

## **Ethernet-коммутаторы агрегации**

MES2300-xx, MES3300-xx, MES3500I-08P,  
MES3500I-10P, MES5312, MES5316A, MES5324A,  
MES5332A, MES5310-48, MES5400-24,  
MES5400-48, MES5410-48, MES5500-32

**Мониторинг и управление Ethernet-коммутаторами MES по SNMP,  
версия ПО 6.6.5.4**

| Версия документа                       | Дата выпуска   | Содержание изменений                                                                                                   |
|----------------------------------------|----------------|------------------------------------------------------------------------------------------------------------------------|
| Версия 1.17                            | 23.12.2024     | Синхронизация с версией ПО 6.6.5.4                                                                                     |
| Версия 1.16                            | 01.11.2024     | Изменения в разделе:<br>4.1 Системные ресурсы                                                                          |
| Версия 1.15                            | 12.08.2024     | Изменения в разделе:<br>3.2 Работа с TFTP-сервером<br>3.4 Обновление программного обеспечения<br>4.1 Системные ресурсы |
| Версия 1.14                            | 24.05.2024     | Изменения в разделе:<br>6.1 Параметры Ethernet-интерфейсов                                                             |
| Версия 1.13                            | 18.04.2024     | Синхронизация с версией ПО 6.6.3                                                                                       |
| Версия 1.12                            | 15.03.2024     | Синхронизация с версией ПО 6.6.2.15                                                                                    |
| Версия 1.11                            | 29.02.2024     | Синхронизация с версией ПО 6.6.2.9                                                                                     |
| Версия 1.10                            | 15.12.2023     | Добавлен раздел:<br>6.2 Группы агрегации каналов — Link Aggregation Group (LAG)                                        |
| Версия 1.9                             | 09.10.2023     | Изменения в разделе:<br>4.1 Системные ресурсы                                                                          |
| Версия 1.8                             | 07.09.2023     | Синхронизация с версией ПО 6.5.1.4                                                                                     |
| Версия 1.7                             | 18.06.2023     | Изменения в разделе:<br>2 КРАТКИЕ ОБОЗНАЧЕНИЯ                                                                          |
| Версия 1.6                             | 07.04.2023     | Изменения в разделе:<br>12 ФУНКЦИИ УПРАВЛЕНИЯ                                                                          |
| Версия 1.5                             | 10.03.2023     | Синхронизация с версией ПО 6.5.0                                                                                       |
| Версия 1.4                             | 30.09.2022     | Изменения в разделе:<br>6.1 Параметры Ethernet-интерфейсов                                                             |
| Версия 1.3                             | 29.07.2022     | Добавлен раздел:<br>20 КОНФИГУРАЦИЯ VXLAN                                                                              |
| Версия 1.2                             | 31.01.2022     | Вторая публикация                                                                                                      |
| Версия 1.1                             | 04.02.2021     | Первая публикация                                                                                                      |
| <b>Версия программного обеспечения</b> | <b>6.6.5.4</b> |                                                                                                                        |

# СОДЕРЖАНИЕ

|      |                                                               |     |
|------|---------------------------------------------------------------|-----|
| 1    | НАСТРОЙКА SNMP-СЕРВЕРА И ОТПРАВКИ SNMP-TRAP .....             | 6   |
| 2    | КРАТКИЕ ОБОЗНАЧЕНИЯ .....                                     | 6   |
| 3    | РАБОТА С ФАЙЛАМИ .....                                        | 9   |
| 3.1  | Сохранение конфигурации .....                                 | 9   |
| 3.2  | Работа с TFTP-сервером .....                                  | 11  |
| 3.3  | Автоконфигурирование коммутатора .....                        | 13  |
| 3.4  | Обновление программного обеспечения .....                     | 14  |
| 4    | УПРАВЛЕНИЕ СИСТЕМОЙ .....                                     | 17  |
| 4.1  | Системные ресурсы .....                                       | 17  |
| 4.2  | Системные параметры .....                                     | 26  |
| 4.3  | Параметры стека .....                                         | 28  |
| 4.4  | Управление устройством .....                                  | 29  |
| 5    | НАСТРОЙКА СИСТЕМНОГО ВРЕМЕНИ .....                            | 32  |
| 6    | КОНФИГУРИРОВАНИЕ ИНТЕРФЕЙСОВ .....                            | 34  |
| 6.1  | Параметры Ethernet-интерфейсов .....                          | 34  |
| 6.2  | Группы агрегации каналов — Link Aggregation Group (LAG) ..... | 44  |
| 6.3  | Конфигурирование VLAN .....                                   | 45  |
| 6.4  | Настройка и мониторинг errdisable-состояния .....             | 50  |
| 6.5  | Настройка voice vlan .....                                    | 52  |
| 6.6  | Настройка LLDP .....                                          | 53  |
| 7    | НАСТРОЙКА IPV4-АДРЕСАЦИИ .....                                | 55  |
| 8    | НАСТРОЙКА IPV6-АДРЕСАЦИИ .....                                | 57  |
| 9    | НАСТРОЙКА GREEN ETHERNET .....                                | 58  |
| 10   | НАСТРОЙКА КОЛЬЦЕВЫХ ПРОТОКОЛОВ .....                          | 59  |
| 10.1 | Протокол ERPS .....                                           | 59  |
| 10.2 | Настройка протокола Spanning Tree .....                       | 61  |
| 11   | ГРУППОВАЯ АДРЕСАЦИЯ .....                                     | 65  |
| 11.1 | Правила групповой адресации (multicast addressing) .....      | 65  |
| 11.2 | Функции ограничения multicast-трафика .....                   | 67  |
| 12   | ФУНКЦИИ УПРАВЛЕНИЯ .....                                      | 70  |
| 12.1 | Механизм AAA .....                                            | 70  |
| 12.2 | Настройка доступа .....                                       | 74  |
| 13   | ЗЕРКАЛИРОВАНИЕ ПОРТОВ .....                                   | 76  |
| 14   | ФУНКЦИИ ДИАГНОСТИКИ ФИЗИЧЕСКОГО УРОВНЯ .....                  | 78  |
| 14.1 | Диагностика оптического трансивера .....                      | 78  |
| 15   | ФУНКЦИИ ОБЕСПЕЧЕНИЯ БЕЗОПАСНОСТИ .....                        | 79  |
| 15.1 | Функции обеспечения защиты портов .....                       | 79  |
| 15.2 | Контроль протокола DHCP и опции 82 .....                      | 83  |
| 15.3 | Защита IP-адреса клиента (IP source Guard) .....              | 85  |
| 15.4 | Контроль протокола ARP (ARP Inspection) .....                 | 87  |
| 15.5 | Проверка подлинности клиента на основе порта (802.1x) .....   | 88  |
| 15.6 | Механизм обнаружения петель (loopback-detection) .....        | 91  |
| 15.7 | Контроль широковещательного шторма (storm-control) .....      | 93  |
| 16   | КОНФИГУРИРОВАНИЕ IP И MAC ACP (СПИСКИ КОНТРОЛЯ ДОСТУПА) ..... | 95  |
| 17   | КОНФИГУРАЦИЯ ЗАЩИТЫ ОТ DOS-АТАК .....                         | 100 |
| 18   | КАЧЕСТВО ОБСЛУЖИВАНИЯ — QOS .....                             | 101 |
| 18.1 | Настройка QoS .....                                           | 101 |
| 18.2 | Статистика QoS .....                                          | 104 |
| 19   | МАРШРУТИЗАЦИЯ .....                                           | 106 |
| 19.1 | Статическая маршрутизация .....                               | 106 |
| 19.2 | Динамическая маршрутизация .....                              | 106 |
| 20   | КОНФИГУРАЦИЯ VXLAN .....                                      | 108 |

---

|                                                                                              |     |
|----------------------------------------------------------------------------------------------|-----|
| ПРИЛОЖЕНИЕ А. МЕТОДИКА РАСЧЕТА БИТОВОЙ МАСКИ .....                                           | 110 |
| ПРИЛОЖЕНИЕ Б. ПРИМЕР СОЗДАНИЯ ТИПОВОГО IP ACL .....                                          | 111 |
| ПРИЛОЖЕНИЕ В. ПРИМЕР СОЗДАНИЯ, НАПОЛНЕНИЯ И УДАЛЕНИЯ OFFSET-LIST С ПРИВЯЗКОЙ К MAC ACL ..... | 116 |

## УСЛОВНЫЕ ОБОЗНАЧЕНИЯ

| Обозначение                      | Описание                                                                                                                                  |
|----------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------|
| [ ]                              | В квадратных скобках в командной строке указываются необязательные параметры, но их ввод предоставляет определенные дополнительные опции. |
| { }                              | В фигурных скобках в командной строке указываются обязательные параметры.                                                                 |
| «,»<br>«-»                       | Данные знаки в описании команды используются для указания диапазонов.                                                                     |
| «   »                            | Данный знак в описании команды обозначает «или».                                                                                          |
| «/»                              | Данный знак при указании значений переменных разделяет возможные значения и значения по умолчанию.                                        |
| <i>Курсив Calibri</i>            | Курсивом Calibri указываются переменные или параметры, которые необходимо заменить соответствующим словом или строкой.                    |
| <b><i>Полужирный курсив</i></b>  | Полужирным курсивом выделены примечания и предупреждения.                                                                                 |
| <b>&lt;Полужирный курсив&gt;</b> | Полужирным курсивом в угловых скобках указываются названия клавиш на клавиатуре.                                                          |
| <b>Courier New</b>               | Полужирным Шрифтом Courier New записаны примеры ввода команд.                                                                             |

### Примечания и предупреждения



Примечания содержат важную информацию, советы или рекомендации по использованию и настройке устройства.



Предупреждения информируют пользователя о ситуациях, которые могут нанести вред устройству или человеку, привести к некорректной работе устройства или потере данных.

## 1 НАСТРОЙКА SNMP-СЕРВЕРА И ОТПРАВКИ SNMP-TRAP

```
snmp-server server
snmp-server community public ro
snmp-server community private rw
snmp-server host 192.168.1.1 traps version 2c private
```

## 2 КРАТКИЕ ОБОЗНАЧЕНИЯ

- **ifIndex** — индекс порта.

Может принимать следующие значения:

### 1. Коммутаторы доступа

| Модель коммутатора                                                       | Индексы                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
|--------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| MES2300-24<br>MES2300-24P<br>MES2300B-24<br>MES2300B-24F<br>MES2300D-24P | <ul style="list-style-type: none"> <li>- индексы 49-72 — gigabitethernet 1/0/1-24;</li> <li>- индексы 105-108 — tengigabitethernet 1/0/1-4;</li> <li>- индексы 157-180 — gigabitethernet 2/0/1-24;</li> <li>- индексы 213-216 — tengigabitethernet 2/0/1-4;</li> <li>- индексы 265-288 — gigabitethernet 3/0/1-24;</li> <li>- индексы 321-324 — tengigabitethernet 3/0/1-4;</li> <li>- индексы 373-396 — gigabitethernet 4/0/1-24;</li> <li>- индексы 429-432 — tengigabitethernet 4/0/1-4;</li> <li>- индексы 481-504 — gigabitethernet 5/0/1-24;</li> <li>- индексы 537-540 — tengigabitethernet 5/0/1-4;</li> <li>- индексы 589-612 — gigabitethernet 6/0/1-24;</li> <li>- индексы 645-648 — tengigabitethernet 6/0/1-4;</li> <li>- индексы 697-720 — gigabitethernet 7/0/1-24;</li> <li>- индексы 753-756 — tengigabitethernet 7/0/1-4;</li> <li>- индексы 805-828 — gigabitethernet 8/0/1-24;</li> <li>- индексы 861-864 — tengigabitethernet 8/0/1-4;</li> <li>- индексы 1000-1031 — Port-Channel 1-32;</li> <li>- индексы 7000-7063 — loopback 1-64.</li> </ul> |

### 2. Коммутаторы агрегации

| Модель коммутатора                                                                   | Индексы                                                                                                                                                                                                                                                                                                                       |
|--------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| MES3300-08F<br>MES3300-16F<br>MES3300-24<br>MES3300-24F<br>MES3300-48<br>MES3300-48F | <ul style="list-style-type: none"> <li>- индексы 49-72 — gigabitethernet 1/0/1-24;</li> <li>- индексы 105-108 — tengigabitethernet 1/0/1-4;</li> <li>- индексы 157-180 — gigabitethernet 2/0/1-24;</li> <li>- индексы 213-216 — tengigabitethernet 2/0/1-4;</li> <li>- индексы 265-288 — gigabitethernet 3/0/1-24;</li> </ul> |

|                                             |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
|---------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                                             | <ul style="list-style-type: none"> <li>- индексы 321-324 — tengigabitethernet 3/0/1-4;</li> <li>- индексы 373-396 — gigabitethernet 4/0/1-24;</li> <li>- индексы 429-432 — tengigabitethernet 4/0/1-4;</li> <li>- индексы 481-504 — gigabitethernet 5/0/1-24;</li> <li>- индексы 537-540 — tengigabitethernet 5/0/1-4;</li> <li>- индексы 589-612 — gigabitethernet 6/0/1-24;</li> <li>- индексы 645-648 — tengigabitethernet 6/0/1-4;</li> <li>- индексы 697-720 — gigabitethernet 7/0/1-24;</li> <li>- индексы 753-756 — tengigabitethernet 7/0/1-4;</li> <li>- индексы 805-828 — gigabitethernet 8/0/1-24;</li> <li>- индексы 861-864 — tengigabitethernet 8/0/1-4;</li> <li>- индексы 1000-1031 — Port-Channel 1-32;</li> <li>- индексы 7000-7063 — loopback 1-64.</li> </ul> |
| MES5312<br>MES5316A<br>MES5324A<br>MES5332A | <ul style="list-style-type: none"> <li>- индексы 1-32 — tengigabitethernet 1/0/1-32;</li> <li>- индексы 53-84 — tengigabitethernet 2/0/1-32;</li> <li>- индексы 105-136 — tengigabitethernet 3/0/1-32;</li> <li>- индексы 157-188 — tengigabitethernet 4/0/1-32;</li> <li>- индексы 209-240 — tengigabitethernet 5/0/1-32;</li> <li>- индексы 261-292 — tengigabitethernet 6/0/1-32;</li> <li>- индексы 313-344 — tengigabitethernet 7/0/1-32;</li> <li>- индексы 365-396 — tengigabitethernet 8/0/1-32;</li> <li>- индексы 1000-1127 — Port-Channel 1-128;</li> <li>- индексы 7000-7063 — loopback 1-64.</li> </ul>                                                                                                                                                              |

### 3. Индустриальные коммутаторы

| Модель коммутатора           | Индексы                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
|------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| MES2300DI-28                 | <ul style="list-style-type: none"> <li>- индексы 49-76 — gigabitethernet 1/0/1-28;</li> <li>- индексы 157-184 — gigabitethernet 2/0/1-28;</li> <li>- индексы 265-292 — gigabitethernet 3/0/1-28;</li> <li>- индексы 373-400 — gigabitethernet 4/0/1-28;</li> <li>- индексы 481-508 — gigabitethernet 5/0/1-28;</li> <li>- индексы 589-616 — gigabitethernet 6/0/1-28;</li> <li>- индексы 697-724 — gigabitethernet 7/0/1-28;</li> <li>- индексы 805-832 — gigabitethernet 8/0/1-28;</li> <li>- индексы 1000-1031 — Port-Channel 1-32;</li> <li>- индексы 7000-7063 — loopback 1-64.</li> </ul> |
| MES3500I-08P<br>MES3500I-10P | <ul style="list-style-type: none"> <li>- индексы 49-60 — gigabitethernet 1/0/1-12;</li> <li>- индексы 1000-1031 — Port-Channel 1-32;</li> <li>- индексы 7000-7063 — loopback 1-64.</li> </ul>                                                                                                                                                                                                                                                                                                                                                                                                  |

#### 4. Коммутаторы для ЦОД

| Модель коммутатора                                                      | Индексы                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
|-------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <p>MES5310-48</p> <p>MES5400-24</p> <p>MES5400-48</p> <p>MES5410-48</p> | <ul style="list-style-type: none"> <li>- индексы 1-48 — tengigabitethernet 1/0/1-48;</li> <li>- индексы 49-72 — twentyfivegigabitethernet 1/0/1-24;</li> <li>- индексы 103-108 — hundredgigabitethernet 1/0/1-6;</li> <li>- индексы 109-156 — tengigabitethernet 2/0/1-48;</li> <li>- индексы 157-180 — twentyfivegigabitethernet 2/0/1-24;</li> <li>- индексы 211-216 — hundredgigabitethernet 2/0/1-6;</li> <li>- индексы 217-264 — tengigabitethernet 3/0/1-48;</li> <li>- индексы 265-288 — twentyfivegigabitethernet 3/0/1-24;</li> <li>- индексы 319-324 — hundredgigabitethernet 3/0/1-6;</li> <li>- индексы 325-372 — tengigabitethernet 4/0/1-48;</li> <li>- индексы 373-396 — twentyfivegigabitethernet 4/0/1-24;</li> <li>- индексы 427-432 — hundredgigabitethernet 4/0/1-6;</li> <li>- индексы 433-480 — tengigabitethernet 5/0/1-48;</li> <li>- индексы 481-504 — twentyfivegigabitethernet 5/0/1-24;</li> <li>- индексы 535-540 — hundredgigabitethernet 5/0/1-6;</li> <li>- индексы 541-588 — tengigabitethernet 6/0/1-48;</li> <li>- индексы 589-612 — twentyfivegigabitethernet 6/0/1-24;</li> <li>- индексы 643-648 — hundredgigabitethernet 6/0/1-6;</li> <li>- индексы 649-696 — tengigabitethernet 7/0/1-48;</li> <li>- индексы 697-720 — twentyfivegigabitethernet 7/0/1-24;</li> <li>- индексы 751-756 — hundredgigabitethernet 7/0/1-6;</li> <li>- индексы 757-804 — tengigabitethernet 8/0/1-48;</li> <li>- индексы 805-828 — twentyfivegigabitethernet 8/0/1-24;</li> <li>- индексы 859-864 — hundredgigabitethernet 8/0/1-6;</li> <li>- индексы 1000-1127 — Port-Channel 1-128;</li> <li>- индексы 7000-7063 — loopback 1-64.</li> </ul> |
| <p>MES5500-32</p>                                                       | <ul style="list-style-type: none"> <li>- индексы 1-32 - hundredgigabitethernet 1/0/1-32</li> <li>- индексы 33-160 - twentyfivegigabitethernet 1/0/1-128</li> <li>- индексы 161-162 - tengigabitethernet 1/0/1-2</li> <li>- индексы 163-194 - hundredgigabitethernet 2/0/1-32</li> <li>- индексы 195-322 - twentyfivegigabitethernet 2/0/1-128</li> <li>- индексы 323-324 - tengigabitethernet 2/0/1-2</li> <li>- индексы 325-356 - hundredgigabitethernet 3/0/1-32</li> <li>- индексы 357-484 - twentyfivegigabitethernet 3/0/1-128</li> <li>- индексы 485-486 - tengigabitethernet 3/0/1-2</li> <li>- индексы 487-518 - hundredgigabitethernet 4/0/1-32</li> <li>- индексы 519-646 - twentyfivegigabitethernet 4/0/1-128</li> </ul>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |



|  |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
|--|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|  | <ul style="list-style-type: none"> <li>- индексы 647-648 - tengigabitethernet 4/0/1-2</li> <li>- индексы 649-680 - hundredgigabitethernet 5/0/1-32</li> <li>- индексы 681-808 - twentyfivegigabitethernet 5/0/1-128</li> <li>- индексы 809-810 - tengigabitethernet 5/0/1-2</li> <li>- индексы 811-842 - hundredgigabitethernet 6/0/1-32</li> <li>- индексы 843-970 - twentyfivegigabitethernet 6/0/1-128</li> <li>- индексы 971-972 - tengigabitethernet 6/0/1-2</li> <li>- индексы 973-1004 - hundredgigabitethernet 7/0/1-32</li> <li>- индексы 1005-1132 - twentyfivegigabitethernet 7/0/1-128</li> <li>- индексы 1133-1134 - tengigabitethernet 7/0/1-2</li> <li>- индексы 1135-1166 - hundredgigabitethernet 8/0/1-32</li> <li>- индексы 1167-1294 - twentyfivegigabitethernet 8/0/1-128</li> <li>- индексы 1295-1296 - tengigabitethernet 8/0/1-2</li> <li>- индексы 3000-3127 - Port-Channel 1-128</li> <li>- индексы 7000-7063 - loopback 1-64</li> </ul> |
|--|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

- **index-of-rule** — индекс правила в ACL всегда кратен 20. Если при создании правил будут указаны индексы не кратные 20, то после перезагрузки коммутатора порядковые номера правил в ACL станут кратны 20;
- **Значение поля N** — в IP и MAC ACL любое правило занимает от 1 до 3 полей в зависимости от его структуры;
- **IP address** — IP-адрес для управления коммутатором;

В приведенных в документе примерах используется следующий IP-адрес для управления:  
**192.168.1.30;**

- **ip address of tftp server** — IP-адрес TFTP-сервера;

В приведенных в документе примерах используется следующий IP-адрес TFTP-сервера:  
**192.168.1.1;**

- **community** — строка сообщества (пароль) для доступа по протоколу SNMP.

