GPS mode.
Step 5	**lte gps nmea** **Example:** `Router(config-controller)# lte gps nmea`	Enables NMEA streaming to Cisco IOS XE. Both GPS and NMEA must be enabled for the router to obtain and display GPS coordinates. As without this command, the GPS status remains `acquiring`, and no coordinates appear in the **show cellular** *slot* **gps** command output.

	Command or Action	Purpose
Step 6	**end** **Example:** `Router(config-controller)# end`	Exits the controller configuration mode and returns to the privileged EXEC mode.
Step 7	**test cellular** *slot* **modem-power-cycle** **Example:** `Router# test cellular 0/4/0 modem-power-cycle`	GPS can take effect only after modem power cycle.
Step 8	**show cellular** *slot* **gps** **Example:** `Router# show cellular 0/4/0 gps` ` GPS Feature = enabled` `GPS Mode Configured = standalone` `GPS Port Selected = Dedicated GPS port` `GPS Status = GPS coordinates acquired` `Last Location Fix Error = Offline [0x0]` `Latitude = 37 Deg 25 Min 4.8915 Sec North` `Longitude = 121 Deg 55 Min 8.5627 Sec West` `Timestamp (GMT) = Wed Nov 7 21:54:18 2018` `Fix type index = 0, Height = 8 m` `Satellite Info` `----------------` `Satellite #1, elevation 45, azimuth 303, SNR 20 *` `Satellite #3, elevation 15, azimuth 296, SNR 21` `Satellite #8, elevation 9, azimuth 227, SNR 27 *` `Satellite #11, elevation 41, azimuth 270, SNR 27 *` `Satellite #18, elevation 64, azimuth 258, SNR 29 *` `Satellite #22, elevation 35, azimuth 303, SNR 22 *` `Satellite #31, elevation 51, azimuth 140, SNR 24 *` `Satellite #32, elevation 46, azimuth 43, SNR 22 *` `Satellite #10, elevation 25, azimuth 97, SNR 0` `Satellite #14, elevation 68, azimuth 26, SNR 0` `!!... truncated ....!!` `Router#`	Displays a summary of the following GPS data: • GPS state information (GPS disabled, GPS acquiring, GPS enabled) • GPS mode configured (standalone) • GPS location and timestamp information • GPS satellite information • GPS feature (enabled or disabled) • GPS port selected (Dedicated GPS and GPS port with voltage-no-bias)
Step 9	**show cellular** *slot* **gps detail** **Example:** `Router# show cellular 0/4/0 gps detail` `GPS Feature = enabled` `GPS Mode Configured = standalone` `GPS Port Selected = Dedicated GPS port` `GPS Status = GPS coordinates acquired` `Last Location Fix Error = Offline [0x0]` `Latitude = 37 Deg 25 Min 4.9282 Sec North` `Longitude = 121 Deg 55 Min 8.5209 Sec West` `Timestamp (GMT) = Wed Nov 7 21:53:52 2018` `Fix type index = 0, Height = 7 m` `HDOP = 1.5, GPS Mode Used = standalone` `Satellite Info`	Displays detailed GPS data.

	Command or Action	Purpose
	``---------- Satellite #8, elevation 9, azimuth 227, SNR 31 * Satellite #11, elevation 41, azimuth 270, SNR 32 * Satellite #18, elevation 64, azimuth 258, SNR 33 * Satellite #22, elevation 35, azimuth 303, SNR 26 * Satellite #31, elevation 51, azimuth 140, SNR 27 * Satellite #32, elevation 46, azimuth 43, SNR 22 Satellite #1, elevation 45, azimuth 303, SNR 0 Satellite #3, elevation 14, azimuth 296, SNR 0 !!!...truncated!!! Router#```	

Enabling NMEA Data Streaming

GPS NMEA data streaming to external NMEA 2.0-compliant GPS plotter applications can be enabled on Cisco 4G LTE Advanced.

