

# FW 1.2.0 LTP-16N(T). OLT configuration and monitoring via SNMP

Station optical terminals LTP-16N(T)

Application to the user manual

OLT configuration and monitoring via SNMP

Firmware version 1.2.0 (28.05.2021)

- 1 [Operation with ONT](#)
  - 1.1 [Configuration](#)
    - 1.1.1 [Addition](#)
    - 1.1.2 [Editing](#)
      - 1.1.2.1 [ONT General Parameters](#)
      - 1.1.2.2 [Cross-Connect and DBA profiles](#)
    - 1.1.3 [ONT Deletion](#)
    - 1.1.4 [ONT replacement](#)
    - 1.1.5 [Reconfiguration](#)
    - 1.1.6 [Reboot](#)
    - 1.1.7 [Reset to the default settings](#)
  - 1.2 [Requests](#)
    - 1.2.1 [Common ONT state](#)
- 2 [OLT configuration:](#)
  - 2.1 [Applying and saving configuration](#)
    - 2.1.1 [Configuration commit](#)
    - 2.1.2 [Saving the configuration to non-volatile memory](#)
  - 2.2 [VLAN Configuration](#)
    - 2.2.1 [Adding VLAN](#)
    - 2.2.2 [VLAN editing](#)
    - 2.2.3 [VLAN Deletion](#)
    - 2.2.4 [Requesting the VLAN list and configuration of the specific VLAN](#)
  - 2.3 [IGMP configuration](#)
    - 2.3.1 [Global settings for enabling IGMP Snooping](#)
    - 2.3.2 [IGMP Snooping and VLAN Querier configuration](#)
  - 2.4 [ONT profiles configuration](#)
    - 2.4.1 [Cross-connect](#)
      - 2.4.1.1 [Addition](#)
      - 2.4.1.2 [Editing](#)
      - 2.4.1.3 [Deletion](#)
      - 2.4.1.4 [Profile list request](#)
    - 2.4.2 [DBA](#)
      - 2.4.2.1 [Addition](#)
      - 2.4.2.2 [Editing](#)
      - 2.4.2.3 [Deletion](#)
      - 2.4.2.4 [Profile list request](#)
    - 2.4.3 [Ports](#)
      - 2.4.3.1 [Addition](#)
      - 2.4.3.2 [Editing](#)
      - 2.4.3.3 [Deletion](#)
      - 2.4.3.4 [Profile list request](#)
    - 2.4.4 [Management](#)
      - 2.4.4.1 [Addition](#)
      - 2.4.4.2 [Editing](#)
      - 2.4.4.3 [Deletion](#)
      - 2.4.4.4 [Profile list request](#)
  - 2.5 [Interface configuration](#)
    - 2.5.1 [GPON-ports](#)
    - 2.5.2 [Front-ports](#)
- 3 [OLT monitoring](#)
  - 3.1 [LTP general information](#)
- 4 [The list of changes](#)

## Notes and warnings



Notes contain important information, tips or recommendations on device operation and setup.



Warnings are used to inform the user about harmful situations for the device and the user alike, which could cause malfunction or data loss.

## Operation with ONT



Matches between parameter names and digital OIDs are described in the MIB files.

If the command contains the ONT serial number, it is specified in the format `AAAAAAAAXXXXXXXX`, where `AAAA` — 4 latin letters, `XXXXXXXX` — 8 HEX-characters.

#### Examples:

Serial number
ELTX24A80012
ELTX6201AAFC

ONT profiles are specified by the profile index for corresponding OID of the ONT configuration. To know profile index using its name, use the following profile tables:

Profile type	Table
Cross-connect	<code>ltp16nONTProfileCrossConnectTable</code>
DBA	<code>ltp16nONTProfileDBATable</code>
Ports	<code>ltp16nONTProfilePortsTable</code>

If the ONT configuration does not allow a profile to be specified, the number 0 is sent instead of the profile index to set such a value.

## Configuration

ONT is configured by the following tables:

- `ltp16nONTConfigTable` — general parameters;
- `ltp16nONTServiceTable` — Cross-Connect and DBA profiles.

## Addition

The following indexes are used in ONT configuration tables: `<gpon_port>`, `<ont_id>`.