В приведенных в документе примерах используются следующие *community*:

**private** — права на запись (rw);

**public** — права на чтение (ro).

## 3 РАБОТА С ФАЙЛАМИ

### 3.1 Сохранение конфигурации

Сохранение конфигурации в энергонезависимую память

**MIB:** rlcopy.mib

**Используемые таблицы: rlCopyEntry — 1.3.6.1.4.1.89.87.2.1**

```
snmpset -v2c -c <community> <IP address> \  
1.3.6.1.4.1.89.87.2.1.3.1 i {local(1)} \  
1.3.6.1.4.1.89.87.2.1.7.1 i {runningConfig(2)} \  
1.3.6.1.4.1.89.87.2.1.8.1 i {local(1)} \  
1.3.6.1.4.1.89.87.2.1.12.1 i {startupConfig (3)} \  
1.3.6.1.4.1.89.87.2.1.17.1 i {createAndGo (4)}
```

**Пример****Команда CLI:**

```
copy running-config startup-config
```

**Команда SNMP:**

```
snmpset -v2c -c private 192.168.1.30 \  
1.3.6.1.4.1.89.87.2.1.3.1 i 1 \  
1.3.6.1.4.1.89.87.2.1.7.1 i 2 \  
1.3.6.1.4.1.89.87.2.1.8.1 i 1 \  
1.3.6.1.4.1.89.87.2.1.12.1 i 3 \  
1.3.6.1.4.1.89.87.2.1.17.1 i 4
```

**Сохранение конфигурации в энергозависимую память из энергонезависимой****MIB: rlcopy.mib****Используемые таблицы: rlCopyEntry — 1.3.6.1.4.1.89.87.2.1**

```
snmpset -v2c -c <community> <IP address> \  
1.3.6.1.4.1.89.87.2.1.3.1 i {local(1)} \  
1.3.6.1.4.1.89.87.2.1.7.1 i {startupConfig (3)} \  
1.3.6.1.4.1.89.87.2.1.8.1 i {local(1)} \  
1.3.6.1.4.1.89.87.2.1.12.1 i {runningConfig(2)} \  
1.3.6.1.4.1.89.87.2.1.17.1 i {createAndGo (4)}
```

**Пример****Команда CLI:**

```
copy startup-config running-config
```

**Команда SNMP:**

```
snmpset -v2c -c private 192.168.1.30 \  
1.3.6.1.4.1.89.87.2.1.3.1 i 1 \  
1.3.6.1.4.1.89.87.2.1.7.1 i 3 \  
1.3.6.1.4.1.89.87.2.1.8.1 i 1 \  
1.3.6.1.4.1.89.87.2.1.12.1 i 2 \  
1.3.6.1.4.1.89.87.2.1.17.1 i 4
```

**Удаление конфигурации из энергонезависимой памяти****MIB: RADLAN-rndMng****Используемые таблицы: rndAction — 1.3.6.1.4.1.89.1.2**

```
snmpset -v2c -c <community> <IP address> \  
1.3.6.1.4.1.89.1.2.0 i {eraseStartupCDB (20)}
```

### Пример удаления startup-config

Команда CLI:

```
delete startup-config
```

Команда SNMP:

```
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.89.1.2.0 i 20
```

## 3.2 Работа с TFTP-сервером

### Копирование конфигурации из энергозависимой памяти на TFTP-сервер

MIB: RADLAN-COPY-MIB

Используемые таблицы: rlCopyEntry — 1.3.6.1.4.1.89.87.2.1

```
snmpset -v2c -c <community> -t 5 -r 3 <IP address> \
1.3.6.1.4.1.89.87.2.1.3.1 i {local(1)} \
1.3.6.1.4.1.89.87.2.1.7.1 i {runningConfig(2)} \
1.3.6.1.4.1.89.87.2.1.8.1 i {tftp(3)} \
1.3.6.1.4.1.89.87.2.1.9.1 a {ip address of tftp server} \
1.3.6.1.4.1.89.87.2.1.11.1 s "MES-config.cfg" \
1.3.6.1.4.1.89.87.2.1.17.1 i {createAndGo (4)} \
1.3.6.1.4.1.35265.1.23.3.11.1.1.1 i {number of vrf (default 1)}
```



Команда 1.3.6.1.4.1.35265.1.23.3.11.1.1.1 i {number of vrf (default 1)} применяется в случае использования отличного от дефолтной VRF.

### Пример копирования из running-config на TFTP-сервер

Команда CLI:

```
copy running-config tftp://192.168.1.1/MES-config.cfg [vrf test]
```

Команда SNMP:

```
snmpset -v2c -c private -t 5 -r 3 192.168.1.30 \
1.3.6.1.4.1.89.87.2.1.3.1 i 1 \
1.3.6.1.4.1.89.87.2.1.7.1 i 2 \
1.3.6.1.4.1.89.87.2.1.8.1 i 3 \
1.3.6.1.4.1.89.87.2.1.9.1 a 192.168.1.1 \
1.3.6.1.4.1.89.87.2.1.11.1 s "conf.cfg" \
1.3.6.1.4.1.89.87.2.1.17.1 i 4 \
1.3.6.1.4.1.35265.1.23.3.11.1.1.1 i 2
```



Команда 1.3.6.1.4.1.35265.1.23.3.11.1.1.1 i 2 применяется в случае использования отличного от дефолтной VRF: vrf test.

### Копирование конфигурации в энергозависимую память с TFTP-сервера

MIB: rlcopymib

Используемые таблицы: rlCopyEntry — 1.3.6.1.4.1.89.87.2.1

```
snmpset -v2c -c <community> -t 5 -r 3 <IP address> \
1.3.6.1.4.1.89.87.2.1.3.1 i {tftp(3)} \
1.3.6.1.4.1.89.87.2.1.4.1 a {ip address of tftp server} \
1.3.6.1.4.1.89.87.2.1.6.1 s "MES-config.cfg" \
1.3.6.1.4.1.89.87.2.1.8.1 i {local(1)} \
1.3.6.1.4.1.89.87.2.1.12.1 i {runningConfig(2)} \
```

```
1.3.6.1.4.1.89.87.2.1.17.1 i {createAndGo (4)} \
1.3.6.1.4.1.35265.1.23.3.11.1.1.1 i {number of vrf (default 1)}
```



Команда 1.3.6.1.4.1.35265.1.23.3.11.1.1.1 i {number of vrf (default 1)} применяется в случае использования отличного от дефолтной VRF.

#### Пример копирования с TFTP-сервера в running-config

Команда CLI:

```
copy tftp://192.168.1.1/MES-config.cfg running-config [vrf test]
```

Команда SNMP:

```
snmpset -v2c -c private -t 5 -r 3 192.168.1.30 \
1.3.6.1.4.1.89.87.2.1.3.1 i 3 \
1.3.6.1.4.1.89.87.2.1.4.1 a 192.168.1.1 \
1.3.6.1.4.1.89.87.2.1.6.1 s "conf.cfg" \
1.3.6.1.4.1.89.87.2.1.8.1 i 1 \
1.3.6.1.4.1.89.87.2.1.12.1 i 2 \
1.3.6.1.4.1.89.87.2.1.17.1 i 4 \
1.3.6.1.4.1.35265.1.23.3.11.1.1.1 i 2
```



Команда 1.3.6.1.4.1.35265.1.23.3.11.1.1.1 i 2 применяется в случае использования отличного от дефолтной VRF: vrf test.

#### Копирование конфигурации из энергонезависимой памяти на TFTP-сервер

**MIB:** файл rlcory.mib

**Используемые таблицы:** rlcoryEntry — 1.3.6.1.4.1.89.87.2.1

```
snmpset -v2c -c <community> -t 5 -r 3 <IP address> \
1.3.6.1.4.1.89.87.2.1.3.1 i {local(1)} \
1.3.6.1.4.1.89.87.2.1.7.1 i {startupConfig (3)} \
1.3.6.1.4.1.89.87.2.1.8.1 i {tftp(3)} \
1.3.6.1.4.1.89.87.2.1.9.1 a {ip address of tftp server} \
1.3.6.1.4.1.89.87.2.1.11.1 s "MES-config.cfg" \
1.3.6.1.4.1.89.87.2.1.17.1 i {createAndGo (4)} \
1.3.6.1.4.1.35265.1.23.3.11.1.1.1 i {number of vrf (default 1)}
```



Команда 1.3.6.1.4.1.35265.1.23.3.11.1.1.1 i {number of vrf (default 1)} применяется в случае использования отличного от дефолтной VRF.

#### Пример копирования из startup-config на TFTP-сервер

Команда CLI:

```
boot config tftp://192.168.1.1/MES-config.cfg [vrf test]
```

Команда SNMP:

```
snmpset -v2c -c private -t 5 -r 3 192.168.1.30 \
1.3.6.1.4.1.89.87.2.1.3.1 i 1 \
1.3.6.1.4.1.89.87.2.1.7.1 i 2 \
1.3.6.1.4.1.89.87.2.1.8.1 i 3 \
1.3.6.1.4.1.89.87.2.1.9.1 a 192.168.1.1 \
1.3.6.1.4.1.89.87.2.1.11.1 s "conf.cfg" \
1.3.6.1.4.1.89.87.2.1.17.1 i 4 \
1.3.6.1.4.1.35265.1.23.3.11.1.1.1 i 2
```



Команда 1.3.6.1.4.1.35265.1.23.3.11.1.1.1 i 2 применяется в случае использования отличного от дефолтной VRF: vrf test.

## Копирование конфигурации в энергонезависимую память с TFTP-сервера

**MIB: RADLAN-COPY-MIB**

**Используемые таблицы: rlCopyEntry — 1.3.6.1.4.1.89.87.2.1**

```
snmpset -v2c -c <community> -t 5 -r 3 <IP address> \
  1.3.6.1.4.1.89.87.2.1.3.1 i {tftp(3)} \
  1.3.6.1.4.1.89.87.2.1.4.1 a {ip address of tftp server} \
  1.3.6.1.4.1.89.87.2.1.6.1 s "MES-config.cfg" \
  1.3.6.1.4.1.89.87.2.1.8.1 i {local(1)} \
  1.3.6.1.4.1.89.87.2.1.12.1 i {startupConfig (3)} \
  1.3.6.1.4.1.89.87.2.1.17.1 i {createAndGo (4)}
```

### **Пример копирования startup-config с TFTP-сервера**

**Команда CLI:**

```
boot config tftp://192.168.1.1/MES-config.cfg
```

**Команда SNMP:**

```
snmpset -v2c -c private -t 5 -r 3 192.168.1.30 \
  1.3.6.1.4.1.89.87.2.1.3.1 i 3 \
  1.3.6.1.4.1.89.87.2.1.4.1 a 192.168.1.1 \
  1.3.6.1.4.1.89.87.2.1.6.1 s "conf.cfg" \
  1.3.6.1.4.1.89.87.2.1.8.1 i 1 \
  1.3.6.1.4.1.89.87.2.1.12.1 i 3 \
  1.3.6.1.4.1.89.87.2.1.17.1 i 4
```

## **3.3 Автоконфигурирование коммутатора**

### Включение автоматического конфигурирования, базирующегося на DHCP (включено по умолчанию)

**MIB: radlan-dhcpcl-mib.mib**

**Используемые таблицы: rIDhcpCLOption67Enable — 1.3.6.1.4.1.89.76.9**

```
snmpset -v2c -c <community> <IP address> \
  1.3.6.1.4.1.89.76.9.0 i {enable(1), disable(2)}
```

### **Пример**

**Команда CLI:**

```
boot host auto-config
```

**Команда SNMP:**

```
snmpset -v2c -c private 192.168.1.30 \
  1.3.6.1.4.1.89.76.9.0 i 1
```

### 3.4 Обновление программного обеспечения

#### Обновление программного обеспечения коммутатора

Проходит в два этапа:

##### 1. Загрузка образа ПО

**MIB:** RADLAN-COPY-MIB

**Используемые таблицы:** rlCopyEntry — 1.3.6.1.4.1.89.87.2.1

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.87.2.1.3.1 i {tftp (3)} \
1.3.6.1.4.1.89.87.2.1.4.1 a {ip add of tftp server} \
1.3.6.1.4.1.89.87.2.1.6.1 s "image name" \
1.3.6.1.4.1.89.87.2.1.8.1 i {local(1)} \
1.3.6.1.4.1.89.87.2.1.12.1 i {image(8)} \
1.3.6.1.4.1.89.87.2.1.17.1 i {createAndGo(4)}\
1.3.6.1.4.1.35265.1.23.3.11.1.1.1 i {number of vrf (default 1)}
```



**Команда 1.3.6.1.4.1.35265.1.23.3.11.1.1.1 i {number of vrf (default 1)}** применяется в случае использования отличного от дефолтной VRF.

##### Пример

**Команда CLI:**

```
boot system tftp://192.168.1.1/mes5300a-611-R2.ros [vrf test]
```

**Команда SNMP:**

```
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.89.87.2.1.3.1 i 3 \
1.3.6.1.4.1.89.87.2.1.4.1 a 192.168.1.1 \
1.3.6.1.4.1.89.87.2.1.6.1 s "mes5300a-611-R2.ros" \
1.3.6.1.4.1.89.87.2.1.8.1 i 1 1.3.6.1.4.1.89.87.2.1.12.1 i 8 \
1.3.6.1.4.1.89.87.2.1.17.1 i 4 \
1.3.6.1.4.1.35265.1.23.3.11.1.1.1 i 2
```



**Команда 1.3.6.1.4.1.35265.1.23.3.11.1.1.1 i 2** применяется в случае использования отличного от дефолтной VRF: vrf test.

##### 2. Смена активного образа коммутатора

**MIB:** RADLAN-DEVICEPARAMS-MIB

**Используемые таблицы:** rndActiveSoftwareFileAfterReset — 1.3.6.1.4.1.89.2.13.1.1.3

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.2.13.1.1.3.1 i {image1 (1), image2 (2)}
```

##### Пример

**Команда CLI:**

```
boot system inactive-image
```

**Команда SNMP:**

```
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.89.2.13.1.1.3.1 i 1
```



После загрузки ПО с TFTP-сервера данная команда применяется автоматически.

### Перезагрузка коммутатора

**MIB:** rlmng.mib

**Используемые таблицы:** rlRebootDelay — 1.3.6.1.4.1.89.1.10

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.1.10.0 t {задержка времени перед перезагрузкой}
```

### **Пример перезагрузки, отложенной на 8 минут**

Команда CLI:

```
reload in 8
```

Команда SNMP:

```
snmpset -v2c -c private -r 0 192.168.1.30 \
1.3.6.1.4.1.89.1.10.0 t 48000
```



Для указания моментальной перезагрузки требуется указать значение t=0.

### Просмотр образа ПО

**MIB:** RADLAN-DEVICEPARAMS-MIB.mib

**Используемые таблицы:** rndActiveSoftwareFile — 1.3.6.1.4.1.89.2.13.1.1.2

```
snmpwalk -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.2.13.1.1.2
```

### **Пример**

Команда CLI:

```
show bootvar
```

Команда SNMP:

```
snmpwalk -v2c -c public 192.168.1.30 \
1.3.6.1.4.1.89.2.13.1.1.2
```



### **1) Возможные варианты:**

image1(1)

image2(2)

**2) Посмотреть активный образ ПО после перезагрузки можно в**  
**rndActiveSoftwareFileAfterReset — 1.3.6.1.4.1.89.2.13.1.1.3.**

### Просмотр загруженных образов ПО

**MIB:** RADLAN-DEVICEPARAMS-MIB.mib

**Используемые таблицы:** rndImageInfoTable — 1.3.6.1.4.1.89.2.16.1

```
snmpwalk -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.2.16.1
```

### Пример

Команда CLI:

```
show bootvar
```

Команда SNMP:

```
snmpwalk -v2c -c public 192.168.1.30 \  
1.3.6.1.4.1.89.2.16.1
```

### Просмотр текущей версии ПО коммутатора

**MIB:** RADLAN-DEVICEPARAMS-MIB.mib

**Используемые таблицы:** rndBrgVersion — 1.3.6.1.4.1.89.2.4

```
snmpwalk -v2c -c <community> <IP address> \  
1.3.6.1.4.1.89.2.4
```

### Пример

Команда CLI:

```
show version
```

Команда SNMP:

```
snmpwalk -v2c -c public 192.168.1.30 \  
1.3.6.1.4.1.89.2.4
```

### Просмотр текущей HW версии

**MIB:** RADLAN-DEVICEPARAMS-MIB.mib

**Используемые таблицы:** genGroupHWVersion — 1.3.6.1.4.1.89.2.11.1

```
snmpwalk -v2c -c <community> <IP address> \  
1.3.6.1.4.1.89.2.11.1
```

### Пример

Команда CLI:

```
show system id
```

Команда SNMP:

```
snmpwalk -v2c -c public 192.168.1.30 \  
1.3.6.1.4.1.89.2.11.1
```



## 4 УПРАВЛЕНИЕ СИСТЕМОЙ

### 4.1 Системные ресурсы

#### Просмотр серийного номера коммутатора

**MIB:** rlpphysdescription.mib

**Используемые таблицы:** rlpHdUnitGenParamSerialNum — 1.3.6.1.4.1.89.53.14.1.5

```
snmpwalk -v2c -c <community> <IP address> \  
1.3.6.1.4.1.89.53.14.1.5
```

#### **Пример**

Команда CLI:

```
show system id
```

Команда SNMP:

```
snmpwalk -v2c -c public 192.168.1.30 \  
1.3.6.1.4.1.89.53.14.1.5
```

#### Просмотр информации о загрузке tcam

**MIB:** RADLAN-QOS-CLI-MIB

**Используемые таблицы:** rlQosClassifierUtilizationPercent — 1.3.6.1.4.1.89.88.36.1.1.2

```
snmpwalk -v2c -c <community> <IP address> \  
1.3.6.1.4.1.89.88.36.1.1.2
```

#### **Пример**

Команда CLI:

```
show system tcam utilization
```

Команда SNMP:

```
snmpwalk -v2c -c public 192.168.1.30 \  
1.3.6.1.4.1.89.88.36.1.1.2
```

#### Просмотр максимального количества хостов

**MIB:** rltuning.mib

**Используемые таблицы:** rsMaxIpSfftEntries — 1.3.6.1.4.1.89.29.8.9.1

```
snmpwalk -v2c -c <community> <IP address> \  
1.3.6.1.4.1.89.29.8.9.1
```

#### **Пример**

Команда CLI:

```
show system router resources
```

Команда SNMP:

```
snmpwalk -v2c -c public 192.168.1.30 \  
1.3.6.1.4.1.89.29.8.9.1
```

### Просмотр используемого количества хостов

**MIB:** rlfft.mib

**Используемые таблицы:** rlSysmngTcamAllocInUseEntries — 1.3.6.1.4.1.89.204.1.1.1.5

```
snmpwalk -v2c -c <community> <IP address> \  
1.3.6.1.4.1.89.204.1.1.1.5.5.116.99.97.109.49.1
```

#### **Пример**

Команда CLI:

```
show system router resources
```

Команда SNMP:

```
snmpwalk -v2c -c public 192.168.1.30 \  
1.3.6.1.4.1.89.204.1.1.1.5.5.116.99.97.109.49.1
```

### Просмотр максимального количества маршрутов

**MIB:** rltuning.mib

**Используемые таблицы:** rsMaxIpPrefixes — 1.3.6.1.4.1.89.29.8.21.1

```
snmpwalk -v2c -c <community> <IP address> \  
1.3.6.1.4.1.89.29.8.21.1
```

#### **Пример**

Команда CLI:

```
show system router resources
```

Команда SNMP:

```
snmpwalk -v2c -c public 192.168.1.30 \  
1.3.6.1.4.1.89.29.8.21.1
```

### Просмотр используемого количества маршрутов

**MIB:** rlip.mib

**Используемые таблицы:** rlipTotalPrefixesNumber — 1.3.6.1.4.1.89.26.25

```
snmpwalk -v2c -c <community> <IP address> \  
1.3.6.1.4.1.89.26.25
```

#### **Пример**

Команда CLI:

```
show system router resources
```

Команда SNMP:

```
snmpwalk -v2c -c public 192.168.1.30 \  
1.3.6.1.4.1.89.26.25
```

### Просмотр максимального количества IP-интерфейсов

**MIB:** rltuning.mib

**Используемые таблицы:** rsMaxIpInterfaces — 1.3.6.1.4.1.89.29.8.25.1

```
snmpwalk -v2c -c <community> <IP address> \  
1.3.6.1.4.1.89.29.8.25.1
```

#### **Пример**

Команда CLI:

```
show system router resources
```

Команда SNMP:

```
snmpwalk -v2c -c public 192.168.1.30 \  
1.3.6.1.4.1.89.29.8.25.1
```

### Просмотр используемого количества IP-интерфейсов

**MIB:** rlip.mib

**Используемые таблицы:** rlipAddressesNumber — 1.3.6.1.4.1.89.26.23

```
snmpwalk -v2c -c <community> <IP address> \  
1.3.6.1.4.1.89.26.23
```

#### **Пример**

Команда CLI:

```
show system router resources
```

Команда SNMP:

```
snmpwalk -v2c -c public 192.168.1.30 \  
1.3.6.1.4.1.89.26.23
```

### Просмотр системного MAC-адреса коммутатора

**MIB:** rphysdescription.mib

**Используемые таблицы:** rIPhdsStackMacAddr — 1.3.6.1.4.1.89.53.4.1.7

```
snmpwalk -v2c -c <community> <IP address> \  
1.3.6.1.4.1.89.53.4.1.7
```

#### **Пример**

Команда CLI:

```
show system
```

Команда SNMP:

```
snmpwalk -v2c -c public 192.168.1.30 \  
1.3.6.1.4.1.89.53.4.1.7
```

### Просмотр Uptime коммутатора



Команда актуальна для Uptime меньше 497 дней.

**MIB:** SNMPv2-MIB

**Используемые таблицы:** sysUpTime — 1.3.6.1.2.1.1.3

```
snmpwalk -v2c -c <community> <IP address> \  
1.3.6.1.2.1.1.3
```

#### **Пример просмотра Uptime коммутатора**

Команда CLI:

```
show system
```

Команда SNMP:

```
snmpwalk -v2c -c public 192.168.1.30 \  
1.3.6.1.2.1.1.3
```

### Просмотр счётчика Uptime коммутатора в секундах

**MIB:** eltSystemGlobal.mib

**Используемые таблицы:** eltSysUpTimeInSec - .1.3.6.1.4.1.35265.1.23.1.10.3.1.1

```
snmpget -v2c -c <community> <IP address> \  
.1.3.6.1.4.1.35265.1.23.1.10.3.1.1.0
```

#### **Просмотр счётчика Uptime коммутатора в секундах**

Команда SNMP:

```
snmpget -v2c -c private 192.168.1.30 \  
.1.3.6.1.4.1.35265.1.23.1.10.3.1.1.0
```

### Просмотр счётчика Uptime коммутатора в секундах для юнита в стеке

**MIB:** ELTEX-PHYSICAL-DESCRIPTION-MIB.mib

**Используемые таблицы:** eltPhdUnitEnvParamTable - .1.3.6.1.4.1.35265.1.23.53.7

```
snmpget -v2c -c <community> <IP address> \  
.1.3.6.1.4.1.35265.1.23.53.7.1.1.{unit}
```

#### **Просмотр счётчика Uptime коммутатора в секундах для юнита в стеке**

Команда SNMP:

```
snmpget -v2c -c private 192.168.1.30 \  
.1.3.6.1.4.1.35265.1.23.53.7.1.1.1
```

### Просмотр Uptime порта

**MIB:** SNMPv2-MIB, IF-MIB

**Используемые таблицы:**

sysUpTime — 1.3.6.1.2.1.1.3

ifLastChange — 1.3.6.1.2.1.2.2.1.9

```
snmpwalk -v2c -c <community> <IP address> \
1.3.6.1.2.1.1.3
snmpwalk -v2c -c <community> <IP address> \
1.3.6.1.2.1.2.2.1.9.{ifindex}
```

### Пример просмотра Uptime порта TenGigabitEthernet 1/0/23

Команда CLI:

```
show interface status TenGigabitEthernet 1/0/23
```

Команда SNMP:

```
snmpwalk -v2c -c public 192.168.1.30 \
1.3.6.1.2.1.1.3
snmpwalk -v2c -c public 192.168.1.30 \
1.3.6.1.2.1.2.2.1.9.23
```



Из вывода первой команды необходимо отнять вывод второй команды. Полученное значение и будет являться uptime порта.

### Включение сервиса мониторинга приходящего на CPU трафика

MIB: rlsct.mib

Используемые таблицы: rISctCpuRateEnabled — 1.3.6.1.4.1.89.203.1

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.203.1.0 i {true(1), false(2)}
```

### Пример

Команда CLI:

```
service cpu-input-rate
```

Команда SNMP:

```
snmpset -v2c -c private 192.168.1.30 1.3.6.1.4.1.89.203.1.0 i 1
```

### Просмотр счетчиков и количества обрабатываемых CPU пакетов в секунду (по типам трафика)

MIB: rlsct.mib

Используемые таблицы: eltCpuRateStatisticsTable — 1.3.6.1.4.1.35265.1.23.1.773.1.2.1

```
snmpwalk -v2c -c <community> <IP address> \
1.3.6.1.4.1.35265.1.23.1.773.1.2.1.1.{rate in pps(2), packets count(3)}
```

### Пример просмотра количества обрабатываемых CPU в секунду пакетов

Команда CLI:

```
show cpu input-rate detailed
```

Команда SNMP:

```
snmpwalk -v2c -c public 192.168.1.30 \
1.3.6.1.4.1.35265.1.23.1.773.1.2.1.1.2
```



### Привязка индексов к типам трафика:

stack(1)  
 http(2)  
 telnet(3)  
 ssh(4)  
 snmp(5)  
 ip(6)  
 arp(7)  
 arpInspec(8)  
 stp(9)  
 ieee(10)  
 routeUnknown(11)  
 ipHopByHop(12)  
 mtuExceeded(13)  
 ipv4Multicast(14)  
 ipv6Multicast(15)  
 dhcpSnooping(16)  
 igmpSnooping(17)  
 mldSnooping(18)  
 ttlExceeded(19)  
 ipv4IllegalAddress(20)  
 ipv4HeaderError(21)  
 ipDaMismatch(22)  
 sflow(23)  
 logDenyAces(24)  
 dhcpv6Snooping(25)  
 vrrp(26)  
 logPermitAces(27)  
 ipv6HeaderError (28)

### Изменение лимитов CPU

**MIB:** eltSwitchRateLimiterMIB.mib

**Используемые таблицы:** eltCPURateLimiterTable — 1.3.6.1.4.1.35265.1.23.1.773.1.1.1

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.35265.1.23.1.773.1.1.1.2.{index} i {limiter value}
```

### **Пример установки ограничения SNMP-трафика для CPU в 512 pps**

Команда CLI:

```
service cpu-rate-limits snmp 512
```

Команда SNMP:

```
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.35265.1.23.1.773.1.1.1.2.4 i 512
```



### Список индексов:

eltCPURLTypeHttp(1)  
 eltCPURLTypeTelnet(2)  
 eltCPURLTypeSsh(3)  
 eltCPURLTypeSnmp(4)

```

eltCPURLTypeIp(5)
eltCPURLTypeLinkLocal(6)
eltCPURLTypeArpRouter(7)
eltCPURLTypeArpInspec(9)
eltCPURLTypeStpBpdu(10)
eltCPURLTypeOtherBpdu(11)
eltCPURLTypeIpRouting(12)
eltCPURLTypeIpOptions(13)
eltCPURLTypeDhcpSnoop(14)
eltCPURLTypeIgmpSnoop(16)
eltCPURLTypeMldSnoop(17)
eltCPURLTypeSflow(18)
eltCPURLTypeLogDenyAces(19)
eltCPURLTypeIpErrors(20)
eltCPURLTypeOther(22)

```

### Мониторинг загрузки CPU

**MIB:** rlmng.mib

#### **Используемые таблицы:**

```

rlCpuUtilDuringLastSecond — 1.3.6.1.4.1.89.1.7
rlCpuUtilDuringLastMinute — 1.3.6.1.4.1.89.1.8
rlCpuUtilDuringLast5Minutes — 1.3.6.1.4.1.89.1.9

```

- Загрузка за последних пять секунд: `snmpwalk -v2c -c <community> <IP address> 1.3.6.1.4.1.89.1.7;`
- Загрузка за 1 минуту: `snmpwalk -v2c -c <community> <IP address> 1.3.6.1.4.1.89.1.8;`
- Загрузка за 5 минут: `snmpwalk -v2c -c <community> <IP address> 1.3.6.1.4.1.89.1.9.`

#### **Пример просмотра загрузки CPU за последние пять секунд**

Команда CLI:

```
show cpu utilization
```

Команда SNMP:

```
snmpwalk -v2c -c public 192.168.1.30 \
1.3.6.1.4.1.89.1.7
```

### Включение мониторинга загрузки CPU по процессам

**MIB:** RADLAN-rndMng

**Используемые таблицы:** rlCpuTasksUtilEnable — 1.3.6.1.4.1.89.1.6

```

snmpset -v2c -c <community> <IP address>
1.3.6.1.4.1.89.1.6.0 i {true(1), false(2)}

```

#### **Пример**

Команда CLI:

```
service tasks-utilization
```

Команда SNMP:

```
snmpset -v2c -c private 192.168.1.30 1.3.6.1.4.1.89.1.6.0 i 1
```

## Мониторинг загрузки CPU по процессам

**MIB:** ELTEX-MES-MNG-MIB

### Используемые таблицы:

eltCpuTasksUtilStatisticsUtilizationDuringLast5Seconds — 1.3.6.1.4.1.35265.1.23.1.9.1.2.1.1.3

eltCpuTasksUtilStatisticsUtilizationDuringLastMinute — 1.3.6.1.4.1.35265.1.23.1.9.1.2.1.1.4

eltCpuTasksUtilStatisticsUtilizationDuringLast5Minutes — 1.3.6.1.4.1.35265.1.23.1.9.1.2.1.1.5

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.35265.1.23.1.9.1.2.1.1.3.{5sec(3), 1min(4), 5min(5)}.{task index}
```

### Пример просмотра загрузки по процессам за последние 5 секунд

Команда CLI:

```
show tasks utilization
```

Команда SNMP:

```
snmpwalk -v2c -c public 192.168.1.30 \
1.3.6.1.4.1.35265.1.23.1.9.1.2.1.1.3
```



### Привязка индексов к процессам

|           |           |            |
|-----------|-----------|------------|
| LTMR (0)  | 3SWQ (62) | MSRP (124) |
| ROOT (1)  | POLI (63) | MUXT (125) |
| HWAW (2)  | OBSR (64) | DMNG (126) |
| IT18 (3)  | NTPL (65) | DSYN (127) |
| IV11 (4)  | L2HU (66) | HSES (128) |
| MEMT (5)  | L2PS (67) | HSEU (129) |
| DDFG (6)  | RROt (68) | HSCS (130) |
| SYLG (7)  | LBDR (69) | UDLS (131) |
| CDB_ (8)  | LBDT (70) | L2TS (132) |
| SNMP (9)  | SFSM (71) | STPS (133) |
| NTST (10) | NSCT (72) | DTSA (134) |
| CNLD (11) | UDLD (73) | SS2M (135) |
| IOTG (12) | NSFP (74) | DSND (136) |
| IOUR (13) | NVCT (75) | STMB (137) |
| IOTM (14) | NACT (76) | MRDP (138) |
| CPUT (15) | NSTM (77) | MLDP (139) |
| HOST (16) | NINP (78) | XMOD (140) |
| CPUM (17) | CNCT (79) | URGN (141) |
| STUT (18) | QStT (80) | AAAT (142) |
| MEMV (19) | STPT (81) | AATT (143) |
| TBI_ (20) | L2PT (82) | SCPT (144) |
| BRMN (21) | L2UT (83) | DH6C (145) |
| TMNG (22) | BRGS (84) | RCLA (146) |
| COPY (23) | FHSS (85) | RCLB (147) |
| TRNS (24) | FHSF (86) | RCDS (148) |
| MROR (25) | FLNK (87) | PVNT (149) |
| DFST (26) | FFTT (88) | SETX (150) |
| SFTR (27) | IPAT (89) | EVTX (151) |
| SFMG (28) | IP6M (90) | SERX (152) |
| HCLT (29) | IP6L (91) | SCRX (153) |
| HCPT (30) | IP6C (92) | EVRX (154) |
| HCWT (31) | IP6R (93) | HLTX (155) |
| EVLC (32) | RPTS (94) | LACP (156) |



|           |            |            |
|-----------|------------|------------|
| SELC (33) | ARPG (95)  | GRN_ (157) |
| SCLC (34) | IPG_ (96)  | VPCM (158) |
| EVAU (35) | DNSC (97)  | VPCB (159) |
| SEAU (36) | ICMP (98)  | OAMT (160) |
| EVFB (37) | TFTP (99)  | IPMT (161) |
| EVLТ (38) | IPRD (100) | SNTP (162) |
| EPOE (39) | PNGA (101) | DHCP (163) |
| EVSC (40) | SLAP (102) | DHCp (164) |
| EVAN (41) | UDPR (103) | RELY (165) |
| CRPT (42) | VRRP (104) | MSSS (166) |
| HUTT (43) | TRCE (105) | WBAM (167) |
| HUTX (44) | SSLP (106) | WNTT (168) |
| ESTC (45) | WBSO (107) | RADS (169) |
| SSTC (46) | WBSR (108) | SPRT (170) |
| DSPT (47) | GOAH (109) | SNAS (171) |
| I2DT (48) | HTTC (110) | SNAE (172) |
| UNQt (49) | ECHO (111) | SNAD (173) |
| BOXS (50) | TNSR (112) | MNGT (174) |
| BSNC (51) | TNSL (113) | UTST (175) |
| BOXM (52) | SSHP (114) | SOCK (176) |
| B_RS (53) | SSHU (115) | SSHC (177) |
| TRMT (54) | SSHS (116) | TCPP (178) |
| D_SP (55) | SSHL (117) | MSCm (179) |
| D_LM (56) | PTPT (118) | STSA (180) |
| PLCT (57) | NBBT (119) | STSB (181) |
| PLCR (58) | MDLS (120) | STSC (182) |
| SW2M (59) | MSDP (121) | STSD (183) |
| exRX (60) | LICT (122) | STSE (184) |
| 3SWF (61) | SQIN (123) | STSF (185) |

### Просмотр общего объема оперативной памяти

**MIB:** ELTEX-PROCESS-MIB.mib

**Используемые таблицы:** eltexProcessMemoryEntry - 1.3.6.1.4.1.35265.41.1.2.1.1

```
snmpwalk -v2c -c <community> <IP address> \
1.3.6.1.4.1.35265.41.1.2.1.1.{Для MES5312, MES5316A, MES5324A, MES5332A - (3),
для MES5400-24, MES5400-48, MES5500-32 - (5)}.0
```

### Пример

**Команда CLI:**

```
show cpu utilization
```

**Команда SNMP:**

```
snmpwalk -v2c -c public 192.168.1.30 \
1.3.6.1.4.1.35265.41.1.2.1.1.3.0
```

### Просмотр свободного объема оперативной памяти

**MIB:** ELTEX-PROCESS-MIB.mib

**Используемые таблицы:** eltexProcessMemoryEntry - 1.3.6.1.4.1.35265.41.1.2.1.1

```
snmpwalk -v2c -c <community> <IP address> \  
1.3.6.1.4.1.35265.41.1.2.1.1.{Для MES5312, MES5316A, MES5324A, MES5332A - (7),  
для MES5400-24, MES5400-48, MES5500-32 - (9)}.0
```

#### **Пример**

**Команда CLI:**

```
show cpu utilization
```

**Команда SNMP:**

```
snmpwalk -v2c -c public 192.168.1.30 \  
1.3.6.1.4.1.35265.41.1.2.1.1.7.0
```

### Включение поддержки сверхдлинных кадров (jumbo-frames)

**MIB:** radlan-jumboframes-mib.mib

**Используемые таблицы:** rJumboFrames — 1.3.6.1.4.1.89.91

```
snmpset -v2c -c <community> <IP address> \  
1.3.6.1.4.1.89.91.2.0 i {enabled(1), disabled(2)}
```

#### **Пример**

**Команда CLI:**

```
port jumbo-frame
```

**Команда SNMP:**

```
snmpset -v2c -c private 192.168.1.30 \  
1.3.6.1.4.1.89.91.2.0 i 1
```

## **4.2 Системные параметры**

### Контроль состояния блоков питания

**MIB:** rPhysDescription.mib

**Используемые таблицы:** rPhdUnitEnvParamTable — 1.3.6.1.4.1.89.53.15

- Основной блок питания: snmpwalk -v2c -c <community> <IP address> 1.3.6.1.4.1.89.53.15.1.2;
- Резервный блок питания: snmpwalk -v2c -c <community> <IP address> 1.3.6.1.4.1.89.53.15.1.3.