Note

- NMEA configuration is mandatory for the router to receive the GPS coordinates. If NMEA is not configured, the GPS status remains in `GPS acquiring` state, and IOS XE does not report the latitude or longitude.
- Enable NMEA for Cisco IOS XE to obtain GPS coordinates, even when you are not streaming data to an external NMEA 2.0 plotter or IOx application.
- NMEA configuration must be enabled whenever NMEA data streaming is required.
- Before configuring NMEA data streaming, ensure that LTE GPS is configured and enabled.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Router# configure terminal`	Enters the configuration mode.
Step 2	**controller cellular** *slot* **Example:** `Router(config)# controller cellular 0/4/0`	Enters the controller cellular configuration mode.
Step 3	**lte gps nmea {ip | udp** [*source address*][*destination address*][*destination port*] } **Example:**	Enables NMEA. Cisco 4G LTE Advanced supports only IP NMEA. Therefore, the IP interface and serial interface options are unavailable.

	Command or Action	Purpose
	`Router(config-controller)# lte gps nmea ip` or `Router(config-controller)# lte gps nmea`	
Step 4	**end** **Example:** `Router(config-controller)# end`	Exits the controller configuration mode and returns to the privileged EXEC mode.
Step 5	**test cellular** *slot* **modem-power-cycle** **Example:** `Router# test cellular 0/4/0 modem-power-cycle`	GPS can take effect only after modem power cycle.
Step 6	**show cellular** *slot* **gps** **Example:** `Router# show cellular 0/4/0 gps` `GPS Feature = enabled` `GPS Mode Configured = standalone` `GPS Port Selected = Dedicated GPS port` `GPS Status = GPS coordinates acquired` `Last Location Fix Error = Offline [0x0]` `Latitude = 37 Deg 25 Min 4.8915 Sec North` `Longitude = 121 Deg 55 Min 8.5627 Sec West` `Timestamp (GMT) = Wed Nov 7 21:54:18 2018` `Fix type index = 0, Height = 8 m` `Satellite Info` `----------------` `Satellite #1, elevation 45, azimuth 303, SNR 20 *` `Satellite #3, elevation 15, azimuth 296, SNR 21` `Satellite #8, elevation 9, azimuth 227, SNR 27 *` `Satellite #11, elevation 41, azimuth 270, SNR 27 *` `Satellite #18, elevation 64, azimuth 258, SNR 29 *` `Satellite #22, elevation 35, azimuth 303, SNR 22 *` `Satellite #31, elevation 51, azimuth 140, SNR 24 *` `Satellite #32, elevation 46, azimuth 43, SNR 22 *` `Satellite #10, elevation 25, azimuth 97, SNR 0` `Satellite #14, elevation 68, azimuth 26, SNR 0` `!!... truncated ....!!` `Router#`	Displays a summary of the following GPS data: • GPS state information (GPS disabled, GPS acquiring, GPS enabled) • GPS mode configured (standalone) • GPS location and timestamp information • GPS satellite information • GPS feature (enabled or disabled) • GPS port selected (Dedicated GPS and GPS port with voltage-no-bias)
Step 7	**show cellular** *slot* **gps detail** **Example:** `Router# show cellular 0/4/0 gps detail` `GPS Feature = enabled` `GPS Mode Configured = standalone` `GPS Port Selected = Dedicated GPS port` `GPS Status = GPS coordinates acquired` `Last Location Fix Error = Offline [0x0]` `Latitude = 37 Deg 25 Min 4.9282 Sec North` `Longitude = 121 Deg 55 Min 8.5209 Sec West`	Displays detailed GPS data.

	Command or Action	Purpose
	Timestamp (GMT) = Wed Nov 7 21:53:52 2018 Fix type index = 0, Height = 7 m HDOP = 1.5, GPS Mode Used = standalone Satellite Info ---------- Satellite #8, elevation 9, azimuth 227, SNR 31 * Satellite #11, elevation 41, azimuth 270, SNR 32 * Satellite #18, elevation 64, azimuth 258, SNR 33 * Satellite #22, elevation 35, azimuth 303, SNR 26 * Satellite #31, elevation 51, azimuth 140, SNR 27 * Satellite #32, elevation 46, azimuth 43, SNR 22 Satellite #1, elevation 45, azimuth 303, SNR 0 Satellite #3, elevation 14, azimuth 296, SNR 0 !!!...truncated!!! Router#	

Debug Cellular-Based GPS

Use the following commands to debug Cellular-Based GPS issues.