When adding an ONT, you must specify an additional parameter from the ONT configuration, such as a serial number or PON password.

*Command format:*

```
snmpset -v2c -c <rw_community> <ipaddr>  
ltp16nONTConfigRowStatus.1.<gpon_port>.<ont_id> i 4  
ltp16nONTConfigSerialNumber.1.<gpon_port>.<ont_id> s <serial>
```

Where:

- `<gpon_port>` — PON port number;
- `<ont_id>` — ONT ID value.

*Example:*

```
snmpset -v2c -c private 192.168.1.2  
ltp16nONTConfigRowStatus.1.3.8 i 4  
ltp16nONTConfigSerialNumber.1.3.8 s ELTX24A80012
```

This command creates ONT 3/8 with serial number ELTX24A80012.

## Editing

### ONT General Parameters

General parameters of ONT are configured by using `ltp16nONTConfigTable`.

*Command format:*

```
snmpset -v2c -c <rw_community> <ipaddr> <parameter_oid_1>.1.<gpon_port>.<ont_id> <par1_type> <par1_value>
<parameter_oid_2>.1.<gpon_port>.<ont_id> <par2_type> <par2_value>
.....
<parameter_oid_N>.1.<gpon_port>.<ont_id> <parN_type> <parN_value>
```

Where:

- <parameter\_oid\_N> — the names of specific MIB parameters;
- <parN\_type> — value type of a parameter;
- <parN\_value> — parameter value.

*Example:*

```
snmpset -v2c -c private 192.168.1.2 ltp16nONTConfigPonPassword.1.3.8 s
"1234567890"
ltp16nONTConfigProfilePortsID.1.3.8 u 1
ltp16nONTConfigBroadcastFilter.1.3.8 i 2
ltp16nONTConfigProfileManagementID.1.3.8 u 1
```

This command sets *password=1234567890*, assigns the *Ports* profile with index 1, assigns the Management profile with index 1 and disables the *Broadcast filter* for ONT 3/8.

## Cross-Connect and DBA profiles

Cross-Connect and DBA profiles are configured by using *ltp16nONTServiceTable*. An additional index is specified — the service number.



Cross-connect and DBA profiles should be specified when creating a service.

*Command format:*

```
snmpset -v2c -c <rw_community> <ipaddr>
ltp16nONTServiceRowStatus.1.<gpon_port>.<ont_id>.<service> i 4
ltp16nONTServiceProfileCrossConnectID.1.<gpon_port>.<ont_id>.<service> u <value>
ltp16nONTServiceProfileDBAID.1.<gpon_port>.<ont_id>.<service> u <value>
```

Where:

- <service> — service sequential number;
- <value> — profile ID, according to ltp16nONTProfileCrossConnectTable, ltp16nONTProfileDBATable.

*Example:*

```
snmpset -v2c -c private 192.168.1.2
ltp16nONTServiceRowStatus.1.3.8.7 i 4
ltp16nONTServiceProfileCrossConnectID.1.3.8.7 u 1
ltp16nONTServiceProfileDBAID.1.3.8.7 u 2
```

This command sets Cross-Connect profile with index 1, and DBA profile with index 2 for service 7 for ONT 3/8.

## ONT Deletion

*Command format:*

```
snmpset -v2c -c <rw_community> <ipaddr>
ltp16nONTConfigRowStatus.1.<gpon_port>.<ont_id> i 6
```

*Example:*

```
snmpset -v2c -c private 192.168.1.2
ltp16nONTConfigRowStatus.1.3.8 i 6
```

This command removes the ONT 3/8 configuration.

## ONT replacement

You can change ONT by using subsequent commands for deleting and creating configurations with new parameters.