#### **Пример просмотра состояния основного блока питания**

**Команда CLI:**

```
show system
```

**Команда SNMP:**

```
snmpwalk -v2c -c public 192.168.1.30 \  
1.3.6.1.4.1.89.53.15.1.2
```



1) для основного блока питания возможны следующие состояния:

normal (1)  
warning (2)  
critical (3)  
shutdown (4)  
notPresent (5)  
notFunctioning (6)

2) для резервного блока питания возможны следующие состояния:

normal (1)  
warning (2)  
critical (3)  
shutdown (4)  
notPresent (5)  
notFunctioning (6)

### Контроль состояния вентиляторов

MIB: rphysdescription.mib

Используемые таблицы: rIPhdUnitEnvParamTable — 1.3.6.1.4.1.89.53.15

- Вентилятор 1: snmpwalk -v2c -c <community> <IP address> 1.3.6.1.4.1.89.53.15.1.4
- Вентилятор 2: snmpwalk -v2c -c <community> <IP address> 1.3.6.1.4.1.89.53.15.1.5
- Вентилятор 3: snmpwalk -v2c -c <community> <IP address> 1.3.6.1.4.1.89.53.15.1.6
- Вентилятор 4: snmpwalk -v2c -c <community> <IP address> 1.3.6.1.4.1.89.53.15.1.7

### **Пример просмотра состояния вентилятора 3 коммутатора MES5332A**

Команда CLI:

```
show system
```

Команда SNMP:

```
snmpwalk -v2c -c public 192.168.1.30 \
1.3.6.1.4.1.89.53.15.1.6
```



Возможны следующие состояния:

normal (1)  
notFunctioning (5)

### Контроль показаний температурных датчиков

MIB: RADLAN-MIB

Используемые таблицы: rEnv — 1.3.6.1.4.1.89.83.2.1.1.1.4

Температурный датчик 1: snmpwalk -v2c -c <community> <IP address> 1.3.6.1.4.1.89.83.2.1.1.1.4

### **Пример просмотра температуры датчика**

Команда CLI:

```
show system sensors
```

Команда SNMP:

```
snmpwalk -v2c -c public 192.168.1.30 \
1.3.6.1.4.1.89.83.2.1.1.1.4
```

### Контроль состояния температурных датчиков

**MIB:** rlphysdescription.mib

**Используемые таблицы:** rIPhdUnitEnvParamTable — 1.3.6.1.4.1.89.53.15

Температурный датчик 1: snmpwalk -v2c -c <community> <IP address> 1.3.6.1.4.1.89.53.15.1.11

#### **Пример**

Команда CLI:

```
show system sensors
```

Команда SNMP:

```
snmpwalk -v2c -c public 192.168.1.30 \  
1.3.6.1.4.1.89.53.15.1.11
```

## **4.3 Параметры стека**

### Мониторинг параметров стека

**MIB:** rlphysdescription.mib

**Используемые таблицы:** rIPhdStackTable — 1.3.6.1.4.1.89.53.4

```
snmpwalk -v2c -c <community> <IP address> \  
1.3.6.1.4.1.89.53.4
```

#### **Пример просмотра параметров стека**

Команда CLI:

```
show stack
```

Команда SNMP:

```
snmpwalk -v2c -c public 192.168.1.30 \  
1.3.6.1.4.1.89.53.4
```

### Мониторинг стековых портов

**MIB:** rlphysdescription.mib

**Используемые таблицы:** rICascadeTable — 1.3.6.1.4.1.89.53.23

```
snmpwalk -v2c -c <community> <IP address> \  
1.3.6.1.4.1.89.53.23
```

#### **Пример просмотра состояния стековых портов**

Команда CLI:

```
show stack links
```

Команда SNMP:

```
snmpwalk -v2c -c public 192.168.1.30 \  
1.3.6.1.4.1.89.53.23
```

## 4.4 Управление устройством

### Задать/сменить hostname на устройстве

**MIB:** SNMPv2-MIB

**Используемые таблицы:** sysName — 1.3.6.1.2.1.1.5

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.2.1.1.5.0 s "{hostname}"
```

#### **Пример присвоения hostname "mes5332A"**

Команда CLI:

```
hostname mes2324
```

Команда SNMP:

```
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.2.1.1.5.0 s "mes5332A"
```

### Включение/отключение management acl

**MIB:** RADLAN-MNGINF-MIB

**Используемые таблицы:**

rlMngInfEnable — 1.3.6.1.4.1.89.89.2

rlMngInfActiveListName — 1.3.6.1.4.1.89.89.3

```
snmpset -v2c -c <community> <IP address>
1.3.6.1.4.1.89.89.2.0 i {true(1), false(2)}\
1.3.6.1.4.1.89.89.3.0 s {name}do ping
```

#### **Пример включения management acl с именем eltex**

Команда CLI:

```
management access-class eltex
```

Команда SNMP:

```
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.89.89.2.0 i 1 \
1.3.6.1.4.1.89.89.3.0 s eltex
```

### Использование утилиты ping

**MIB:** rlapplication.mib

**Используемые таблицы:** rsPingInetTable — 1.3.6.1.4.1.89.35.4.2

```
snmpset -v2c -c <community> <IP address>\
1.3.6.1.4.1.89.35.4.1.1.2.{IP address}> i {Packet count}\
1.3.6.1.4.1.89.35.4.1.1.3.{IP address}> i {Packet Size}\
1.3.6.1.4.1.89.35.4.1.1.4.{IP address}> i {Packet Timeout}\
1.3.6.1.4.1.89.35.4.1.1.5.{IP address}> i {Ping Delay}\
1.3.6.1.4.1.89.35.4.1.1.6.{IP address}> i {Send SNMP Trap(2)}\
1.3.6.1.4.1.89.35.4.1.1.14.{IP address}> i {createAndGo(4), destroy(6),
active(1)}
```

### Пример команды ping узла 192.168.1.1

Команда CLI:

```
ping 192.168.1.1 count 10 size 250 timeout 1000
```

Команда SNMP:

```
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.89.35.4.1.1.2.192.168.1.1 i 10 \
1.3.6.1.4.1.89.35.4.1.1.3.192.168.1.1 i 250 \
1.3.6.1.4.1.89.35.4.1.1.4.192.168.1.1 i 1000 \
1.3.6.1.4.1.89.35.4.1.1.5.192.168.1.1 i 0 \
1.3.6.1.4.1.89.35.4.1.1.6.192.168.1.1 i 2 \
1.3.6.1.4.1.89.35.4.1.1.14.192.168.1.1 i 4
```



При установке в поле **rsPingEntryStatus** значения 4 (**createAndGo**) создается и активируется операция **ping**.

Чтобы повторно пропинговать удаленный хост, требуется в поле **rsPingEntryStatus** выставить значение 1(**active**).

После окончания операции обязательно надо удалить все записи, выставив в поле **rsPingEntryStatus** значение 6 (**destroy**). Иначе через CLI и SNMP операцию **ping** до другого хоста выполнить не удастся.

Пример удаления:

```
snmpset -v2c -c private 192.168.1.30\
1.3.6.1.4.1.89.35.4.1.1.2.192.168.1.1 i 10\
1.3.6.1.4.1.89.35.4.1.1.3.192.168.1.1 i 250\
1.3.6.1.4.1.89.35.4.1.1.4.192.168.1.1 i 1000\
1.3.6.1.4.1.89.35.4.1.1.5.192.168.1.1 i 0\
1.3.6.1.4.1.89.35.4.1.1.6.192.168.1.1 i 2\
1.3.6.1.4.1.89.35.4.1.1.14.192.168.1.1 i 6
```

### Мониторинг утилиты ping

**MIB:** rlapplcation.mib

**Используемые таблицы:** rsPingEntry — 1.3.6.1.4.1.89.35.4.1.1

```
snmpwalk -v2c -c <community> <IP address>\
```

```
1.3.6.1.4.1.89.35.4.1.1.{Количество отправленных пакетов(7), Количество принятых
пакетов(8), Минимальное время ответа(9), Среднее время ответа(10), Максимальное
время ответа(11)}
```

**Пример просмотра количества принятых пакетов**

Команда CLI:

```
ping 192.168.1.1
```

Команда SNMP:

```
snmpwalk -v2c -c public 192.168.1.30 \
1.3.6.1.4.1.89.35.4.1.1.8
```



При установке в поле **rsPingEntryStatus** значения 6 (**destroy**) мониторинг будет запрещен до создания новой операции.

## Настройка системного журнала

MIB: DRAFT-IETF-SYSLOG-DEVICE-MIB

Используемые таблицы: snmpSyslogCollectorEntry — 1.3.6.1.4.1.89.82.1.2.4.1

```
snmpset -v2c -c <community> -t 10 -r 5 <IP address> \
1.3.6.1.4.1.89.82.1.2.4.1.2.1 s "{name}" \
1.3.6.1.4.1.89.82.1.2.4.1.3.1 i {ipv4(1), ipv6(2)} \
1.3.6.1.4.1.89.82.1.2.4.1.4.1 x {ip add in HEX} \
1.3.6.1.4.1.89.82.1.2.4.1.5.1 u {udp port number} \
1.3.6.1.4.1.89.82.1.2.4.1.6.1 i {syslog facility(16-24)} \
1.3.6.1.4.1.89.82.1.2.4.1.7.1 i {severity level} \
1.3.6.1.4.1.89.82.1.2.4.1.9.1 i {createAndGo(4), destroy(6)}
```

### Пример добавления сервера для логирования

Команда CLI:

```
logging host 192.168.1.1 description 11111
```

Команда SNMP:

```
snmpset -v2c -c private -t 10 -r 5 192.168.1.30 \
1.3.6.1.4.1.89.82.1.2.4.1.2.1 s "11111" \
1.3.6.1.4.1.89.82.1.2.4.1.3.1 i 1 \
1.3.6.1.4.1.89.82.1.2.4.1.4.1 x C0A80101 \
1.3.6.1.4.1.89.82.1.2.4.1.5.1 u 514 \
1.3.6.1.4.1.89.82.1.2.4.1.6.1 i 23 \
1.3.6.1.4.1.89.82.1.2.4.1.7.1 i 6 \
1.3.6.1.4.1.89.82.1.2.4.1.9.1 i 4
```



**Severity level задается следующим образом:**

emergency(0),  
alert(1),  
critical(2),  
error(3),  
warning(4),  
notice(5),  
info(6),  
debug(7)

**Facility:**

local0(16),  
local1(17),  
local2(18),  
local3(19),  
local4(20),  
local5(21),  
local6(22),  
local7(23),  
no-map(24)

## 5 НАСТРОЙКА СИСТЕМНОГО ВРЕМЕНИ

### Настройка адреса SNMP-сервера

**MIB:** rlsntp.mib

**Используемые таблицы:** rlSntpConfigServerInetTable — 1.3.6.1.4.1.89.92.2.2.17

```
snmpset -v2c -c <community> <IP address> \
  1.3.6.1.4.1.89.92.2.2.17.1.3.1.4.{ip address in DEC. Байты IP-адреса
разделяются точками} i {true(1), false(2). Указание значения poll} \
  1.3.6.1.4.1.89.92.2.2.17.1.9.1.4.{ip address in DEC. Байты IP-адреса
разделяются точками} u 0 \
  1.3.6.1.4.1.89.92.2.2.17.1.10.1.4.{ip address in DEC. Байты IP-адреса
разделяются точками} i {createAndGo(4), destroy(6)}
```

#### **Пример указания SNMP-сервера с IP-адресом 91.226.136.136**

**Команда CLI:**

```
sntp server 91.226.136.136 poll
```

**Команда SNMP:**

```
snmpset -v2c -c private 192.168.1.30 \
  1.3.6.1.4.1.89.92.2.2.17.1.3.1.4.91.226.136.136 i 1 \
  1.3.6.1.4.1.89.92.2.2.17.1.9.1.4.91.226.136.136 u 0 \
  1.3.6.1.4.1.89.92.2.2.17.1.10.1.4.91.226.136.136 i 4
```

### Установка времени опроса для SNMP-клиента

**MIB:** rlsntp.mib

**Используемые таблицы:** rlSntpNtpConfig — 1.3.6.1.4.1.89.92.2.1

```
snmpset -v2c -c <community> <IP address> \
  1.3.6.1.4.1.89.92.2.1.4.0 i {range 60-86400}
```

#### **Пример установки времени опроса в 60 секунд**

**Команда CLI:**

```
sntp client poll timer 60
```

**Команда SNMP:**

```
snmpset -v2c -c private 192.168.1.30 \
  1.3.6.1.4.1.89.92.2.1.4.0 i 60
```



Чтобы вернуться к настройкам по умолчанию достаточно установить время в 1024 сек.

### Настройка работы одноадресных SNMP-клиентов

**MIB:** rlsntp.mib

**Используемые таблицы:** rlSntpConfig — 1.3.6.1.4.1.89.92.2.2

```
snmpset -v2c -c <community> <IP address> \
  1.3.6.1.4.1.89.92.2.2.5.0 i {true(1), false(2)}
```



### Пример разрешения последовательного опроса SNMP-серверов

Команда CLI:

```
sntp unicast client poll
```

Команда SNMP:

```
snmpset -v2c -c private 192.168.1.30 \  
1.3.6.1.4.1.89.92.2.2.5.0 i 1
```

### Добавление часового пояса

MIB: rlsntp.mib

Используемые таблицы: rlTimeSyncMethodMode — 1.3.6.1.4.1.89.92.1

```
snmpset -v2c -c <community> <IP address> \  
1.3.6.1.4.1.89.92.1.6.0 s "{TimeZone}" \  
1.3.6.1.4.1.89.92.1.7.0 s "{NameZone}"
```

### Пример добавления часового пояса на устройстве

Команда CLI:

```
clock timezone test +7
```

Команда SNMP:

```
snmpset -v2c -c private 192.168.1.30 \  
1.3.6.1.4.1.89.92.1.6.0 s "+7:00" \  
1.3.6.1.4.1.89.92.1.7.0 s "test"
```

## 6 КОНФИГУРИРОВАНИЕ ИНТЕРФЕЙСОВ

### 6.1 Параметры Ethernet-интерфейсов

#### Просмотр Description порта

**MIB:** IF-MIB или eltMng.mib

**Используемые таблицы:** ifAlias — 1.3.6.1.2.1.31.1.1.1.18 или iflongDescr — 1.3.6.1.4.1.35265.1.23.1.1.31.1.1.1.1

```
snmpwalk -v2c -c <community> <IP address> \  
1.3.6.1.2.1.31.1.1.1.18.{ifIndex}
```

```
snmpwalk -v2c -c <community> <IP address> \  
1.3.6.1.4.1.35265.1.23.1.1.31.1.1.1.1.{ifIndex}
```

#### **Пример просмотра Description на интерфейсе TenGigabitEthernet 1/0/23**

**Команда CLI:**

```
show interfaces description TenGigabitEthernet 1/0/23
```

**Команда SNMP:**

```
snmpwalk -v2c -c public 192.168.1.30 \  
1.3.6.1.2.1.31.1.1.1.18.23  
snmpwalk -v2c -c public 192.168.1.30 \  
1.3.6.1.4.1.35265.1.23.1.1.31.1.1.1.1.23
```

#### Просмотр Description vlan

**MIB:** Q-BRIDGE-MIB

**Используемые таблицы:** dot1qVlanStaticTable — 1.3.6.1.2.1.17.7.1.4.3

```
snmpwalk -v2c -c <community> <IP address> \  
1.3.6.1.2.1.17.7.1.4.3.1.1.{vlan id}
```

#### **Пример просмотра Description vlan 100**

**Команда CLI:**

```
show interfaces description vlan 100
```

**Команда SNMP:**

```
snmpwalk -v2c -c public 192.168.1.30 \  
1.3.6.1.2.1.17.7.1.4.3.1.1.1
```

#### Просмотр скорости на интерфейсе

**MIB:** IF-MIB

**Используемые таблицы:** ifHighSpeed — 1.3.6.1.2.1.31.1.1.1.15

```
snmpwalk -v2c -c <community> <IP address> \  
1.3.6.1.2.1.31.1.1.1.15.{ifIndex}
```

### Пример выключения negotiation на TenGigabitethernet 1/0/23

Команда CLI:

```
show interface status TenGigabitethernet 1/0/23
```

Команда SNMP:

```
snmpwalk -v2c -c public 192.168.1.30 \
1.3.6.1.2.1.31.1.1.1.15.23
```

### Включение/выключение автосогласования скорости на интерфейсе

MIB: rlintefaces.mib

Используемые таблицы: swlfSpeedDuplexAutoNegotiation — 1.3.6.1.4.1.89.43.1.1.16

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.43.1.1.16.{ifIndex} i {negotiation(1), no negotiation(2)}
```

### Пример выключения negotiation на TenGigabitethernet 1/0/23

Команда CLI:

```
interface TenGigabitethernet 1/0/23
no negotiation
```

Команда SNMP:

```
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.89.43.1.1.16.23 i 2
```

### Установка режимов автосогласования скорости на интерфейсе

MIB: swinterfaces.mib

Используемые таблицы: swlfAdminSpeedDuplexAutoNegotiationLocalCapabilities — 1.3.6.1.4.1.89.43.1.1.40

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.43.1.1.40.{ifIndex} x "{negotiation mode(HEX-string)}"
```

### Пример настройки автосогласования на скорости 1000f и 10000f на интерфейсе TenGigabitethernet 1/0/23

Команда CLI:

```
interface TenGigabitethernet 1/0/23
negotiation 1000f 10000f
```

Команда SNMP:

```
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.89.43.1.1.40.23 x 14
```



1) В двоичной системе 1000f и 10000f записывается как 00110000000. В HEX системе счисления это 180.

### 2) Описание битов

Default(0),  
Unknown(1),  
TenHalf(2),  
TenFull(3),

FastHalf(4),  
FastFull(5),  
GigaHalf(6),  
GigaFull(7),  
TenGigaFull(8),  
FiveGigaFull(9),  
TwoPointFiveFull(10).

**Порядок битов**

**10 9 8 7 6 5 4 3 2 1 0**

**Просмотр duplex-режима порта**

**MIB:** EtherLike-MIB

**Используемые таблицы:** dot3StatsDuplexStatus — 1.3.6.1.2.1.10.7.2.1.19

```
snmpwalk -v2c -c <community> <IP address> \
1.3.6.1.2.1.10.7.2.1.19.{ifindex}
```

**Пример просмотра режима duplex порта TenGigabitEthernet 1/0/23**

Команда CLI:

```
show interfaces status TenGigabitEthernet 1/0/23
```

Команда SNMP:

```
snmpwalk -v2c -c public 192.168.1.30 \
1.3.6.1.2.1.10.7.2.1.19.23
```



**Расшифровка выдаваемых значений**

unknown (1)  
halfDuplex (2)  
fullDuplex (3)

**Смена duplex-режима на интерфейсе**

**MIB:** RADLAN-rlInterfaces

**Используемые таблицы:** swIfDuplexAdminMode — 1.3.6.1.4.1.89.43.1.1.3

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.43.1.1.3.{ifIndex} i {none(1),half(2),full (3)}
```

**Пример смены режима duplex порта TenGigabitEthernet 1/0/23**

Команда CLI:

```
interface TenGigabitEthernet 1/0/23
duplex half
```

Команда SNMP:

```
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.89.43.1.1.3.23 i 2
```

## Просмотр среды передачи интерфейса

**MIB:** EtherLike-MIB

**Используемые таблицы:** swlfTransceiverType — 1.3.6.1.4.1.89.43.1.1.7

```
snmpwalk -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.43.1.1.7.{ifindex}
```

### **Пример просмотра среды передачи порта TenGigabitEthernet 1/0/23**

Команда CLI:

```
show interfaces status TenGigabitEthernet 1/0/1
```

Команда SNMP:

```
snmpwalk -v2c -c public 192.168.1.30 \
1.3.6.1.4.1.89.43.1.1.7.23
```



### Расшифровка выдаваемых значений

**Copper (1)**

**FiberOptics (2)**

**ComboCopper (3)**

**ComboFiberOptics (4)**

## Управление потоком (flowcontrol)

**MIB:** RADLAN-rlInterfaces

**Используемые таблицы:** swlfFlowControlMode — 1.3.6.1.4.1.89.43.1.1.14

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.43.1.1.14.{ifindex} i {on(1),off(2),auto (3)}
```

### **Пример включения управления потоком на интерфейсе TenGigabitEthernet 1/0/23**

Команда CLI:

```
interface TenGigabitEthernet 1/0/23
flowcontrol on
```

Команда SNMP:

```
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.89.43.1.1.14.23 i 1
```

## Просмотр административного состояния порта

**MIB:** IF-MIB

**Используемые таблицы:** ifAdminStatus — 1.3.6.1.2.1.2.2.1.7

```
snmpwalk -v2c -c <community> <IP address> \
1.3.6.1.2.1.2.2.1.7.{ifIndex}
```

### **Пример просмотра статуса порта TenGigabitEthernet 1/0/23**

Команда CLI:

```
show interfaces status TenGigabitEthernet 1/0/23
```

Команда SNMP:

```
snmpwalk -v2c -c public 192.168.1.30 \
1.3.6.1.2.1.2.2.1.7.23
```



### Возможные варианты

up(1)  
down(2)  
testing(3)

### Включить/выключить конфигурируемый интерфейс

**MIB:** IF-MIB

**Используемые таблицы:** ifAdminStatus — 1.3.6.1.2.1.2.2.1.7

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.2.1.2.2.1.7.{ifIndex} i {up(1),down(2)}
```

#### Пример

Команда CLI:  
interface TenGigabitEthernet 1/0/23  
shutdown

Команда SNMP:  
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.2.1.2.2.1.7.23 i 2

### Включить/выключить расщепление на конфигурируемом интерфейсе

**MIB:** eltMesHardwareMib

**Используемые таблицы:** eltHardwareInterfaceEntry — 1.3.6.1.4.1.35265.1.23.14.1.1.1.3.1

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.35265.1.23.14.1.1.1.3.1.2.{ifIndex} i {4*25g(3),1*100g(1)}
```

#### Пример включения расщепления порта HundredGigabitEthernet1/0/2 для коммутатора MES5500-32

Команда CLI:  
interface HundredGigabitEthernet1/0/2  
hardware profile portmode 4x25g

Команда SNMP:  
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.35265.1.23.14.1.1.1.3.1.2.1 i 3

### Просмотр фактического состояния расщепления порта

**MIB:** eltMesHardwareMib

**Используемые таблицы:** eltHardwareInterfaceEntry — 1.3.6.1.4.1.35265.1.23.14.1.1.1.3.1

```
snmpwalk -v2c -c <community> <IP address> \
1.3.6.1.4.1.35265.1.23.14.1.1.1.3.1.3.{ifIndex}
```

## Пример просмотра статуса расщепления порта HundredGigabitEthernet1/0/2 для коммутатора MES5500-32

Команда CLI:

```
show hardware profile portmode
```

Команда SNMP:

```
snmpwalk -v2c -c public 192.168.1.30 \
1.3.6.1.4.1.35265.1.23.14.1.1.1.3.1.3.2
```



### Возможные варианты

**1\*100g(1)**

**4\*25g(3)**

### Просмотр состояния расщепления порта после перезагрузки

MIB: eltMesHardwareMib

Используемые таблицы: eltHardwareInterfaceEntry — 1.3.6.1.4.1.35265.1.23.14.1.1.1.3.1