- **debug cellular** <*slot*> **messages gps**
- **debug cellular** <*slot*> **messages nmea**

NTP Clock Sync with GPS

This feature enables GPS time as the time source for NTP from LTE modems (except LM960) on the router.

GPS time will act as a stratum 0 source and IOS NTP server will act as a stratum 1 device, which in turn provides clock information to its NTP clients (stratum 2 and 3).

In order to use the NTP clock sync feature, you must first enable cellular modem GPS. See the command examples in the following section.

Command Line Interface

To enable LTE GPS time to NTP feature, the following CLI is available and needs to be configured. If the CLI is not configured, LTE time acquired by the GPS over the LTE modem will not be the time source.

The following example uses the CLI to select the interface/modem as time source:



Note Make sure the LTE modems are present on the slot you are trying to configure.

```
Route(config)# ntp refclock gps interface <Cellular Interface>

Router(config)# ntp refclock gps interface ?
  Cellular     Cellular WAN interface
```

Once the above CLIs are configured, all NTP CLI show commands shall reflect the time from the LTE modems.

Example Output

Using Cellular Interface 0/1/0

Make sure that the interface is configured for GPS.

```
Router#show cellular 0/1/0 gps
GPS Feature =  enabled
GPS Mode Configured =  standalone
GPS Port Selected =  Dedicated GPS port
GPS Status =  GPS coordinates acquired
Last Location Fix Error =  Offline [0x0]
Latitude =  37 Deg 25 Min 5.1159 Sec North
Longitude =  121 Deg 55 Min 8.4338 Sec West
Timestamp (GMT) =  Thu Apr 22 02:03:07 2021

Fix type index =  0, Height =  28 m
Satellite Info
----------------
Satellite #5, elevation 39, azimuth 49, SNR 29 *
Satellite #10, elevation 1, azimuth 212, SNR 22
Satellite #12, elevation 2, azimuth 164, SNR 22
Satellite #15, elevation 23, azimuth 130, SNR 25
Satellite #16, elevation 7, azimuth 323, SNR 26
Satellite #18, elevation 51, azimuth 286, SNR 32 *
Satellite #20, elevation 73, azimuth 188, SNR 29 *
Satellite #23, elevation 30, azimuth 206, SNR 28 *
Satellite #25, elevation 29, azimuth 189, SNR 25 *
Satellite #26, elevation 25, azimuth 305, SNR 29 *
Satellite #29, elevation 75, azimuth 53, SNR 31 *
Satellite #2, elevation 0, azimuth 74, SNR 0
Satellite #13, elevation 15, azimuth 94, SNR 0
Satellite #78, elevation 21, azimuth 309, SNR 31 *
Satellite #77, elevation 26, azimuth 250, SNR 14
Satellite #76, elevation 4, azimuth 203, SNR 24
Satellite #65, elevation 19, azimuth 170, SNR 24
Satellite #88, elevation 19, azimuth 299, SNR 29 *
Satellite #87, elevation 47, azimuth 337, SNR 30 *
Satellite #71, elevation 38, azimuth 50, SNR 28 *
Satellite #72, elevation 49, azimuth 125, SNR 24
Satellite #70, elevation 0, azimuth 0, SNR 0
Satellite #86, elevation 31, azimuth 70, SNR 0
Satellite #1, elevation 13, azimuth 185, SNR 0
Satellite #7, elevation 19, azimuth 53, SNR 0
Satellite #19, elevation 16, azimuth 157, SNR 0
Satellite #24, elevation 28, azimuth 300, SNR 0
Satellite #31, elevation 40, azimuth 239, SNR 0
Satellite #33, elevation 64, azimuth 12, SNR 0  **
```

Configure the interface to use GPS as a time source.

```
Router#config term
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#ntp refclock gps interface Cellular0/1/0
Router(config)#end
```