## Reconfiguration

*Command format:*

```
snmpset -v2c -c <rw_community> <ipaddr>  
ltp16nONTOperationsReconfigure.1.<gpon_port>.<ont_id> i 1
```

*Example:*

```
snmpset -v2c -c private 192.168.1.2  
ltp16nONTOperationsReconfigure.1.3.8 i 1
```

## Reboot

*Command format:*

```
snmpset -v2c -c <rw_community> <ipaddr>  
ltp16nONTOperationsReboot.1.<gpon_port>.<ont_id> i 1
```

*Example:*

```
snmpset -v2c -c private 192.168.1.2  
ltp16nONTOperationsReboot.1.3.8 i 1
```

## Reset to the default settings

*Command format:*

```
snmpset -v2c -c <rw_community> <ipaddr>  
ltp16nONTOperationsDefault.1.<gpon_port>.<ont_id> i 1
```

*Example:*

```
snmpset -v2c -c private 192.168.1.2  
ltp16nONTOperationsDefault.1.3.8 i 1
```

## Requests

### Common ONT state

The parameters of ONT state can be requested by using *ltp16nONTInfoTable*.

*Command format:*

```
snmpget -v2c -c <ro_community> <ipaddr> <parameter_oid>.1.<gpon_port>.<ont_id>
```

*Example:*

```
snmpget -v2c -c public 192.168.1.2  
ltp16nONTInfoState.1.3.8  
ltp16nONTInfoRSSI.1.3.8
```

The command reaquests the status and RSSI for the ONT 3/8.

## OLT configuration:

### Applying and saving configuration

Configuration actions are performed using *ltp16nConfigOperations*.

### Configuration commit

To write the configuration to running-config, execute Commit.

*Command format:*

**snmpset -v2c -c <rw\_community> -t 20 <ipaddr> ltp16nConfigOperationsCommit.0 i 1**

*Example:*

```
snmpset -v2c -c private -t 20 192.168.1.2 ltp16nConfigOperationsCommit.0 i 1
```

## Saving the configuration to non-volatile memory

To write the configuration to the non-volatile memory, execute Save.

*Command format:*

**snmpset -v2c -c <rw\_community> -t 20 <ipaddr> ltp16nConfigOperationsSave.0 i 1**

*Example:*

```
snmpset -v2c -c private -t 20 192.168.1.2 ltp16nConfigOperationsSave.0 i 1
```

## VLAN Configuration

VLAN configuration is done using the tables:

- *ltp16nVLANSettingsTable* — general VLAN parameters;
- *ltp16nVLANSettings/GMPTTable* — IGMP parameters;
- *ltp16nVLANSettings/GMPSnoopingTable* — IGMP Snooping parameters;
- *ltp16nVLANSettings/GMPSnoopingStaticGroupTable* — static group management.

## Adding VLAN

*Command format:*

**snmpset -v2c -c <rw\_community> <ipaddr> ltp16nVLANSettingsRowStatus.1.<vlan\_id> i 4**

*Example:*

```
snmpset -v2c -c private 192.168.1.2 ltp16nVLANSettingsRowStatus.1.156 i 4
```

The command creates VLAN 156.

## VLAN editing

It is possible to change the name and IGMP settings for any VLAN via SNMP.

*Command format:*

**snmpset -v2c -c <rw\_community> <ipaddr> ltp16nVLANSettingsName.1.<vlan\_id> s "vlan\_name"**  
**ltp16nVLANSettingsIGMPQueryInterval.1.<vlan\_id> u <value>**  
**ltp16nVLANSettingsIGMPSnoopingEnabled.1.<vlan\_id> i 1/2**  
**ltp16nVLANSettingsIGMPSnoopingQuerierEnabled.1.<vlan\_id> i 1/2**  
**ltp16nVLANSettingsIGMPSnoopingStaticGroupRowStatus.1.<vlan\_id>.4.<group\_ip>.<port\_id> i 4/6**

*Example:*

```
snmpset -v2c -c private 192.168.1.2  
ltp16nVLANSettingsName.1.156 s "edited_by_snmp"  
ltp16nVLANSettingsIGMPQueryInterval.1.156 u 300  
ltp16nVLANSettingsIGMPSnoopingEnabled.1.156 i 1  
ltp16nVLANSettingsIGMPSnoopingQuerierEnabled.1.156 i 1  
ltp16nVLANSettingsIGMPSnoopingStaticGroupRowStatus.1.156.4.225.25.32.185.5 i 4
```

The command sets VLAN 156 to be named «edited\_by\_snmp», enables IGMP snooping and IGMP snooping querier, sets IGMPQueryInterval=300, adds a static multicast group 225.25.32.185 for interface gpon-port 5.