```
snmpwalk -v2c -c <community> <IP address> \
1.3.6.1.4.1.35265.1.23.14.1.1.1.3.1.2.{ifIndex}
```

## Пример просмотра статуса расщепления порта HundredGigabitEthernet1/0/2 после перезагрузки для коммутатора MES5500-32

Команда CLI:

```
show hardware profile portmode
```

Команда SNMP:

```
snmpwalk -v2c -c public 192.168.1.30 \
1.3.6.1.4.1.35265.1.23.14.1.1.1.3.1.2.2
```



### Возможные варианты

**1\*100g(1)**

**4\*25g(3)**

### Просмотр оперативного состояния порта

MIB: IF-MIB

Используемые таблицы: ifOperStatus — 1.3.6.1.2.1.2.2.1.8

```
snmpwalk -v2c -c <community> <IP address> \
1.3.6.1.2.1.2.2.1.8.{ifIndex}
```

## Пример просмотра статуса порта TenGigabitEthernet 1/0/23

Команда CLI:

```
show interfaces status TenGigabitEthernet 1/0/23
```

Команда SNMP:

```
snmpwalk -v2c -c public 192.168.1.30 \
1.3.6.1.2.1.2.2.1.8.23
```



### Возможные варианты

up(1)  
down(2)

### Определение типа подключения порта

**MIB:** rlintefaces.mib

**Используемые таблицы:** swIfTransceiverType — 1.3.6.1.4.1.89.43.1.1.7

```
snmpwalk -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.43.1.1.7.{ifIndex}
```

### **Пример определения типа порта TenGigabitethernet 1/0/23**

Команда CLI:

```
show interfaces status
```

Команда SNMP:

```
snmpwalk -v2c -c public 192.168.1.30 \
1.3.6.1.4.1.89.43.1.1.7.23
```



### Возможные варианты

regular (1)  
fiberOptics (2)  
comboRegular (3)  
comboFiberOptics (4)

### Просмотр счетчика unicast-пакетов на интерфейсе

**MIB:** IF-MIB

**Используемые таблицы:** ifInUcastPkts — 1.3.6.1.2.1.2.2.1.11

```
snmpwalk -v2c -c <community> <IP address> \
1.3.6.1.2.1.2.2.1.11.{ifIndex}
```

### **Пример просмотра счетчика входящих unicast-пакетов на интерфейсе TenGigabitethernet 1/0/23**

Команда CLI:

```
show interface counters TenGigabitethernet 1/0/23
```

Команда SNMP:

```
snmpwalk -v2c -c public 192.168.1.30 \
1.3.6.1.2.1.2.2.1.11.23
```

### Просмотр счетчика multicast-пакетов на интерфейсе

**MIB:** IF-MIB

**Используемые таблицы:** ifInMulticastPkts — 1.3.6.1.2.1.31.1.1.1.2

```
snmpwalk -v2c -c <community> <IP address> \
1.3.6.1.2.1.31.1.1.1.2.{ifindex}
```



### Пример просмотра счетчика входящих multicast-пакетов на интерфейсе TenGigabitEthernet 1/0/23

Команда CLI:

```
show interface counters TenGigabitEthernet 1/0/23
```

Команда SNMP:

```
snmpwalk -v2c -c public 192.168.1.30 \
1.3.6.1.2.1.31.1.1.1.2.23
```

### Просмотр счетчика broadcast-пакетов на интерфейсе

MIB: IF-MIB

Используемые таблицы: ifInBroadcastPkts — 1.3.6.1.2.1.31.1.1.1.3

```
snmpwalk -v2c -c <community> <IP address> \
1.3.6.1.2.1.31.1.1.1.3.{ifindex}
```

### Пример просмотра счетчика входящих broadcast-пакетов на интерфейсе TenGigabitEthernet 1/0/23

Команда CLI:

```
show interface counters TenGigabitEthernet 1/0/23
```

Команда SNMP:

```
snmpwalk -v2c -c public 192.168.1.30 \
1.3.6.1.2.1.31.1.1.1.3.23
```

### Просмотр счетчиков октетов на интерфейсе

MIB: IF-MIB

Используемые таблицы:

ifInOctets — 1.3.6.1.2.1.2.2.1.10

ifHCInOctets — 1.3.6.1.2.1.31.1.1.1.6

ifOutOctets — 1.3.6.1.2.1.2.2.1.16

ifHCOctets — 1.3.6.1.2.1.31.1.1.1.10

```
snmpwalk -v2c -c <community> <IP address> \
1.3.6.1.2.1.2.2.1.10.{ifindex}
```

### Пример просмотра счетчика принятых октетов на интерфейсе TenGigabitEthernet 1/0/23

Команда CLI:

```
show interface counters TenGigabitEthernet 1/0/23
```

Команда SNMP:

```
snmpwalk -v2c -c public 192.168.1.30 \
1.3.6.1.2.1.2.2.1.10.23
```



Под октетом имеется в виду количество байт.

1 октет = 1 байт

### Просмотр счетчика FCS Errors на интерфейсе

MIB: EtherLike-MIB

Используемые таблицы: dot3StatsFCSErrors — 1.3.6.1.2.1.10.7.2.1.3

```
snmpwalk -v2c -c <community> <IP address> \  
1.3.6.1.2.1.10.7.2.1.3.{ifindex}
```

#### **Пример просмотра счетчика FCS Errors на интерфейсе TenGigabitEthernet 1/0/23**

Команда CLI:

```
show interface counters TenGigabitEthernet 1/0/23
```

Команда SNMP:

```
snmpwalk -v2c -c public 192.168.1.30 \  
1.3.6.1.2.1.10.7.2.1.3.23
```

#### **Просмотр счетчика Internal MAC Rx Errors на интерфейсе**

**MIB:** EtherLike-MIB

**Используемые таблицы:** dot3StatsInternalMacReceiveErrors — 1.3.6.1.2.1.10.7.2.1.16

```
snmpwalk -v2c -c <community> <IP address> \  
1.3.6.1.2.1.10.7.2.1.16.{ifindex}
```

#### **Пример просмотра счетчика Internal MAC Rx Errors на интерфейсе TenGigabitEthernet 1/0/23**

Команда CLI:

```
show interface counters TenGigabitEthernet 1/0/23
```

Команда SNMP:

```
snmpwalk -v2c -c public 192.168.1.30 \  
1.3.6.1.2.1.10.7.2.1.16.23
```

#### **Просмотр счетчика Transmitted Pause Frames на интерфейсе**

**MIB:** EtherLike-MIB

**Используемые таблицы:** dot3OutPauseFrames — 1.3.6.1.2.1.10.7.10.1.4

```
snmpwalk -v2c -c <community> <IP address> \  
1.3.6.1.2.1.10.7.10.1.4.{ifindex}
```

#### **Пример просмотра счетчика Transmitted Pause Frames на интерфейсе TenGigabitEthernet 1/0/23**

Команда CLI:

```
show interface counters TenGigabitEthernet 1/0/23
```

Команда SNMP:

```
snmpwalk -v2c -c public 192.168.1.30 \  
1.3.6.1.2.1.10.7.10.1.4.23
```

#### **Просмотр счетчика Received Pause Frames на интерфейсе**

**MIB:** EtherLike-MIB

**Используемые таблицы:** dot3InPauseFrames — 1.3.6.1.2.1.10.7.10.1.3

```
snmpwalk -v2c -c <community> <IP address> \  
1.3.6.1.2.1.10.7.10.1.3.{ifindex}
```

### Пример просмотра счетчика Received Pause Frames на интерфейсе TenGigabitEthernet 1/0/23

Команда CLI:

```
show interface counters TenGigabitEthernet 1/0/23
```

Команда SNMP:

```
snmpwalk -v2c -c public 192.168.1.30 \
1.3.6.1.2.1.10.7.10.1.3.23
```

### Очистка счетчиков интерфейсов

MIB: rllInterfaces.mib

Используемые таблицы: rllfClearPortMibCounters — 1.3.6.1.4.1.89.54.4

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.54.4.0 x {битовая маска}
```

### Пример очистки счетчика интерфейсов

Команда CLI:

```
clear counters
```

Команда SNMP:

```
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.89.54.4.0 x
FFFFFFFF000000FFFFFFFF000000FFFFFFFF000000FFFFFFFF000000FFFFFFFF000000F
FFFFFFFF000000FFFFFFFF00000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000
00000000000001FFFFFFFFE000000
```



- 1) В значении очистки счетчиков в стеке задается битовая маска для всех портов всех юнитов стека:

```
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.89.54.4.0 x 000000000000FFFFFFFF000000000F0000000000000000F
000000000000000000000000000000000000000000000000000000000000000000
00000000F0000000000000000000000000000000000000000000000000000000
0000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000
```

- 2) Посмотреть значение битовой маски можно командой:

```
snmpwalk -v2c -c public 192.168.1.30 \
1.3.6.1.4.1.89.54.9.0
```

### Мониторинг загрузки портов коммутатора

MIB: eltMes.mib

Используемые таблицы: eltSwIfUtilizationEntry — 1.3.6.1.4.1.35265.1.23.43.2.1

```
snmpwalk -v2c -c <community> <IP address> \
1.3.6.1.4.1.35265.1.23.43.2.1.{parametr}
```

### Пример

Команда CLI:

```
show interfaces utilization
```

Команда SNMP:

```
snmpwalk -v2c -c public 192.168.1.30 \
1.3.6.1.4.1.35265.1.23.43.2.1.1
```



#### Список возможных параметров

```
eltSwIfUtilizationIfIndex(1)
eltSwIfUtilizationAverageTime(2)
eltSwIfUtilizationCurrentInPkts(3)
eltSwIfUtilizationCurrentInRate(4)
eltSwIfUtilizationCurrentOutPkts(5)
eltSwIfUtilizationCurrentOutRate(6)
eltSwIfUtilizationAverageInPkts(7)
eltSwIfUtilizationAverageInRate(8)
eltSwIfUtilizationAverageOutPkts(9)
eltSwIfUtilizationAverageOutRate(10)
```

#### Включение/выключение режима однонаправленной передачи порта

**MIB:** ELTEX-MES-eltInterfaces

**Используемые таблицы:** eltSwIfTable — 1.3.6.1.4.1.35265.1.23.43.1

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.35265.1.23.43.1.1.15.{index} i {disable(0), send-only(1)}
```

#### **Пример включения режима однонаправленной передачи порта**

Команда CLI:

```
interface TenGigabitEthernet1/0/1
unidirectional send-only
exit
```

Команда SNMP:

```
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.35265.1.23.43.1.1.15.1 i 1
```

## 6.2 Группы агрегации каналов — Link Aggregation Group (LAG)

#### Включение/выключение работы интерфейса в составе группы агрегации

**MIB:** IEEE8023-LAG-MIB

**Используемые таблицы:**

dot3adAggPortTable — 1.2.840.10006.300.43.1.2.1

```
snmpset -v2c -c <community> <IP address> \
1.2.840.10006.300.43.1.2.1.1.20.{ifIndex} x {auto and long timeout(A2), auto and
short timeout(E2), on(22)}\
1.2.840.10006.300.43.1.2.1.1.4.{ifIndex} i {ifIndex}
```





```
rldot1qPortVlanStaticUntaggedEgressList1to1024 —
1.3.6.1.4.1.89.48.68.1.5.{ifindex}
rldot1qPortVlanStaticUntaggedEgressList1025to2048
1.3.6.1.4.1.89.48.68.1.6.{ifindex}
rldot1qPortVlanStaticUntaggedEgressList2049to3072
1.3.6.1.4.1.89.48.68.1.7.{ifindex}
rldot1qPortVlanStaticUntaggedEgressList3073to4094
1.3.6.1.4.1.89.48.68.1.8.{ifindex}
```

2. Пример составления битовой маски приведен в разделе «Приложение А. Методика расчета битовой маски».

3. Битовая маска должна включать в себя не менее 10 символов.

### Запретить default VLAN на порту

MIB: eltVlan.mib

Используемые таблицы: eltVlanDefaultForbiddenPorts — 1.3.6.1.4.1.35265.1.23.5.5.1

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.35265.1.23.5.5.1.0 x {порт в виде битовой маски}
```

#### Пример запрета default vlan на порту TenGigabitEthernet 1/0/23

Команда CLI:

```
interface TenGigabitEthernet 1/0/23
switchport forbidden default-vlan
```

Команда SNMP:

```
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.35265.1.23.5.5.1.0 x 0000020000
```



1. Пример составления битовой маски приведен в разделе «Приложение А. Методика расчета битовой маски».

2. Битовая маска должна включать в себя не менее 10 символов.

### Просмотр имени VLAN

MIB: rlvlan.mib

Используемые таблицы: rldot1qVlanStaticName — 1.3.6.1.4.1.89.48.70.1.1

```
snmpwalk -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.48.70.1.1.{vlan}
```

#### Пример просмотра имени vlan 994

Команда CLI:

```
show vlan tag 994
```

Команда SNMP:

```
snmpwalk -v2c -c public 192.168.1.30 \
1.3.6.1.4.1.89.48.70.1.1.994
```

## Просмотр членства порта во VLAN

**MIB:** rlvlan.mib

**Используемые таблицы:** rldot1qPortVlanStaticTable — 1.3.6.1.4.1.89.48.68

```
snmpwalk -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.48.68.1.{1-4}.{ifindex}
snmpwalk -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.48.68.1.{5-8}.{ifindex}
```

### Пример просмотра VLAN на TenGigabitEthernet 1/0/23

**Команда CLI:**

```
show interfaces switchport TenGigabitEthernet 1/0/23
```

**Команда SNMP:**

```
snmpwalk -v2c -c public 192.168.1.30 \
1.3.6.1.4.1.89.48.68.1.1.23
snmpwalk -v2c -c public 192.168.1.30 \
1.3.6.1.4.1.89.48.68.1.5.23
```



1. В примере представлены 2 команды snmpwalk. Если порт Tagged — значения в выводе второй команды принимают нулевое значение и номер VLAN соответствует значениям вывода первой команды. Если порт Untagged — в выводе второй команды присутствуют значения, отличные от нуля, и номер VLAN соответствует этим значениям.

#### 2. Перечень таблиц:

```
rldot1qPortVlanStaticEgressList1to1024 — 1.3.6.1.4.1.89.48.68.1.1.{ifindex}
rldot1qPortVlanStaticEgressList1025to2048 — 1.3.6.1.4.1.89.48.68.1.2.{ifindex}
rldot1qPortVlanStaticEgressList2049to3072 — 1.3.6.1.4.1.89.48.68.1.3.{ifindex}
rldot1qPortVlanStaticEgressList3073to4094 — 1.3.6.1.4.1.89.48.68.1.4.{ifindex}
rldot1qPortVlanStaticUntaggedEgressList1to1024 —
1.3.6.1.4.1.89.48.68.1.5.{ifindex}
rldot1qPortVlanStaticUntaggedEgressList1025to2048
— 1.3.6.1.4.1.89.48.68.1.6.{ifindex}
rldot1qPortVlanStaticUntaggedEgressList2049to3072
— 1.3.6.1.4.1.89.48.68.1.7.{ifindex}
rldot1qPortVlanStaticUntaggedEgressList3073to4094
— 1.3.6.1.4.1.89.48.68.1.8.{ifindex}
```

3. Полученные в результате выполнения запроса значения представляют из себя битовую маску, методика расчета которой приведена в разделе «Приложение А. Методика расчета битовой маски».

## Настройка режима работы порта

**MIB:** rlvlan.mib

**Используемые таблицы:** vlanPortModeEntry — 1.3.6.1.4.1.89.48.22.1

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.48.22.1.1.{ifIndex} i {general(1), access(2), trunk(3),
customer(7)}
```



### Пример настройки интерфейса TenGigabitEthernet 1/0/23 в режим trunk

Команда CLI:

```
interface TenGigabitEthernet 1/0/23
switchport mode trunk
```

Команда SNMP:

```
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.89.48.22.1.1.21 i 2
```

### Просмотр режима порта

**MIB:** rlvlan.mib

**Используемые таблицы:** vlanPortModeState — 1.3.6.1.4.1.89.48.22.1

```
snmpwalk -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.48.22.1.1.{ifindex}
```

### Пример просмотра режима на TenGigabitEthernet 1/0/23

Команда CLI:

```
show interfaces switchport TenGigabitEthernet 1/0/23
```

Команда SNMP:

```
snmpwalk -v2c -c public 192.168.1.30 \
1.3.6.1.4.1.89.48.22.1.1.23
```



### Возможные варианты

**general(1)**  
**access(2)**  
**trunk (3)**  
**customer (7)**

### Назначить pvid на интерфейс

**MIB:** Q-BRIDGE-MIB.mib

**Используемые таблицы:** dot1qPortVlanTable — 1.3.6.1.2.1.17.7.1.4.5

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.2.1.17.7.1.4.5.1.1.{ifindex} u {1-4094}
```

### Пример назначения pvid 15 для TenGigabitEthernet 1/0/23

Команда CLI:

```
interface TenGigabitEthernet 1/0/23
switchport general pvid 15
```

Команда SNMP:

```
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.2.1.17.7.1.4.5.1.1.23 u 15
```

## Настройка map mac

**MIB:** rlvlan.mib

**Используемые таблицы:** vlanMacBaseVlanGroupTable — 1.3.6.1.4.1.89.48.45

```
snmpset -v2c -c <community> <IP address> \  
1.3.6.1.4.1.89.48.45.1.3.{MAC address in DEC}.{mask} i {map-group number} \  
1.3.6.1.4.1.89.48.45.1.4.{MAC address in DEC}.{mask} i {createAndGo(4),  
destroy(6)}
```

### **Пример**

**Команда CLI:**

```
vlan database  
map mac a8:f9:4b:33:29:c0 32 macs-group 1
```

**Команда SNMP:**

```
snmpset -v2c -c private 192.168.1.30 \  
1.3.6.1.4.1.89.48.45.1.3.168.249.75.51.41.192.32 i 1 \  
1.3.6.1.4.1.89.48.45.1.4.168.249.75.51.41.192.32 i 4
```

## Установка правила классификации VLAN, основанного на привязке к MAC-адресу, для интерфейса

**MIB:** rlvlan.mib

**Используемые таблицы:** vlanMacBaseVlanPortTable — 1.3.6.1.4.1.89.48.46.1.2

```
snmpset -v2c -c <community> <IP address> \  
1.3.6.1.4.1.89.48.46.1.2.58.1 u {vlan} 1.3.6.1.4.1.89.48.46.1.3.58.1 i  
{createAndGo(4), destroy(6)}
```

### **Пример включения правила классификации VLAN для интерфейса TenGigabitEthernet 1/0/23**

**Команда CLI:**

```
interface TenGigabitEthernet 1/0/23  
switchport general map macs-group 1 vlan 20
```

**Команда SNMP:**

```
snmpset -v2c -c private 192.168.1.30 \  
1.3.6.1.4.1.89.48.46.1.2.23.1 u 1 \  
1.3.6.1.4.1.89.48.46.1.3.23.1 i 4
```

## **6.4 Настройка и мониторинг errdisable-состояния**

### Просмотр настроек для автоматической активации интерфейса

**MIB:** rlinterfaces\_recovery.mib

**Используемые таблицы:** rlErrdisableRecoveryEnable — 1.3.6.1.4.1.89.128.2.1.2

```
snmpwalk -v2c -c <community> <IP address> \  
1.3.6.1.4.1.89.128.2.1.2
```

### Пример просмотра настроек для автоматической активации интерфейса

Команда CLI:

```
show errdisable recovery
```

Команда SNMP:

```
snmpwalk -v2c -c public 192.168.1.30 \
1.3.6.1.4.1.89.128.2.1.2
```

### Просмотр причины блокировки порта

MIB: rErrdisableRecoveryIfReason

Используемые таблицы: rErrdisableRecoveryIfReason — 1.3.6.1.4.1.89.128.3.1.1

```
snmpwalk -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.128.3.1.1
```

### Пример

Команда CLI:

```
show errdisable interfaces
```

Команда SNMP:

```
snmpwalk -v2c -c public 192.168.1.30 \
1.3.6.1.4.1.89.128.3.1.1
```



### Возможные варианты:

- loopback-detection (1)
- port-security (2)
- dot1x-src-address (3)
- acl-deny (4)
- stp-bpdu-guard (5)
- stp-loopback-guard (6)
- unidirectional-link (7)
- dhcp-rate-limit (8)
- l2pt-guard (9)
- storm-control (10)