Verify the action.

```
Router#show run | sec ntp
ntp refclock gps interface Cellular0/1/0

Router#show ntp associations

  address         ref clock       st   when   poll reach  delay  offset   disp
 ~127.127.5.1     .GPS.            0      -     16     0  0.000   0.000 16000.
 * sys.peer, # selected, + candidate, - outlyer, x falseticker, ~ configured

Router#show ntp status
Clock is synchronized, stratum 1, reference is .GPS.
nominal freq is 250.0000 Hz, actual freq is 249.8750 Hz, precision is 2**10
ntp uptime is 393200 (1/100 of seconds), resolution is 4016
reference time is E42B582E.00022760 (02:04:30.000 UTC Thu Apr 22 2021)
clock offset is -0.0328 msec, root delay is 0.00 msec
root dispersion is 7939.08 msec, peer dispersion is 7937.98 msec
loopfilter state is 'CTRL' (Normal Controlled Loop), drift is 0.000499999 s/s
system poll interval is 64, last update was 6 sec ago.
```

Yang Model Support

YANG model support is available for the CLI below:

Note Make sure the LTE modems are present on the slot you are trying to configure.

Use the following command to configure Cellular interface timestamp as the NTP source.

ntp refclock gps interface *<cellular 0/x/0>*

Use the following command to disable the Cellular interface timestamp as the NTP source.

[no] ntp refclock gps interface *<cellular 0/x/0>*

Note A model file, Cisco-IOS-XE-ntp.yang, is available for the configuration model.

Yang Model Examples

Cellular Slot 1

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="101">
  <edit-config>
    <target>
      <running/>
    </target>
    <config>
      <native xmlns="http://cisco.com/ns/yang/Cisco-IOS-XE-native">
        <ntp>
          <refclock xmlns="http://cisco.com/ns/yang/Cisco-IOS-XE-ntp">
            <gps>
              <interface>Cellular0/1/0</interface>
            </gps>
          </refclock>
        </ntp>
      </native>
```

```
    </config>
  </edit-config>
</rpc>
```

Get the configuration

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="101">
  <get-config>
    <source>
      <running/>
    </source>
    <filter>
      <native xmlns="http://cisco.com/ns/yang/Cisco-IOS-XE-native">
        <ntp>
          <refclock xmlns="http://cisco.com/ns/yang/Cisco-IOS-XE-ntp"
xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0" nc:operation="delete"/>
        </ntp>
      </native>
    </filter>
  </get-config>
</rpc>
```

Delete the configuration

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="101">
  <edit-config>
    <target>
      <running/>
    </target>
    <config>
      <native xmlns="http://cisco.com/ns/yang/Cisco-IOS-XE-native">
        <ntp>
          <refclock xmlns="http://cisco.com/ns/yang/Cisco-IOS-XE-ntp"
xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0" nc:operation="delete"/>
        </ntp>
      </native>
    </config>
  </edit-config>
</rpc>
```

National Marine Electronics Association (NMEA) IOx Support

From linux or the IOx container, the following tty is available for NMEA traffic:

- /dev/ttyTun9
- /dev/ttyS2

Example: Connecting to a Server Hosting a GPS Application

You can feed the NMEA data to a remote server that hosts the GPS application. The server can be connected to the router either directly using an Ethernet cable or through a LAN or WAN network. If the application supports serial port, run a serial port emulation program to create a virtual serial port over the LAN or WAN connection.

Note Microsoft Streets & Trips is a licensed software that you can download from the Microsoft website.

To connect a Cisco 4G LTE-Advanced through IP to a PC running Microsoft Streets & Trips, perform the following steps:

1. Connect the PC to the router using an Ethernet cable.
2. Ensure that the PC and router can ping.
3. Launch the serial port redirector on the PC.
4. Create a virtual serial port that connects to the NMEA port on the router.
5. Launch **Microsoft Streets & Trips** on your PC.
6. Select the GPS Menu.
7. Click Start Tracking.
8. If you have acquired a location fix from the **show cellular 0/4/0 gps** command output on the router, the current location is plotted on the graph, and a reddish brown dotted cursor with a circle around it is seen on the map.

Note If you have not acquired a location fix, the Microsoft application times out and disconnects.

GPS NMEA unique identifier

Starting with Cisco IOS-XE Release 26.1.1, Cisco routers can send a Unique Identifier (UID) to identify the source device when streaming NMEA GPS sentences over UDP on the Cisco IR1101 and Cisco IR1835 routers. UID streaming is supported from the Dead Reckoning (DR) module and from cellular modems that support GNSS. The UID uniquely identifies each router in GPS data processing systems.

Table 1: Feature History Table

Feature name	Release information	Feature description
Send a UID to identify the unique source when sending GPS coordinates data	Release 26.1.1	This feature allows you to configure Cisco routers to send a Unique Identifier (UID) while streaming NMEA GPS sentences over UDP. This UID streaming is supported by both the Dead Reckoning (DR) module and cellular modems with GNSS capabilities, enabling each router to be uniquely identified in GPS data processing systems.

Cisco proprietary NMEA sentence format

When you configure a UID, a Cisco proprietary NMEA sentence is sent alongside the standard NMEA sentence in the same UDP packet. The format is:

```
$PCSCU,<UID>,<Source>,<Router Time>,<NMEA Sentence Time>,<NMEA Sentence Checksum>,,,,,,*95
```

- `CSC` indicates Cisco
- `U` indicates this sentence provides a unique identifier
- `<UID>` is the serial number, hostname, or custom ID
- `<Source>` indicates the origin of the NMEA stream:
  - `0`: cellular modem0
  - `1`: cellular modem1
  - `2`: DR module

For example, for cellular modem1, if you use the serial number as the UID, the Cisco proprietary NMEA sentence appears as:

```
$PCSCU,FCW2447P0EU,1,213431.100,213431.000,75,,,,,,*95
```

In this NMEA sentence

- `CSC` indicates Cisco
- `U` indicates that this sentence provides a unique identifier
- `FCW2447P0EU` indicates serial number as the UID
- `1` indicates source as cellular modem1
- `1213431.100` indicates the router time
- `213431.000` indicates the NMEA sentence time
- `75` indicates the NMEA sentence checksum

NMEA sentence selection

By default, all NMEA sentences from the modem or DR module are sent to the configured UDP destination. To reduce UDP traffic, you can select specific NMEA sentence types to send using a hex bitmask. The table presents each NMEA sentence type and its corresponding HEX value.

Table 2: NMEA sentence type and HEX value

HEX value	NMEA sentence type
0x01	GGA
0x02	GSA
0x04	GSV
0x08	RMC

HEX value	NMEA sentence type
0x10	VTG
0x20	GNS

Configure cellular modem GPS NMEA unique identifier

Follow these steps to configure cellular modem GPS NMEA UID.

Procedure

Step 1 Use the **configure terminal** command to enter the configuration mode.

Example:

```
Router# configure terminal
```

Step 2 Use the **controller cellular** *slot* command to enter the controller cellular configuration mode.

Example:

```
Router(config)# controller cellular 0/x/0
```

Step 3 Use the **lte gps nmea uid** {**custom** *unique-identifier* **| hostname | serial-number**} command to enable the GPS NMEA UID.

UID can be any of these:

- **custom**: user-configured string of upto 50 characters.
- **hostname**: host name of the router.
- **serial-number**: serial number of the router.

When using **serial-number** or **hostname**, the UID is automatically extracted and added to the proprietary NMEA sentence. This allows uniform CLI configuration across routers.

When using a **custom ID**, each router must be configured with a *unique-identifier* value to ensure distinct identification.

Example:

```
Router(config-controller)# lte gps nmea uid custom 1428
```

Example:

```
Router(config-controller)# lte gps nmea uid hostname
```

Example:

```
Router(config-controller)# lte gps nmea uid serial-number
```

Step 4 Use the **lte gps nmea ip udp** *source-address destination-address destination-port* **stream** *stream-id* command to specify the source and destination IP address.

Example:

```
Router(config-controller)# lte gps nmea ip udp 10.195.79.179 171.70.55.77 14013 stream 1
```

Step 5 Use the **lte gps nmea filter** *hex-value* command to select the sentence types.

hex-value: HEX bitmask for NMEA message filtering (0x01:GGA, 0x02:GSA, 0x04:GSV, 0x08:RMC, 0x010:VTG, 0x20:GNS)

This example shows the HEX value if you want to send sentence type as only RMC (0x08) and GNS (0x20):

Example:

```
Router(config-controller)# lte gps nmea filter 28
```

This example shows the HEX value if you want to send sentence type as only GGA (0x01) and GSA (0x02):

Example:

```
Router(config-controller)# lte gps nmea filter 03
```

Step 6 Use the **end** command to exit the controller configuration mode and return to the privileged EXEC mode.

Example:

```
Router(config-controller)# end
```