## VLAN Deletion

*Command format:*

**snmpset -v2c -c <rw\_community> <ipaddr> ltp16nVLANSettingsRowStatus.1.<vlan\_id> i 6**

*Example:*

```
snmpset -v2c -c private 192.168.1.2 ltp16nVLANSettingsRowStatus.1.156 i 6
```

The command deletes VLAN 156 from the configuration.

## Requesting the VLAN list and configuration of the specific VLAN

To request VLAN list, use the following request:

*Command format:*

**snmpwalk -v2c -c <ro\_community> <ipaddr> ltp16nVLANSettingsName.1**

*Example:*

```
snmpwalk -v2c -c public 192.168.1.2 ltp16nVLANSettingsName.1
```

The command lists VLAN

To obtain configuration of the specific VLAN, you can use the following request:

*Command format:*

**snmpget -v2c -c <ro\_community> <ipaddr> ltp16nVLANSettingsName.1.<vlan\_id>  
ltp16nVLANSettingsIGMPVersion.1.<vlan\_id>  
ltp16nVLANSettingsIGMPRobustnessVariable.1.<vlan\_id>  
ltp16nVLANSettingsIGMPQueryInterval.1.<vlan\_id>  
ltp16nVLANSettingsIGMPQueryResponseInterval.1.<vlan\_id>  
ltp16nVLANSettingsIGMPLastMemberQueryInterval.1.<vlan\_id>  
ltp16nVLANSettingsIGMPSnoopingQuerierEnabled.1.<vlan\_id>  
ltp16nVLANSettingsIGMPSnoopingQuerierDSCP.1.<vlan\_id>  
ltp16nVLANSettingsIGMPSnoopingQuerierFastLeaveEnabled.1.<vlan\_id>  
ltp16nVLANSettingsIGMPSnoopingQuerierIpAddress.1.<vlan\_id>  
ltp16nVLANSettingsIGMPSnoopingReplaceSourceIpAddress.1.<vlan\_id>**

*Example:*

```
snmpget -v2c -c public 192.168.1.2 ltp16nVLANSettingsName.1.156  
ltp16nVLANSettingsIGMPVersion.1.156  
ltp16nVLANSettingsIGMPRobustnessVariable.1.156  
ltp16nVLANSettingsIGMPQueryInterval.1.156  
ltp16nVLANSettingsIGMPQueryResponseInterval.1.156  
ltp16nVLANSettingsIGMPLastMemberQueryInterval.1.156  
ltp16nVLANSettingsIGMPSnoopingQuerierEnabled.1.156  
ltp16nVLANSettingsIGMPSnoopingQuerierDSCP.1.156  
ltp16nVLANSettingsIGMPSnoopingQuerierFastLeaveEnabled.1.156  
ltp16nVLANSettingsIGMPSnoopingQuerierIpAddress.1.156  
ltp16nVLANSettingsIGMPSnoopingReplaceSourceIpAddress.1.156
```

The command outputs the VLAN 156 configuration.

## IGMP configuration

### Global settings for enabling IGMP Snooping

To configure IGMP, use *ltp16nIGMPSnoopingTable*. OIDs for configuring global parameters are listed in the table below:

*Command format:*

**snmpset -v2c -c <rw\_community> <ipaddr> ltp16nIGMPSnoopingEnabled.1 i 1/2**

*Example:*

```
snmpset -v2c -c private 192.168.1.2 ltp16nIGMPSnoopingEnabled.1 i 1
```

The command enables IGMP Snooping.

## IGMP Snooping and VLAN Querier configuration

This setting is carried out similarly to the «VLAN Editing» section

## ONT profiles configuration

### Cross-connect

For operation with cross-connect profile, use *ltp16nONTProfileCrossConnectTable*.

#### Addition

*Command format:*

**snmpset -v2c -c <rw\_community> <ipaddr>  
ltp16nONTProfileCrossConnectRowStatus.<profile\_index> i 4**

*Example:*

```
snmpset -v2c -c private 192.168.1.2  
ltp16nONTProfileCrossConnectRowStatus.2 i 4
```

The command adds Cross-connect profile with index 2.