### Настройка автоматической активации интерфейса

MIB: rlintefaces\_recovery.mib

Используемые таблицы: rErrdisableRecoveryEnable — 1.3.6.1.4.1.89.128.2.1.2

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.128.2.1.2. {index of reason} i {true(1), false(2)}
```

### Пример включения автоматической активации интерфейса в случае loopback detection

Команда CLI:

```
errdisable recovery cause loopback-detection
```

Команда SNMP:

```
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.89.128.2.1.2.1 i 1
```



**Возможные значения *index of reason*, в зависимости от типа выполняемой настройки:**

loopback detection — (1)  
 port-security — (2)  
 dot1x-src-address — (3)  
 acl-deny — (4)  
 stp-bpdu-guard — (5)  
 stp-loopback-guard (6)  
 unidirectional-link — (8)  
 storm-control — (9)  
 l2pt-guard — (11)

**Настройка интервала выхода интерфейса из errdisable состояния**

**MIB:** rlinterfaces\_recovery.mib

**Используемые таблицы:** rIErrdisableRecoveryInterval — 1.3.6.1.4.1.89.128.1

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.128.1.0 i {interval 30-86400}
```

**Пример настройки 30 секундного интервала выхода из состояния errdisable**

Команда CLI:  
errdisable recovery interval 30

Команда SNMP:  
snmpset -v2c -c private 192.168.1.30 \  
1.3.6.1.4.1.89.128.1.0 i 30

## 6.5 Настройка voice vlan

**Добавление voice vlan**

**MIB:** RADLAN-vlanVoice-MIB

**Используемые таблицы:** vlanVoiceAdminVid — 1.3.6.1.4.1.89.48.54.8

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.48.54.8.0 i {vlan id}
```

**Пример добавления voice vlan id 10**

Команда CLI:  
voice vlan id 10

Команда SNMP:  
snmpset -v2c -c private 192.168.1.30 \  
1.3.6.1.4.1.89.48.54.8.0 i 10

## Активация voice vlan на интерфейсе

**MIB:** RADLAN-vlanVoice-MIB

**Используемые таблицы:** vlanVoiceOUIBasedPortTable — 1.3.6.1.4.1.89.48.54.12.5

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.48.54.12.5.1.1.{ifIndex} i 1 \
1.3.6.1.4.1.89.48.54.12.5.1.2.{ifIndex} u {voice vlan id}
```

### Пример

Команда CLI:

```
interface TenGigabitEthernet 1/0/23
voice vlan enable
```

Команда SNMP:

```
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.89.48.54.12.5.1.1.23 i 1 \
1.3.6.1.4.1.89.48.54.12.5.1.2.23 u 10
```

## Редактирование таблицы OUI

**MIB:** rlvlanVoice.mib

**Используемые таблицы:** vlanVoiceOUIBasedTable — 1.3.6.1.4.1.89.48.54.12.4

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.48.54.12.4.1.3.{OUI in DEC. Байты разделяются точками} i
{createAndGo(4), destroy(6)}
```

### Пример

Команда CLI:

```
voice vlan oui-table add 002618
```

Команда SNMP:

```
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.89.48.54.12.4.1.3.0.38.24 i 4
```

## 6.6 Настройка LLDP

### Глобальное включение/отключение LLDP

**MIB:** rLldp.mib

**Используемые таблицы:** rLldpEnabled — 1.3.6.1.4.1.89.110.1.1.1

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.110.1.1.1.0 i {true (1), false (2)}
```

### Пример отключения LLDP

Команда CLI:

```
no Lldp run
```

Команда SNMP:

```
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.89.110.1.1.1.0 i 2
```

### Настройка lldp-med политики с указанием номера voice vlan для тегированного трафика voice vlan

**MIB:** rllldb.mib

**Используемые таблицы:** rLLdpXMedLocMediaPolicyContainerTable — 1.3.6.1.4.1.89.110.1.2.1

```
snmpset -v2c -c <community> <IP address> \  
1.3.6.1.4.1.89.110.1.2.1.1.2.1 i {voice(1), voice-signaling(2), guest-voice(3),  
guest-voice-signaling(4), softphone-voice(5), video-conferencing(6), streaming-  
video(7), video-signaling(8)} \  
1.3.6.1.4.1.89.110.1.2.1.1.3.1 i {vlan} \  
1.3.6.1.4.1.89.110.1.2.1.1.4.1 i {priority} \  
1.3.6.1.4.1.89.110.1.2.1.1.7.1 {true(1), false(2)} \  
1 1.3.6.1.4.1.89.110.1.2.1.1.9.1 i {createAndGo(4), destroy(6)}
```

#### **Пример настройки политики lldp-med с указанием VLAN 10, указанием приоритета 4**

**Команда CLI:**

```
lldp med network-policy 1 voice vlan 10 vlan-type tagged up 4
```

**Команда SNMP:**

```
snmpset -v2c -c private 192.168.1.30 \  
1.3.6.1.4.1.89.110.1.2.1.1.2.1 i 1 \  
1.3.6.1.4.1.89.110.1.2.1.1.3.1 i 10 \  
1.3.6.1.4.1.89.110.1.2.1.1.4.1 i 4 \  
1.3.6.1.4.1.89.110.1.2.1.1.7.1 i 1 \  
1.3.6.1.4.1.89.110.1.2.1.1.9.1 i 4
```

### Настройка lldp-med политики для тегированного трафика voice vlan

**MIB:** rllldb.mib

**Используемые таблицы:** rLLdpXMedNetPolVoiceUpdateMode — 1.3.6.1.4.1.89.110.1.7

```
snmpset -v2c -c <community> <IP address> \  
1.3.6.1.4.1.89.110.1.7.0 i {manual(0), auto(1)}
```

#### **Пример настройки политики lldp-med в режиме auto**

**Команда CLI:**

```
no lldp med network-policy voice auto
```

**Команда SNMP:**

```
snmpset -v2c -c private 192.168.1.30 \  
1.3.6.1.4.1.89.110.1.7.0 i 1
```

## 7 НАСТРОЙКА IPV4-АДРЕСАЦИИ

### Создание IP-адреса на interface vlan

**MIB:** rlip.mib

**Используемые таблицы:** rslpAddrEntry — 1.3.6.1.4.1.89.26.1.1

```
snmpset -v2c -c <community> <IP address> \
  1.3.6.1.4.1.89.26.1.1.2.{ip address(DEC)} i {ifIndex} \
  1.3.6.1.4.1.89.26.1.1.3.{ip address(DEC)} a {netmask}
```

#### **Пример настройки IP-адреса 192.168.10.30/24 на vlan 30**

**Команда CLI:**

```
interface vlan 30
ip address 192.168.10.30 /24
```

**Команда SNMP:**

```
snmpset -v2c -c private 192.168.1.30 \
  1.3.6.1.4.1.89.26.1.1.2.192.168.10.30 i 100029 \
  1.3.6.1.4.1.89.26.1.1.3.192.168.10.30 a 255.255.255.0
```

### Удаление IP-адреса на interface vlan

**MIB:** rlip.mib

**Используемые таблицы:** rslpAddrEntry — 1.3.6.1.4.1.89.26.1.1

```
snmpset -v2c -c <community> <IP address> \
  1.3.6.1.4.1.89.26.1.1.2.{ip address(DEC)} i {ifIndex} \
  1.3.6.1.4.1.89.26.1.1.3.{ip address(DEC)} a {netmask} \
  1.3.6.1.4.1.89.26.1.1.6.{ip address(DEC)} i 2
```

#### **Пример удаления IP-адреса 192.168.10.30 на интерфейсе vlan 30**

**Команда CLI:**

```
interface vlan 30
no ip address 192.168.10.30
```

**Команда SNMP:**

```
snmpset -v2c -c private 192.168.1.30 \
  1.3.6.1.4.1.89.26.1.1.2.192.168.10.30 i 100029 \
  1.3.6.1.4.1.89.26.1.1.3.192.168.10.30 a 255.255.255.0 \
  1.3.6.1.4.1.89.26.1.1.6.192.168.10.30 i 2
```

### Получение IP-адреса по DHCP на interface vlan

**MIB:** radlan-dhcpcl-mib.mib

**Используемые таблицы:** rIDhcpClActionStatus — 1.3.6.1.4.1.89.76.3.1.2

```
snmpset -v2c -c <community> <IP address> \
  1.3.6.1.4.1.89.76.3.1.2.{ifIndex} i {createAndGo(4), destroy(6)}
```

### Пример

**Команда CLI:**

```
interface vlan 30
 ip address dhcp
```

**Команда SNMP:**

```
snmpset -v2c -c private 192.168.1.30 \1.3.6.1.4.1.89.76.3.1.2.100029 i 4
```

### Добавить/удалить шлюз по умолчанию

**MIB:** rlip.mib

**Используемые таблицы:** rllnetStaticRouteEntry — 1.3.6.1.4.1.89.26.28.1

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.26.28.1.4.0.0.0.0.0.1.4.{IP address}.0 i {metric(4)} \
1.3.6.1.4.1.89.26.28.1.4.0.0.0.0.0.1.4.{IP address}.0 i {remote(4)} \
1.3.6.1.4.1.89.26.28.1.4.0.0.0.0.0.1.4.{IP address}.0 i {createAndGo (4),
destroy(6)}
```

### Пример добавления ip default-gateway 192.168.1.10

**Команда CLI:**

```
ip default-gateway 192.168.1.10
```

**Команда SNMP:**

```
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.89.26.28.1.7.1.4.0.0.0.0.0.1.4.192.168.1.10.0 u 4 \
1.3.6.1.4.1.89.26.28.1.8.1.4.0.0.0.0.0.1.4.192.168.1.10.0 i 4 \
1.3.6.1.4.1.89.26.28.1.10.1.4.0.0.0.0.0.1.4.192.168.1.10.0 i 4
```





## 9 НАСТРОЙКА GREEN ETHERNET

### Глобальное отключение green-ethernet short-reach

**MIB:** rlgreeneth.mib

**Используемые таблицы:** rlGreenEthShortReachEnable — 1.3.6.1.4.1.89.134.2

```
snmpset -v2c -c <community> <IP address> \  
1.3.6.1.4.1.89.134.2.0 i {true (1), false (2)}
```

#### **Пример отключения green-ethernet short-reach**

Команда CLI:

```
no green-ethernet short-reach
```

Команда SNMP:

```
snmpset -v2c -c private 192.168.1.30 \  
1.3.6.1.4.1.89.134.2.0 i 2
```

### Глобальное отключение green-ethernet energy-detect

**MIB:** rlgreeneth.mib

**Используемые таблицы:** rlGreenEthEnergyDetectEnable — 1.3.6.1.4.1.89.134.1

```
snmpset -v2c -c <community> <IP address> \  
1.3.6.1.4.1.89.134.1.0 i {true (1), false (2)}
```

#### **Пример отключения green-ethernet energy-detect**

Команда CLI:

```
no green-ethernet energy-detect
```

Команда SNMP:

```
snmpset -v2c -c private 192.168.1.30 \  
1.3.6.1.4.1.89.134.1.0 i 2
```

### Просмотр параметров green-ethernet

**MIB:** rlGreenEth.mib

**Используемые таблицы:** rlGreenEthCumulativePowerSaveMeter — 1.3.6.1.4.1.89.134.5

```
snmpwalk -v2c -c <community> <IP address> \  
1.3.6.1.4.1.89.134.5
```

#### **Пример просмотра параметров green-ethernet**

Команда CLI:

```
show green-ethernet
```

Команда SNMP:

```
snmpwalk -v2c -c public 192.168.1.30 \  
1.3.6.1.4.1.89.134.5
```

## 10 НАСТРОЙКА КОЛЬЦЕВЫХ ПРОТОКОЛОВ

### 10.1 Протокол ERPS

#### Определение номера west-порта

**MIB:** ELTEX-BRIDGE-ERPS-V2-MIB.mib

**Используемые таблицы:** eltexErpsMgmtRAPSWestPort — 1.3.6.1.4.1.35265.35.1.1.3.1.1.2

```
snmpwalk -v2c -c <community> <IP address> \
1.3.6.1.4.1.35265.35.1.1.3.1.1.2
```

#### Пример

Команда CLI:

```
show erps
```

Команда SNMP:

```
snmpwalk -v2c -c public 192.168.1.30 \
1.3.6.1.4.1.35265.35.1.1.3.1.1.2
```

#### Просмотр состояния west-порта

**MIB:** ELTEX-BRIDGE-ERPS-V2-MIB.mib

**Используемые таблицы:** eltexErpsMgmtRAPSWestPortState — 1.3.6.1.4.1.35265.35.1.1.3.1.1.3

```
snmpwalk -v2c -c <community> <IP address> \
1.3.6.1.4.1.35265.35.1.1.3.1.1.3
```

#### Пример

Команда CLI:

```
show erps vlan 10
```

Команда SNMP:

```
snmpwalk -v2c -c public 192.168.1.30 \
1.3.6.1.4.1.35265.35.1.1.3.1.1.3
```



#### Возможные состояния порта:

1. Forwarding (1)
2. Blocking (2)
3. Signal-fail (3)
4. Manual-switch (4)
5. Forced-switch (5)

#### Определение номера east-порта

**MIB:** ELTEX-BRIDGE-ERPS-V2-MIB.mib

**Используемые таблицы:** eltexErpsMgmtRAPSEastPort — 1.3.6.1.4.1.35265.35.1.1.3.1.1.4

```
snmpwalk -v2c -c <community> <IP address> \
1.3.6.1.4.1.35265.35.1.1.3.1.1.4
```

### Пример

Команда CLI:

```
show erps
```

Команда SNMP:

```
snmpwalk -v2c -c public 192.168.1.30 \
1.3.6.1.4.1.35265.35.1.1.3.1.1.4
```

### Просмотр состояния east-порта

**MIB:** ELTEX-BRIDGE-ERPS-V2-MIB.mib

**Используемые таблицы:** eltexErpsMgmtRAPSEastPortState — 1.3.6.1.4.1.35265.35.1.1.3.1.1.5

```
snmpwalk -v2c -c <community> <IP address> \
1.3.6.1.4.1.35265.35.1.1.3.1.1.5
```

### Пример

Команда CLI:

```
show erps vlan 10
```

Команда SNMP:

```
snmpwalk -v2c -c public 192.168.1.30 \
1.3.6.1.4.1.35265.35.1.1.3.1.1.5
```



### Возможные состояния порта:

1. Forwarding (1)
2. Blocking (2)
3. Signal-fail (3)
4. Manual-switch (4)
5. Forced-switch (5)

### Просмотр состояния кольца

**MIB:** ELTEX-BRIDGE-ERPS-V2-MIB.mib

**Используемые таблицы:** eltexErpsMgmtRAPSRingState — 1.3.6.1.4.1.35265.35.1.1.3.1.1.12

```
snmpwalk -v2c -c <community> <IP address> \
1.3.6.1.4.1.35265.35.1.1.3.1.1.12
```

### Пример

Команда CLI:

```
show erps vlan 10
```

Команда SNMP:

```
snmpwalk -v2c -c public 192.168.1.30 \
1.3.6.1.4.1.35265.35.1.1.3.1.1.12
```



### Возможные состояния кольца erps:

1. Init (1)
2. Idle(2)
3. Protection (3)
4. Manual-switch (4)
5. Forced-switch (5)
6. Pending (6)

## 10.2 Настройка протокола Spanning Tree

### Включение/отключение протокола spanning-tree

**MIB:** radlan-brgmacswitch.mib

**Используемые таблицы:** rldot1dStp — 1.3.6.1.4.1.89.57.2.3

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.57.2.3.0 i {enabled(1), disabled(2)}
```

#### Пример отключения spanning-tree

Команда CLI:

```
no spanning-tree
```

Команда SNMP:

```
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.89.57.2.3.0 i 2
```

### Включение/отключение протокола spanning-tree на конфигурируемом интерфейсе

**MIB:** BRIDGE-MIB

**Используемые таблицы:** dot1dStpPortTable — 1.3.6.1.2.1.17.2.15.1.4

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.2.1.17.2.15.1.4.{ifIndex} i {enabled(1), disabled(2)}
```

#### Пример отключения работы spanning-tree на интерфейсе TenGigabitEthernet 1/0/23

Команда CLI:

```
interface TenGigabitEthernet 1/0/23
spanning-tree disable
```

Команда SNMP:

```
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.2.1.17.2.15.1.4.23 i 2
```

### Включение/выключение режима обработки пакетов BPDU интерфейсом, на котором выключен протокол STP

**MIB:** radlan-bridgemibobjects-mib.mib

**Используемые таблицы:** rldot1dStpPortTable — 1.3.6.1.4.1.89.57.2.13.1.4

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.57.2.13.1.4.{ifIndex} i {filtering(1), flooding(2)}
```

#### Пример включения фильтрации BPDU на интерфейсе TenGigabitEthernet 1/0/23

Команда CLI:

```
interface tengigabitEthernet 1/0/23
spanning-tree bpdu filterin
```

Команда SNMP:

```
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.89.57.2.13.1.4.23 i 1
```

### Настройка режима работы протокола spanning-tree

**MIB:** draft-ietf-bridge-rstpmib.mib

**Используемые таблицы:** dot1dStpVersion — 1.3.6.1.2.1.17.2.16

```
snmpset -v2c -c <community> <IP address> \  
1.3.6.1.2.1.17.2.16.0 i {stp(0), rstp(2), mstp(3)}
```

#### **Пример установки режима работы протокола Spanning-tree**

Команда CLI:

```
spanning-tree mode rstp
```

Команда SNMP:

```
snmpset -v2c -c private 192.168.1.30 \  
1.3.6.1.2.1.17.2.16.0 i 2
```

### Просмотр роли порта в STP

**MIB:** radlan-bridgemibobjects-mib.mib

**Используемые таблицы:** rldot1dStpPortRole — 1.3.6.1.4.1.89.57.2.13.1.7

```
snmpwalk -v2c -c <community> <IP address> \  
1.3.6.1.4.1.89.57.2.13.1.7.{ifindex}
```

#### **Пример просмотра роли TenGigabitEthernet 1/0/23 в STP**

Команда CLI:

```
show spanning-tree TenGigabitEthernet 1/0/23
```

Команда SNMP:

```
snmpwalk -v2c -c public 192.168.1.30 \  
1.3.6.1.4.1.89.57.2.13.1.7.23
```



#### **Возможные состояния порта:**

- 1. Disabled (1)**
- 2. Alternate (2)**
- 3. Backup(3)**
- 4. Root(4)**
- 5. Designated(5)**

### Просмотр состояния порта в MSTP

**MIB:** radlan-bridgemibobjects-mib.mib

**Используемые таблицы:** rldot1sMstpInstancePortState — 1.3.6.1.4.1.89.57.6.2.1.4

```
snmpwalk -v2c -c <community> <IP address> \  
1.3.6.1.4.1.89.57.6.2.1.4.1.{ifindex}
```

### Пример просмотра состояния TenGigabitEthernet 1/0/23 в mstp

Команда CLI:

```
show spanning-tree TenGigabitEthernet0/23
```

Команда SNMP:

```
snmpwalk -v2c -c public 192.168.1.30 \
1.3.6.1.4.1.89.57.6.2.1.4.1.23
```



#### Возможные состояния порта:

1. Disabled (1)
2. Blocking (2)
3. Listening (3)
4. Forwarding(5)

#### Количество перестроений (topology change)

MIB: BRIDGE-MIB

Используемые таблицы: dot1dStpTopChanges — 1.3.6.1.2.1.17.2.4.0

```
snmpwalk -v2c -c <community> <IP address> \
1.3.6.1.2.1.17.2.4.0
```

### Пример просмотра количества перестроений

Команда CLI:

```
show spanning-tree
```

Команда SNMP:

```
snmpwalk -v2c -c public 192.168.1.30 \
1.3.6.1.2.1.17.2.4.0
```

#### Просмотр времени с последнего перестроения (topology change)

MIB: MIB: BRIDGE-MIB

Используемые таблицы: dot1dStpTimeSinceTopologyChange — 1.3.6.1.2.1.17.2.3.0

```
snmpwalk -v2c -c <community> <IP address> \
1.3.6.1.2.1.17.2.3.0
```

### Пример просмотра с последнего перестроения

Команда CLI:

```
show spanning-tree
```

Команда SNMP:

```
snmpwalk -v2c -c public 192.168.1.30 \
1.3.6.1.2.1.17.2.3.0
```