#### Editing

*Command format:*

**snmpset -v2c -c <rw\_community> <ipaddr>  
<parameter\_oid\_1>.<profile\_id> <par1\_type> <par1\_value>  
<parameter\_oid\_2>.<profile\_id> <par2\_type> <par2\_value>  
...  
<parameter\_oid\_N>.<profile\_id> <parN\_type> <parN\_value>**

Where:

- <parameter\_oid\_N> — the names of specific MIB parameters;
- <profile\_id> — profile index;
- <parN\_type> — value type of a parameter;
- <parN\_value> — parameter value.

*Example:*

```
snmpset -v2c -c private 192.168.1.2 ltp16nONTProfileCrossConnectName.2 s  
"edited_by_snmp" ltp16nONTProfileCrossConnectTrafficModel.2 i 1  
ltp16nONTProfileCrossConnectOntMode.2 i 1  
ltp16nONTProfileCrossConnectBridgeGroup.2 u 5 ltp16nONTProfileCrossConnectOuterVid.2 u 156
```

The command sets the name «edited\_by\_snmp», traffic-model multicast, ont mode = bridge, bridge group = 5 and Outer VID 156 for the Cross-connect profile with index 2.

#### Deletion

*Command format:*

```
snmpset -v2c -c <rw_community> <ipaddr>  
ltp16nONTProfileCrossConnectRowStatus.<profile_index> i 6
```

*Example:*

```
snmpset -v2c -c private 192.168.1.2 ltp16nONTProfileCrossConnectRowStatus.2 i 6
```

The command deletes Cross-connect profile with index 2.

## Profile list request

*Command format:*

```
snmpwalk -v2c -c <ro_community> <ipaddr> ltp16nONTProfileCrossConnectName
```

*Example:*

```
snmpwalk -v2c -c public 192.168.1.2 ltp16nONTProfileCrossConnectName
```

## DBA

For operation with DBA profiles, use *ltp16nONTProfileDBATable*.

### Addition

*Command format:*

```
snmpset -v2c -c <rw_community> <ipaddr>  
ltp16nONTProfileDBARowStatus.<profile_index> i 4
```

*Example:*

```
snmpset -v2c -c private 192.168.1.2 ltp16nONTProfileDBARowStatus.3 i 4
```

The command adds DBA profile with index 3.

### Editing

*Command format:*

```
snmpset -v2c -c <rw_community> <ipaddr>  
<parameter_oid_1>.<profile_id> <par1_type> <par1_value>  
<parameter_oid_2>.<profile_id> <par2_type> <par2_value>  
.....  
<parameter_oid_N>.<profile_id> <parN_type> <parN_value>
```

Where:

- <parameter\_oid\_N> — the names of specific MIB parameters;
- <profile\_id> — profile index;
- <parN\_type> — value type of a parameter;
- <parN\_value> — parameter value.

*Example:*

```
snmpset -v2c -c private 192.168.1.2  
ltp16nONTProfileDBAName.3 s "edited_by_snmp"  
ltp16nONTProfileDBAReportingMode.3 i 2  
ltp16nONTProfileDBABandwidthBesteffort.3 u 269248
```

The command sets the name «edited\_by\_snmp», status-reporting mode and besteffort bandwidth 3 for the DBA profile with index 3.

### Deletion



*Command format:*

```
snmpset -v2c -c <rw_community> <ipaddr>  
ltp16nONTProfileDBARowStatus.<profile_index> i 6
```

*Example:*

```
snmpset -v2c -c private 192.168.1.2 ltp16nONTProfileDBARowStatus.3 i 6
```

The command deletes DBA profile with index 3.

## Profile list request

*Command format:*

```
snmpwalk -v2c -c <ro_community> <ipaddr> ltp16nONTProfileDBAName
```

*Example:*

```
snmpwalk -v2c -c public 192.168.1.2 ltp16nONTProfileDBAName
```

## Ports

For operation with Ports profile, use the following tables:

- *ltp16nONTProfilePortsTable* — general profile parameters;
- *ltp16nONTProfilePortsUNIPortTable* — UNI ports.

## Addition

*Command format:*

```
snmpset -v2c -c <rw_community> <ipaddr>  
ltp16nONTProfilePortsRowStatus.<profile_index> i 4
```

*Example:*

```
snmpset -v2c -c private 192.168.1.2 ltp16nONTProfilePortsRowStatus.4 i 4
```

The command adds Ports profile with index 4.

## Editing

General parameters:

*Command format:*

```
snmpset -v2c -c <rw_community> <ipaddr> <parameter_oid_1>.<profile_id>  
<par1_type> <par1_value> <parameter_oid_2>.<profile_id> <par2_type>  
<par2_value> ... <parameter_oid_N>.<profile_id>  
<parN_type> <parN_value>
```

Where:

- <parameter\_oid\_N> — the names of specific MIB parameters;
- <profile\_id> — profile index;
- <parN\_type> — value type of a parameter;
- <parN\_value> — parameter value.

*Example:*

```
snmpset -v2c -c private 192.168.1.2 ltp16nONTProfilePortsName.4 s
"edited_by_snmp" ltp16nONTProfilePortsIGMPVersion.4 i 2
ltp16nONTProfilePortsIGMPQueryInterval.4 u 120
ltp16nONTProfilePortsVeipMulticast.4 i 1
ltp16nONTProfilePortsVeipIgmpDownstreamPriority.4 u 0
ltp16nONTProfilePortsVeipIgmpDownstreamVid.4 u 10
ltp16nONTProfilePortsVeipIgmpDownstreamTagControl.4 i 2
```

The command sets the name "edited\_by\_snmp", use IGMP v2, IGMP query interval 120, enable veip multicast, priority for igmp downstream traffic 0, vid 10 for igmp downstream traffic and adds a label for igmp downstream traffic for the Ports profile with index 4.

The parameters of UNI ports:

In addition to profile index, you should specify port index (1-4)

*Example:*

```
snmpset -v2c -c private 192.168.1.2 ltp16nONTProfilePortsUniPortMulticast.4.1 i 1
ltp16nONTProfilePortsUniPortBridgeGroup.4.1 u 70
ltp16nONTProfilePortsUniPortIgmpUpstreamPriority.4.1 u 0
ltp16nONTProfilePortsUniPortIgmpUpstreamVid.4.1 u 20
ltp16nONTProfilePortsUniPortIgmpUpstreamTagControl.4.1 i 2
```

The command sets bridge group = 100, enables multicast, priority for igmp upstream traffic 0, vid 20 for igmp upstream traffic and substitutes a label for igmp upstream traffic for profile Ports with index 4, for port with index 1.

## Deletion

*Command format:*

```
snmpset -v2c -c <rw_community> <ipaddr>
ltp16nONTProfilePortsRowStatus.<profile_index> i 6
```

*Example:*

```
snmpset -v2c -c private 192.168.1.2 ltp16nONTProfilePortsRowStatus.4 i 6
```

The command deletes Ports profile with index 4.

## Profile list request

*Command format:*

```
snmpwalk -v2c -c <ro_community> <ipaddr> ltp16nONTProfilePortsName
```

*Example:*

```
snmpwalk -v2c -c public 192.168.1.2 ltp16nONTProfilePortsName
```

## Management

For operation with Management profiles, use *ltp16nONTProfileManagementTable*.

### Addition

*Command format:*

```
snmpset -v2c -c <rw_community> <ipaddr>
ltp16nONTProfileManagementRowStatus.<profile_index> i 4
```

*Example:*

```
snmpset -v2c -c private 192.168.1.2 ltp16nONTProfileManagementRowStatus.10 i 4
```

The command adds Management profile with index 10.

## Editing

General parameters:

*Command format:*

```
snmpset -v2c -c <rw_community> <ipaddr> <parameter_oid_1>.<profile_id>  
<par1_type> <par1_value> <parameter_oid_2>.<profile_id> <par2_type>  
<par2_value> ... <parameter_oid_N>.<profile_id>  
<parN_type> <parN_value>
```

Where:

- <parameter\_oid\_N> — the names of specific MIB parameters;
- <profile\_id> — profile index;
- <parN\_type> — value type of a parameter;
- <parN\_value> — parameter value.