Просмотр интерфейса, с которого принят последний topology change

**MIB: eltBridgeExtMIB.mib**

**Используемые таблицы:** eltdot1dStpLastTopologyChangePort — 1.3.6.1.4.1.35265.1.23.1.401.0.5.2

`snmpwalk -v2c -c <community> <IP address> \1.3.6.1.4.1.35265.1.23.1.401.0.5.2`

**Пример просмотра интерфейса, с которого принят последний topology change**

Команда CLI:

```
show spanning-tree
```

Команда SNMP:

```
snmpwalk -v2c -c public 192.168.1.30 \  
1.3.6.1.4.1.35265.1.23.1.401.0.5.2
```



## 11 ГРУППОВАЯ АДРЕСАЦИЯ

### 11.1 Правила групповой адресации (multicast addressing)

#### Запрещение динамического добавления порта к многоадресной группе

**MIB:** rlbrgmulticast.mib

**Используемые таблицы:** rIBrgStaticInetMulticastEntry — 1.3.6.1.4.1.89.116.5.1

```
snmpset -v2c -c <community> <IP address> \
  1.3.6.1.4.1.89.116.5.1.6.{vlan id}.1.4.{ip address(DEC)}.1.4.0.0.0.0 x
  0000000000000000 \
  1.3.6.1.4.1.89.116.5.1.7.{vlan id}.1.4.{ip address(DEC)}.1.4.0.0.0.0 x
  {Битовая маска интерфейса} \
  1.3.6.1.4.1.89.116.5.1.8.{vlan id}.1.4.{ip address(DEC)}.1.4.0.0.0.0 i
  {createAndGo(4), destroy (6)}
```

#### **Пример запрета изучения группы 239.200.200.17 на порту TenGigabitEthernet 1/0/23 в vlan 622**

**Команда CLI:**

```
interface vlan 622
bridge multicast forbidden ip-address 239.200.200.17 add TenGigabitEthernet
1/0/23
```

**Команда SNMP:**

```
snmpset -v2c -c private 192.168.1.30 \
  1.3.6.1.4.1.89.116.5.1.6.622.1.4.239.200.200.17.1.4.0.0.0.0 x 0000000000000000 \
  1.3.6.1.4.1.89.116.5.1.7.622.1.4.239.200.200.17.1.4.0.0.0.0 x 0000020000 \
  1.3.6.1.4.1.89.116.5.1.8.622.1.4.239.200.200.17.1.4.0.0.0.0 i 4
```



**1) Суммарное количество цифр в OID 1.3.6.1.4.1.89.116.5.1.6 и OID 1.3.6.1.4.1.89.116.5.1.7 должно быть одинаковым и чётным.**

**2) Методику расчета битовой маски можно посмотреть в разделе «Приложение А. Методика расчета битовой маски».**

#### Запрещение прохождения незарегистрированного Multicast-трафика

**MIB:** rlbrgmulticast.mib

**Используемые таблицы:** rIMacMulticastUnregFilterEnable — 1.3.6.1.4.1.89.55.4.1

```
snmpset -v2c -c <community> <IP address> \
  1.3.6.1.4.1.89.55.4.1.0 x "{Битовая маска для интерфейсов}"
```

#### **Пример запрещения прохождения незарегистрированного Multicast-трафика для портов TenGigabitEthernet 1/0/20-21**

**Команда CLI:**

```
interface range TenGigabitEthernet 1/0/20-21
bridge multicast unregistered filtering
```

**Команда SNMP:**

```
snmpset -v2c -c private 192.168.1.30 \
  1.3.6.1.4.1.89.55.4.1.0 x "000018000000000000"
```



1) Для удаления настройки надо заменить соответствующие портам поля в битовой маске на 0.

2) Методику расчета битовой маски можно посмотреть в разделе «Приложение А. Методика расчета битовой маски».

### Фильтрация многоадресного трафика

**MIB:** rlbmgmulticast.mib

**Используемые таблицы:** rIMacMulticastEnable — 1.3.6.1.4.1.89.55.1

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.55.1.0 i {true(1), false(2)}
```

#### **Пример включения фильтрации многоадресного трафика**

Команда CLI:

```
bridge multicast filtering
```

Команда SNMP:

```
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.89.55.1.0 i 1
```

### Глобальное включение igmp snooping

**MIB:** rlbmgmulticast.mib

**Используемые таблицы:** rllgmpSnoopEnable — 1.3.6.1.4.1.89.55.2.2

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.55.2.2.0 i {true(1), false(2)}
```

#### **Пример**

Команда CLI:

```
ip igmp snooping
```

Команда SNMP:

```
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.89.55.2.2.0 i 1
```

### Включение igmp snooping в vlan

**MIB:** rlbmgmulticast.mib

**Используемые таблицы:** rllgmpMldSnoopVlanEnable — 1.3.6.1.4.1.89.55.5.5.1.3

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.55.5.5.1.3.1.{vlan id} i {true(1), false(2)}
```

#### **Пример включения igmp snooping в vlan 30**

Команда CLI:

```
ip igmp snooping vlan 30
```

Команда SNMP:

```
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.89.55.5.5.1.3.1.30 i 1
```

### Просмотр таблицы igmp snooping

**MIB:** rlbmgmulticast.mib

**Используемые таблицы:** rllgmpMldSnoopMembershipTable — 1.3.6.1.4.1.89.55.5.4

```
snmpwalk -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.55.5.4
```

#### Пример

**Команда CLI:**

```
show ip igmp snooping groups
```

**Команда SNMP:**

```
snmpwalk -v2c -c public 192.168.1.30 \
1.3.6.1.4.1.89.55.5.4
```

### Настройка multicast-tv vlan (MVR)

**MIB:** rlvlan.mib

**Используемые таблицы:** vlanMulticastTvEntry — 1.3.6.1.4.1.89.48.44.1

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.48.44.1.1.{ifIndex} u {vlan-id} \
1.3.6.1.4.1.89.48.44.1.2.50 i {createAndGo(4), destroy (6)}
```

#### Пример настройки multicast-tv vlan 622 на интерфейсе TenGigabitEthernet 1/0/23

**Команда CLI:**

```
interface tengigabitethernet 1/0/23
switchport access multicast-tv vlan 622
```

**Команда SNMP:**

```
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.89.48.44.1.1.23 u 622 \
1.3.6.1.4.1.89.48.44.1.2.23 i 4
```



Настройка режима работы multicast-tv vlan <customer/access/trunk/general> зависит от режима настройки порта, т.е. от команды switchport mode customer/access/trunk/general.

## 11.2 Функции ограничения multicast-трафика

### Создание multicast snooping profile

**MIB:** eltIpMulticast.mib

**Используемые таблицы:** eltMesIpMulticast — 1.3.6.1.4.1.35265.1.23.46.1

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.35265.1.23.46.1.1.2.{Index of profile} s {profile name} \
1.3.6.1.4.1.35265. 1.23.46.1.1.3.{Index of profile} i {deny(1), permit(2)} \
1.3.6.1.4.1.35265. 1.23.46.1.1.4.{Index of profile} i {createAndGo(4),
destroy(6)}
```

**Пример создания профиля с именем IPTV (предположим, что профиль будет иметь порядковый номер 3)**

Команда CLI:

```
multicast snooping profile IPTV
```

Команда SNMP:

```
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.35265.1.23.46.1.1.2.3 s IPTV \
1.3.6.1.4.1.35265.1.23.46.1.1.3.3 i 1 \
1.3.6.1.4.1.35265.1.23.46.1.1.4.3 i 4
```

### Указание диапазонов Multicast-адресов в multicast snooping profile

**MIB:** eltlpMulticast.mib

**Используемые таблицы:** eltMeslpMulticast — 1.3.6.1.4.1.35265. 1.23.46.3

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.35265. 1.23.46.3.1.3.{index of rule}.{Index of profile} i
{ip(1),ipv6(2)} \
1.3.6.1.4.1.35265. 1.23.46.3.1.4.{index of rule}.{Index of profile} x {ip-адрес
начала диапазона в шестнадцатеричном виде} \
1.3.6.1.4.1.35265. 1.23.46.3.1.5.{index of rule}.{Index of profile} x {ip-адрес
конца диапазона в шестнадцатеричном виде} \
1.3.6.1.4.1.35265. 1.23.46.3.1.6.{index of rule}.{Index of profile} i
{createAndGo(4), destroy(6)}
```

**Пример ограничения Multicast-групп 233.7.70.1-233.7.70.10 для профиля с именем IPTV (предположим, что профиль имеет порядковый номер 3. В первом профиле 2 правила, во втором — 1)**

Команда CLI:

```
multicast snooping profile IPTV
match ip 233.7.70.1 233.7.70.10
```

Команда SNMP:

```
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.35265.1.23.46.3.1.3.4.3 i 1 \
1.3.6.1.4.1.35265.1.23.46.3.1.4.4.3 x E9074601 \
1.3.6.1.4.1.35265.1.23.46.3.1.5.4.3 x E907460A \
1.3.6.1.4.1.35265.1.23.46.3.1.6.4.3 i 4
```



*index of rule* — считается по сумме всех правил во всех профилях.

### Назначение multicast snooping profile на порт

**MIB:** eltlpMulticast.mib

**Используемые таблицы:** eltMeslpMulticast — 1.3.6.1.4.1.35265. 1.23.46.7.1

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.35265. 1.23.46.7.1.1.{ifIndex}.{Index of profile} i {ifIndex} \
1.3.6.1.4.1.35265. 1.23.46.7.1.2.{ifIndex}.{Index of profile} i {Index of
profile} \
1.3.6.1.4.1.35265. 1.23.46.7.1.3.{ifIndex}.{Index of profile} i
{createAndGo(4), destroy(6)}
```

**Пример добавления профиля test (с индексом профиля 3) на интерфейс TenGigabitEthernet 1/0/23**

Команда CLI:

```
interface TenGigabitEthernet 1/0/23
 multicast snooping add test
```

Команда SNMP:

```
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.35265.1.23.46.7.1.1.23.3 i 23 \
1.3.6.1.4.1.35265.1.23.46.7.1.2.23.3 i 3 \
1.3.6.1.4.1.35265.1.23.46.7.1.3.23.3 i 4
```

**Настройка ограничения количества Multicast-групп на порту**

MIB: eltlpMulticast.mib

Используемые таблицы: eltMeslpMulticast — 1.3.6.1.4.1.35265.1.23.46.6.1

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.35265.1.23.46.6.1.2.{ifIndex} i {MAX number}
```

**Пример настройки ограничения в три Multicast-группы на интерфейсе TenGigabitEthernet 1/0/23**

Команда CLI:

```
interface TenGigabitEthernet 1/0/23
 multicast snooping max-groups 3
```

Команда SNMP:

```
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.35265.1.23.46.6.1.2.23 i 3
```

## 12 ФУНКЦИИ УПРАВЛЕНИЯ

### 12.1 Механизм AAA

#### Добавление нового пользователя

MIB: rlaaa.mib

Используемые таблицы: rIAAALocalUserTable — 1.3.6.1.4.1.89.79.17

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.79.17.1.1.{number of letters}. {Login in DEC, каждая буква логина
отделяется от следующей точкой} s {login} \
1.3.6.1.4.1.89.79.17.1.2.{number of letters}. {Login in DEC, каждая буква логина
отделяется от следующей точкой} s "#{encoding password}" \
1.3.6.1.4.1.89.79.17.1.3.{number of letters}. {Login in DEC, каждая буква логина
отделяется от следующей точкой} i {privelege level(1-15)} \
1.3.6.1.4.1.89.79.17.1.4.{number of letters}. {Login in DEC, каждая буква логина
отделяется от следующей точкой} i {create and go(4)}
```

#### **Пример добавления пользователя techsup с паролем password и уровнем привилегий 15**

Команда CLI:

```
username techsup password password privilege 15
```

Команда SNMP:

```
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.89.79.17.1.1.7.116.101.99.104.115.117.112 s techsup \
1.3.6.1.4.1.89.79.17.1.2.7.116.101.99.104.115.117.112 s
"#5baa61e4c9b93f3f0682250b6cf8331b7ee68fd8" \
1.3.6.1.4.1.89.79.17.1.3.7.116.101.99.104.115.117.112 i 15
\1.3.6.1.4.1.89.79.17.1.4.7.116.101.99.104.115.117.112 i 4
```



1. Логин переводится из ASCII в HEX с помощью таблицы, которую можно найти по ссылке <https://ru.wikipedia.org/wiki/ASCII>.

2. Пароль задается исключительно в зашифрованном виде, пишется обязательно в кавычках, перед паролем добавляется #.

#### Настройка методов авторизации для login-пользователя

MIB: rlaaa.mib

Используемые таблицы: rIAAMethodListEntry — 1.3.6.1.4.1.89.79.15.1

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.79.15.1.{authindex}.15. {"login_c_default" in DEC, каждая буква
логина отделяется от следующей точкой} i {deny (0),
line(1), enable(2), local(3) radius(4), tacacs(5), none(6)} \
1.3.6.1.4.1.89.79.15.1.{authindex}.15. {"login_n_default" in DEC, каждая буква
логина отделяется от следующей точкой} i {deny (0),
line(1), enable(2), local(3) radius(4), tacacs(5), none(6)} \
1.3.6.1.4.1.89.79.15.1.{authindex}.15. {"login_c_default" in DEC, каждая буква
логина отделяется от следующей точкой} i {deny (0),
line(1), enable(2), local(3) radius(4), tacacs(5), none(6)} \
1.3.6.1.4.1.89.79.15.1.{authindex}.15. {"login_n_default" in DEC, каждая буква
логина отделяется от следующей точкой} i {deny (0),
line(1), enable(2), local(3) radius(4), tacacs(5), none(6)} \
```

```
1.3.6.1.4.1.89.79.15.1.10.15.{"login_c_default" in DEC, каждая буква логина
отделяется от следующей точкой} i {disable (0), enable(1)} \
1.3.6.1.4.1.89.79.15.1.10.15.{"login_n_default" in DEC, каждая буква логина
отделяется от следующей точкой} i {disable (0), enable(1)}
```

### Пример

#### Команда CLI:

```
aaa authentication login authorization default radius local
```

#### Команда SNMP:

```
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.89.79.15.1.2.15.108.111.103.105.110.95.99.95.100.101.102.97.117.108
.116 i 4 \
1.3.6.1.4.1.89.79.15.1.2.15.108.111.103.105.110.95.110.95.100.101.102.97.117.10
8.116 i 4 \
1.3.6.1.4.1.89.79.15.1.3.15.108.111.103.105.110.95.99.95.100.101.102.97.117.108
.116 i 3 \
1.3.6.1.4.1.89.79.15.1.3.15.108.111.103.105.110.95.110.95.100.101.102.97.117.10
8.116 i 3 \
1.3.6.1.4.1.89.79.15.1.10.15.108.111.103.105.110.95.99.95.100.101.102.97.117.10
8.116 i 1 \
1.3.6.1.4.1.89.79.15.1.10.15.108.111.103.105.110.95.110.95.100.101.102.97.117.1
08.116 i 1
```



**authindex** — индекс метода авторизации. Доступные значения от 2 до 7. Первым используется метод с наименьшим номером.

**Поле 1.3.6.1.4.1.89.79.15.1.10.15** разрешает прохождение авторизации для login-пользователя.



**108.111.103.105.110.95.99.95.100.101.102.97.117.108.116** переводится из ASCII-таблицы (расшифровывается login\_c\_default).

**108.111.103.105.110.95.110.95.100.101.102.97.117.108.116** переводится из ASCII-таблицы (расшифровывается login\_n\_default).

### Удаление настройки методов авторизации для login-пользователя

**MIB:** rlaaa.mib

**Используемые таблицы:** rIAAAMethodListEntry — 1.3.6.1.4.1.89.79.15.1

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.79.15.1.{"authindex".15.{"login_c_default" in DEC, каждая буква
логина отделяется от следующей точкой} i {deny(0),
line(1),enable(2),local(3)radius(4),tacacs(5),none(6)} \
1.3.6.1.4.1.89.79.15.1.{"authindex".15.{"login_n_default" in DEC, каждая буква
логина отделяется от следующей точкой} i {deny (0),
line(1),enable(2),local(3)radius(4),tacacs(5),none(6)} \
1.3.6.1.4.1.89.79.15.1.{"authindex".15.{"login_c_default" in DEC, каждая буква
логина отделяется от следующей точкой} i {deny (0),
line(1),enable(2),local(3)radius(4),tacacs(5),none(6)} \
1.3.6.1.4.1.89.79.15.1.{"authindex".15.{"login_n_default" in DEC, каждая буква
логина отделяется от следующей точкой} i {deny (0),
line(1),enable(2),local(3)radius(4),tacacs(5),none(6)} \
1.3.6.1.4.1.89.79.15.1.10.15.{"login_c_default" in DEC, каждая буква логина
отделяется от следующей точкой} i {disable (0), enable(1)} \
1.3.6.1.4.1.89.79.15.1.10.15.{"login_n_default" in DEC, каждая буква логина
отделяется от следующей точкой} i {disable (0), enable(1)}
```

### Пример удаления методов авторизации для login-пользователя

Команда CLI:

```
no aaa authentication login default
```

Команда SNMP:

```
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.89.79.15.1.2.15.108.111.103.105.110.95.99.95.100.101.102.97.117.108
.116 i 3 \
1.3.6.1.4.1.89.79.15.1.2.15.108.111.103.105.110.95.110.95.100.101.102.97.117.10
8.116 i 3 \
1.3.6.1.4.1.89.79.15.1.3.15.108.111.103.105.110.95.99.95.100.101.102.97.117.108
.116 i 0 \
1.3.6.1.4.1.89.79.15.1.3.15.108.111.103.105.110.95.110.95.100.101.102.97.117.10
8.116 i 0 \
1.3.6.1.4.1.89.79.15.1.10.15.108.111.103.105.110.95.99.95.100.101.102.97.117.10
8.116 i 0 \
1.3.6.1.4.1.89.79.15.1.10.15.108.111.103.105.110.95.110.95.100.101.102.97.117.1
08.116 i
```



**authindex** — индекс метода авторизации. Доступные значения от 2 до 7. Первым используется метод с наименьшим номером.

Поле **1.3.6.1.4.1.89.79.15.1.10.15** разрешает прохождение авторизации для login-пользователя.



**108.111.103.105.110.95.99.95.100.101.102.97.117.108.116** переводится из ASCII-таблицы (расшифровывается login\_c\_default).

**108.111.103.105.110.95.110.95.100.101.102.97.117.108.116** переводится из ASCII-таблицы (расшифровывается login\_n\_default).

### Настройка методов авторизации для enable-пользователя

MIB: rlaaa.mib

**Используемые таблицы:** rIAAAMethodListEntry — 1.3.6.1.4.1.89.79.15.1

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.79.15.1.{authindex}.16.{“login_c_default” in DEC, каждая буква
логина отделяется от следующей точкой} i {deny (0),
line(1),enable(2),local(3)radius(4),tacacs(5),none(6)} \
1.3.6.1.4.1.89.79.15.1.{authindex}.16.{“login_n_default” in DEC, каждая буква
логина отделяется от следующей точкой} i {deny (0),
line(1),enable(2),local(3)radius(4),tacacs(5),none(6)} \
1.3.6.1.4.1.89.79.15.1.{authindex}.16.{“login_c_default” in DEC, каждая буква
логина отделяется от следующей точкой} i {deny (0),
line(1),enable(2),local(3)radius(4),tacacs(5),none(6)} \
1.3.6.1.4.1.89.79.15.1.{authindex}.16.{“login_n_default” in DEC, каждая буква
логина отделяется от следующей точкой} i {deny (0),
line(1),enable(2),local(3)radius(4),tacacs(5),none(6)} \
1.3.6.1.4.1.89.79.15.1.10.16.{“login_c_default” in DEC, каждая буква логина
отделяется от следующей точкой} i {disable (0), enable(1)} \
1.3.6.1.4.1.89.79.15.1.10.16.{“login_n_default” in DEC, каждая буква логина
отделяется от следующей точкой} i {disable (0), enable(1)}
```



## Пример настройки методов авторизации для enable-пользователя

### Команда CLI:

```
aaa authentication enable authorization default radius enable
```

### Команда SNMP:

```
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.89.79.15.1.2.16.101.110.97.98.108.101.95.99.95.100.101.102.97.117.1
08.116 i 4 \
1.3.6.1.4.1.89.79.15.1.2.16.101.110.97.98.108.101.95.110.95.100.101.102.97.117.
108.116 i 4 \
1.3.6.1.4.1.89.79.15.1.3.16.101.110.97.98.108.101.95.99.95.100.101.102.97.117.1
08.116 i 2 \
1.3.6.1.4.1.89.79.15.1.3.16.101.110.97.98.108.101.95.110.95.100.101.102.97.117.
108.116 i 2 \
1.3.6.1.4.1.89.79.15.1.10.16.101.110.97.98.108.101.95.99.95.100.101.102.97.117.
108.116 i 1 \
1.3.6.1.4.1.89.79.15.1.10.16.101.110.97.98.108.101.95.110.95.100.101.102.97.117
.108.116 i 1
```



**authindex** — индекс метода авторизации. Доступные значения от 2 до 7. Первым используется метод с наименьшим номером.

Поле **1.3.6.1.4.1.89.79.15.1.10.16** разрешает прохождение авторизации для enable-пользователя.