*Example:*

```
snmpset -v2c -c private 192.168.1.2 ltp16nONTProfileManagementName.10 s testProfile  
ltp16nONTProfileManagementDescription.10 s "Profile Management 10"  
ltp16nONTProfileManagementIphostId.10 u 10  
ltp16nONTProfileManagementOmciConfiguration.10 i 2  
ltp16nONTProfileManagementAcsUrl.10 s http://192.168.1.5  
ltp16nONTProfileManagementAcsUsername.10 s acs1  
ltp16nONTProfileManagementAcsPassword.10 s acsacs1
```

The command sets name "testProfile", iphost ID 10, disables configuration via OMCI, sets management server URL "<http://192.168.1.5>", management server username "acs1", password "acsacs1" for the Management profile with index 10.

## Deletion

*Command format:*

```
snmpset -v2c -c <rw_community> <ipaddr>  
ltp16nONTProfileManagementRowStatus.<profile_index> i 6
```

*Example:*

```
snmpset -v2c -c private 192.168.1.2 ltp16nONTProfileManagementRowStatus.10 i 6
```

The command removes Management profile with index 10.

## Profile list request

*Command format:*

```
snmpwalk -v2c -c <ro_community> <ipaddr> ltp16nONTProfileManagementName
```

*Example:*

```
snmpwalk -v2c -c public 192.168.1.2 ltp16nONTProfileManagementName
```

## Interface configuration

### GPON-ports

The configuration of the interface gpon-ports is performed using *ltp16nGponPortConfigTable*.

*Command format:*

```
snmpset -v2c -c <rw_community> <ipaddr>  
ltp16nGponPortConfigShutdown.1.<port_id> i 1/2
```

*Example:*

```
snmpset -v2c -c private 192.168.1.2 ltp16nGponPortConfigShutdown.1.1 i 1
```

The command disables interface gpon-port 1 in the configuration.

## Front-ports

The configuration of the interface front-ports is performed using *ltp16nFrontPortConfigTable*, *ltp16nFrontPortConfigAllowedVLANTable*.

*Command format:*

```
snmpset -v2c -c <rw_community> <ipaddr> <parameter_oid_1>.1.<port_id>  
<par1_type> <par1_value> <parameter_oid_2>.1.<port_id> <par2_type>  
<par2_value> ... <parameter_oid_N>.1.<port_id>  
<parN_type> <parN_value>
```

Where:

- <parameter\_oid\_N> — the names of specific MIB parameters;
- <port\_id> — port index;
- <parN\_type> — value type of a parameter;
- <parN\_value> — parameter value.

*Example:*

```
snmpset -v2c -c private 192.168.1.2 ltp16nFrontPortConfigShutdown.1.8 i 1
```

The command disables 10G-front-port 8 in the configuration.

Adding VLANs for interfaces is performed by creating entries with the desired VLAN as an index.

*Command format:*

```
snmpset -v2c -c <rw_community> <ipaddr>  
ltp16nFrontPortConfigAllowedVLANRowStatus.1.<port_id>.<vlan_id> i 4
```

*Example:*

```
snmpset -v2c -c private 192.168.1.2 ltp16nFrontPortConfigAllowedVLANRowStatus.1.5.156 i 4
```

The command adds VLAN 156 for interface 10G-front-port 5.

The VLAN for interfaces is removed by setting the value to 6.

*Example:*

```
snmpset -v2c -c private 192.168.1.2 ltp16nFrontPortConfigAllowedVLANRowStatus.1.5.156 i 6
```

The command removes VLAN 156 for interface 10G-front-port 5.

## OLT monitoring

### LTP general information

General information about LTP is collected in the *ltp16nSystem* and *ltp16nBoardStatus* groups.

*Command format:*

```
snmpget -v2c -c <ro_community> <ipaddr> <parameter_oid>.0
```

*Example:*

```
snmpget -v2c -c public 192.168.1.2 ltp16nSystemMacAddress.0  
ltp16nBoardStatusFan1RPM.0 ltp16nBoardStatusRAMFree.0
```

The command displays the LTP MAC address, the current Fan1 fan speed and the amount of free memory in Mb.

## The list of changes

Document version	Issue date	Firmware version	Revisions
Version 1.1	28.05.2021	1.2.0	Synchronization with firmware version 1.2.0  Sections added: <ul style="list-style-type: none"><li>• ONT Reboot and Reset;</li><li>• Ports profile configuration.</li></ul>
Version 1.0	30.12.2020	1.1.0	First issue