**101.110.97.98.108.101.95.99.95.100.101.102.97.117.108.116** переводится из ASCII-таблицы (расшифровывается enable\_c\_default).

**101.110.97.98.108.101.95.110.95.100.101.102.97.117.108.116** переводится из ASCII-таблицы (расшифровывается enable\_n\_default).

## Удаление настройки методов авторизации для enable-пользователя

MIB: rlaaa.mib

Используемые таблицы: rIAAAMethodListEntry — 1.3.6.1.4.1.89.79.15.1

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.79.15.1.{authindex}.16.{“login_c_default” in DEC, каждая буква
логина отделяется от следующей точкой} i {deny (0),
line(1),enable(2),local(3)radius(4),tacacs(5),none(6)} \
1.3.6.1.4.1.89.79.15.1.{authindex}.16.{“login_n_default” in DEC, каждая буква
логина отделяется от следующей точкой} i {deny (0),
line(1),enable(2),local(3)radius(4),tacacs(5),none(6)} \
1.3.6.1.4.1.89.79.15.1.{authindex}.16.{“login_c_default” in DEC, каждая буква
логина отделяется от следующей точкой} i {deny (0),
line(1),enable(2),local(3)radius(4),tacacs(5),none(6)} \
1.3.6.1.4.1.89.79.15.1.{authindex}.16.{“login_n_default” in DEC, каждая буква
логина отделяется от следующей точкой} i {deny (0),
line(1),enable(2),local(3)radius(4),tacacs(5),none(6)} \
1.3.6.1.4.1.89.79.15.1.10.16.{“login_c_default” in DEC, каждая буква логина
отделяется от следующей точкой} i {disable (0), enable(1)} \
```

### Пример удаления методов авторизации для enable-пользователя

Команда CLI:

```
no aaa authentication enable default
```

Команда SNMP:

```
snmpset -v2c -c private 192.168.1.30 \
>1.3.6.1.4.1.89.79.15.1.2.16.101.110.97.98.108.101.95.99.95.100.101.102.97.117.
108.116 i 2 \
>1.3.6.1.4.1.89.79.15.1.2.16.101.110.97.98.108.101.95.110.95.100.101.102.97.117.
.108.116 i 2 \
>1.3.6.1.4.1.89.79.15.1.3.16.101.110.97.98.108.101.95.99.95.100.101.102.97.117.
108.116 i 0 \
>1.3.6.1.4.1.89.79.15.1.3.16.101.110.97.98.108.101.95.110.95.100.101.102.97.117.
.108.116 i 0 \
>1.3.6.1.4.1.89.79.15.1.10.16.101.110.97.98.108.101.95.99.95.100.101.102.97.117.
.108.116 i 0 \
>1.3.6.1.4.1.89.79.15.1.10.16.101.110.97.98.108.101.95.110.95.100.101.102.97.11
7.108.116 i 0
```



**authindex** — индекс метода авторизации. Доступные значения от 2 до 7. Первым используется метод с наименьшим номером.

Поле **1.3.6.1.4.1.89.79.15.1.10.16** разрешает прохождение авторизации для enable-пользователя.



**101.110.97.98.108.101.95.99.95.100.101.102.97.117.108.116** переводится из ASCII-таблицы (расшифровывается enable\_c\_default).

**101.110.97.98.108.101.95.110.95.100.101.102.97.117.108.116** переводится из ASCII-таблицы (расшифровывается enable\_n\_default).

## 12.2 Настройка доступа

### Включение TELNET-сервера

**MIB:** radlan-telnet-mib.mib

**Используемые таблицы:** rITelnetEnable — 1.3.6.1.4.1.89.58.7

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.58.7.0 i {on(1), off(2)}
```

### Пример включения TELNET-сервера

Команда CLI:

```
ip telnet server
```

Команда SNMP:

```
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.89.58.7.0 i 1
```

### Включение SSH-сервера

**MIB:** rlssh.mib

**Используемые таблицы:** rlSshServerEnable — 1.3.6.1.4.1.89.78.2.102

```
snmpset -v2c -c <community> <IP address> \  
1.3.6.1.4.1.89.78.2.102.0 i {on(1), off(2)}
```

#### **Пример включения SSH-сервера**

Команда CLI:

```
ip ssh server
```

Команда SNMP:

```
snmpset -v2c -c private 192.168.1.30 \  
1.3.6.1.4.1.89.78.2.102.0 i 1
```

### Просмотр активных сессий

**MIB:** rIAAA.mib

**Используемые таблицы:** rIAAAUserInetName — 1.3.6.1.4.1.89.79.57.1.5

```
snmpwalk -v2c -c <community> <IP address> \  
1.3.6.1.4.1.89.79.57.1.5
```

#### **Пример просмотра активных сессий**

Команда CLI:

```
show users
```

Команда SNMP:

```
snmpwalk -v2c -c public 192.168.1.30 \  
1.3.6.1.4.1.89.79.57.1.5
```

## 13 ЗЕРКАЛИРОВАНИЕ ПОРТОВ

### Настройка зеркалирования портов

**MIB:** rspan.mib

**Используемые таблицы:**

rspanDestinationTable — 1.3.6.1.4.1.89.219.2

rspanSourceTable — 1.3.6.1.4.1.89.219.3

```
snmpset -v2c -c private 192.168.1.30 \  
1.3.6.1.4.1.89.219.2.1.2.{session id} i {ifindex dst port} \  
1.3.6.1.4.1.89.219.2.1.3.{session id} i {span(1), rspan-start(2), rspan-  
final(3)} \  
1.3.6.1.4.1.89.219.2.1.4.{session id} i {monitor-only(1), network(2)} \  
1.3.6.1.4.1.89.219.2.1.5.{session id} i {vlan id} \  
1.3.6.1.4.1.89.219.2.1.6.{session id} i {createAndGo(4), destroy(6)}  
  
snmpset -v2c -c private 192.168.1.30 \  
1.3.6.1.4.1.89.219.3.1.4.{session id}.1.{ifindex src port} i {rx(1), tx(2),  
both(3)} \  
1.3.6.1.4.1.89.219.3.1.5.{session id}.1.{ifindex src port} i {createAndGo(4),  
destroy(6)}
```

### **Пример зеркалирования трафика с интерфейса TenGigabitEthernet 1/0/16 на интерфейс TenGigabitEthernet 1/0/17**

**Команда CLI:**

```
monitor session 7 destination interface TenGigabitEthernet 1/0/17  
monitor session 7 source interface TenGigabitEthernet 1/0/16
```

**Команда SNMP:**

```
snmpset -v2c -c private 192.168.1.30 \  
1.3.6.1.4.1.89.219.2.1.2.7 i 17 \  
1.3.6.1.4.1.89.219.2.1.3.7 i 1 \  
1.3.6.1.4.1.89.219.2.1.4.7 i 1 \  
1.3.6.1.4.1.89.219.2.1.5.7 i 1 \  
1.3.6.1.4.1.89.219.2.1.6.7 i 4  
  
snmpset -v2c -c private 192.168.1.30 \  
1.3.6.1.4.1.89.219.3.1.4.7.1.16 i 3 \  
1.3.6.1.4.1.89.219.3.1.5.7.1.16 i 4
```

### Настройка зеркалирования vlan

**MIB:** rspan.mib

**Используемые таблицы:**

rspanDestinationTable — 1.3.6.1.4.1.89.219.2

rspanSourceTable — 1.3.6.1.4.1.89.219.3

```
snmpset -v2c -c <community> <IP address> \  
1.3.6.1.2.1.16.22.1.3.1.1.4.{ifindex vlan}.{ifindex dst port} i  
{copyRxOnly(1)} \  
1.3.6.1.2.1.16.22.1.3.1.1.5.{ifindex vlan}.{ifindex dst port} i  
{createAndGo(4), destroy(6)}
```

### Пример настройки зеркалирования vlan 622 на интерфейс TenGigabitEthernet 1/0/17

#### Команда CLI:

```
monitor session 7 destination interface TenGigabitEthernet 1/0/17  
monitor session 7 source interface vlan 622
```

#### Команда SNMP:

```
snmpset -v2c -c private 192.168.1.30 \  
1.3.6.1.4.1.89.219.2.1.2.1 i 17 \  
1.3.6.1.4.1.89.219.2.1.3.1 i 1 \  
1.3.6.1.4.1.89.219.2.1.4.1 i 1 \  
1.3.6.1.4.1.89.219.2.1.5.1 i 1 \  
1.3.6.1.4.1.89.219.2.1.6.1 i 4  
  
snmpset -v2c -c private 192.168.1.30 \  
1.3.6.1.4.1.89.219.3.1.4.1.2.100621 i 1 \  
1.3.6.1.4.1.89.219.3.1.5.1.2.100621 i 4
```

## 14 ФУНКЦИИ ДИАГНОСТИКИ ФИЗИЧЕСКОГО УРОВНЯ

### 14.1 Диагностика оптического трансивера

#### Снятие показаний DDM

**MIB:** rlphy.mib

**Используемые таблицы:** rlPhyTestGetResult — 1.3.6.1.4.1.89.90.1.2.1.3

```
snmpwalk -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.90.1.2.1.3.{индекс порта}.{тип параметра}
```

**Пример запроса показаний DDM с интерфейса TenGigabitEthernet 1/0/23 (для всех параметров)**

Команда CLI:

```
show fiber-ports optical-transceiver interface TenGigabitEthernet 1/0/23
```

Команда SNMP:

```
snmpwalk -v2c -c public 192.168.1.30 \
1.3.6.1.4.1.89.90.1.2.1.3.23
```



**Тип параметра может принимать следующие значения:**

rlPhyTestTableTransceiverTemp (5) — температура SFP-трансивера;

rlPhyTestTableTransceiverSupply (6) — напряжение питания в мкВ;

rlPhyTestTableTxBias (7) — ток смещения в мкА;

rlPhyTestTableTxOutput (8) — уровень мощности на передаче в mDbm;

rlPhyTestTableRxOpticalPower (9) — уровень мощности на приеме в mDbm.

#### Просмотр серийного номера SFP-трансивера

**MIB:** eltMes.mib

**Используемые таблицы:** eltMesPhdTransceiver — 1.3.6.1.4.1.35265.1.23.53

```
snmpwalk -v2c -c <community> <IP address> \
1.3.6.1.4.1.35265.1.23.53.1.1.1.6.{индекс порта}
```

**Пример просмотра серийного номера SFP с интерфейса TenGigabitEthernet 1/0/23 (для всех параметров)**

Команда CLI:

```
show fiber-ports optical-transceiver interface TenGigabitEthernet 1/0/23
```

Команда SNMP:

```
snmpwalk -v2c -c public 192.168.1.30 \
1.3.6.1.4.1.35265.1.23.53.1.1.1.6.23
```

## 15 ФУНКЦИИ ОБЕСПЕЧЕНИЯ БЕЗОПАСНОСТИ

### 15.1 Функции обеспечения защиты портов

#### Ограничение количества MAC-адресов, изучаемых на Ethernet-портах

**MIB:** rlintefaces.mib

**Используемые таблицы:** swIfTable — 1.3.6.1.4.1.89.43.1

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.43.1.1.38.{ifIndex} i {max mac addresses}
```

#### **Пример ограничения в 20 MAC-адресов на порт TenGigabitEthernet 1/0/23**

**Команда CLI:**

```
interface TenGigabitEthernet 1/0/23
port security max 20
```

**Команда SNMP:**

```
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.89.43.1.1.38.23 i 20
```

#### Включение port security

**MIB:** rlintefaces.mib

**Используемые таблицы:** swIfPortLockIfRangeTable — 1.3.6.1.4.1.89.43.6

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.43.6.1.3.1 i {locked(1), unlocked(2)} \
1.3.6.1.4.1.89.43.6.1.4.1 i {discard(1), forwardNormal(2), discardDisable(3),
действие над пакетом, не попавшим под правила port security} \
1.3.6.1.4.1.89.43.6.1.5.1 i {true(1), false(2). Для отправки трапов} \
1.3.6.1.4.1.89.43.6.1.6.1 i {частота отправки трапов (сек)} \
1.3.6.1.4.1.89.43.6.1.2.1 x {ifindex в виде битовой маски}
```

#### **Пример настройки port security для интерфейсов TenGigabitEthernet 1/0/21-23**

**Команда CLI:**

```
interface range TenGigabitEthernet 1/0/21-23
port security discard trap 30
```

**Команда SNMP:**

```
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.89.43.6.1.3.1 i 1 \
1.3.6.1.4.1.89.43.6.1.4.1 i 1 \
1.3.6.1.4.1.89.43.6.1.5.1 i 1 \
1.3.6.1.4.1.89.43.6.1.6.1 i 30 \
1.3.6.1.4.1.89.43.6.1.2.1 x "00000E0000"
```



Методика расчета битовой маски приведена в разделе «Приложение А. Методика расчета битовой маски».

### Установка режима работы port security

**MIB:** rlintefaces.mib

**Используемые таблицы:** swIfTable — 1.3.6.1.4.1.89.43.1

```
snmpset -v2c -c <community> <IP address> \  
1.3.6.1.4.1.89.43.1.1.37.{ifIndex} i {disabled(1), dynamic(2), secure-  
permanent(3), secure-delete-on-reset(4)}
```

### **Пример настройки режима ограничения по количеству изученных MAC-адресов на порту TenGigabitEthernet 1/0/23**

Команда CLI:

```
interface TenGigabitEthernet 1/0/23  
port security mode max-addresses
```

Команда SNMP:

```
snmpset -v2c -c private 192.168.1.30 \  
1.3.6.1.4.1.89.43.1.1.37.23 i 2
```

### Просмотр статуса port security

**MIB:** rlintefaces.mib

**Используемые таблицы:** swIfLockAdminStatus — 1.3.6.1.4.1.89.43.1.1.8

```
snmpwalk -v2c -c <community> <IP address> \  
1.3.6.1.4.1.89.43.1.1.8
```

### **Пример просмотра статуса port security**

Команда CLI:

```
show ports security
```

Команда SNMP:

```
snmpwalk -v2c -c public 192.168.1.30 \  
1.3.6.1.4.1.89.43.1.1.8
```

### Просмотр типа port security

**MIB:** rlintefaces.mib

**Используемые таблицы:** swIfAdminLockAction — 1.3.6.1.4.1.89.43.1.1.20

```
snmpwalk -v2c -c <community> <IP address> \  
1.3.6.1.4.1.89.43.1.1.20
```

### **Пример просмотра типа port security**

Команда CLI:

```
show ports security
```

Команда SNMP:

```
snmpwalk -v2c -c public 192.168.1.30 \  
1.3.6.1.4.1.89.43.1.1.20
```



### Просмотр максимально заданного количества MAC-адресов, изучаемых на Ethernet портах

**MIB:** rlinterfaces.mib

**Используемые таблицы:** swIfLockMaxMacAddresses — 1.3.6.1.4.1.89.43.1.1.38

```
snmpwalk -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.43.1.1.38
```

**Пример просмотра максимально заданного количества MAC-адресов, изучаемых на Ethernet-портах**

Команда CLI:

```
show ports security
```

Команда SNMP:

```
snmpwalk -v2c -c public 192.168.1.30 \
1.3.6.1.4.1.89.43.1.1.38
```

### Перевод порта в режим изоляции и внутри группы портов

**MIB:** rlprotectedport.mib

**Используемые таблицы:** rlProtectedPortsTable — 1.3.6.1.4.1.89.132.1

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.132.1.1.1.{Ifindex} i {not-protected(1), protected(2)}
```

**Пример настройки изоляции на портах TenGigabitEthernet 1/0/21 и TenGigabitEthernet 1/0/23**

Команда CLI:

```
interface range TenGigabitEthernet 1/0/23
switchport protected-port
```

Команда SNMP:

```
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.89.132.1.1.1.21 i 2 \
1.3.6.1.4.1.89.132.1.1.1.23 i 2
```

### Создание статической привязки в MAC-таблице

**MIB:** Q-BRIDGE-MIB

**Используемые таблицы:** dot1qStaticUnicastTable — 1.3.6.1.2.1.17.7.1.3.1

```
snmpset -v2c -c <community> -t 20 -r 0 <IP address> \
1.3.6.1.2.1.17.7.1.3.1.1.4.{vlan id}.{mac address(DEC)}. Байты MAC-адреса
разделяются точками}.{ifIndex} i {other(1), invalid(2), permanent(3),
deleteOnReset(4), deleteOnTimeout(5)}
```

**Пример привязки MAC-адреса 00:22:68:7d:0f:3f в vlan 622 к интерфейсу TenGigabitEthernet 1/0/23 в режиме secure (по умолчанию используется режим permanent)**

Команда CLI:

```
mac address-table static 00:22:68:7d:0f:3f vlan 622 interface tenGigabitEthernet
1/0/23 secure
```

Команда SNMP:

```
snmpset -v2c -c private -t 20 -r 0 192.168.1.30 \
1.3.6.1.2.1.17.7.1.3.1.1.4.622.0.34.104.125.15.63.23 i 1
```

### Просмотр MAC-таблицы

**MIB:** Q-BRIDGE-MIB

**Используемые таблицы:** dot1qTpFdbTable — 1.3.6.1.2.1.17.7.1.2.2

```
snmpwalk -v2c -c <community> <IP address> \
1.3.6.1.2.1.17.7.1.2.2
```

#### Пример просмотра MAC-таблицы

Команда CLI:

```
show mac address-table
```

Команда SNMP:

```
snmpwalk -v2c -c public 192.168.1.30 \
1.3.6.1.2.1.17.7.1.2.2
```

### Создание статической привязки в ARP-таблице

**MIB:** RFC1213-MIB

**Используемые таблицы:** ipNetToMediaTable — 1.3.6.1.2.1.4.22

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.2.1.4.22.1.2.{vlan id}.{IP address} x {"MAC address"} \
1.3.6.1.2.1.4.22.1.3.{vlan id}.{IP address} a {IP address} \
1.3.6.1.2.1.4.22.1.4.{vlan id}.{IP address} i 4
```

#### Пример привязки ip 192.168.1.21 и MAC aa:bb:cc:dd:ee:ff к vlan 1

Команда CLI:

```
arp 192.168.1.21 aa:bb:cc:dd:ee:ff vlan 1
```

Команда SNMP:

```
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.2.1.4.22.1.2.100000.192.168.1.21 x "aabbccddeeff" \
1.3.6.1.2.1.4.22.1.3.100000.192.168.1.21 a 192.168.1.21 \
1.3.6.1.2.1.4.22.1.4.100000.192.168.1.21 i 4
```



1. Для удаления привязки необходимо в поле 1.3.6.1.2.1.4.22.1.4 присвоить значение 2.
2. IP-адрес устройства и IP-адрес создаваемой статической записи в ARP-таблице должны находиться в одной подсети.

### Просмотр ARP-таблицы

**MIB:** RFC1213-MIB.mib, Q-BRIDGE-MIB.mib

**Используемые таблицы:**

pNetToMediaPhysAddress — 1.3.6.1.2.1.4.22.1.2

dot1qTpFdbEntry — 1.3.6.1.2.1.17.7.1.2.2.1

```
snmpwalk -v2c -c <community> <IP address> \
1.3.6.1.2.1.4.22.1.2.{(2) ip address, (3)MAC address}
```

```
snmpwalk -v2c -c <community> <IP address> \
1.3.6.1.2.1.17.7.1.2.2.1
```

### Пример просмотра ARP-таблицы

Команда CLI:

```
show arp
```

Команда SNMP:

```
snmpwalk -v2c -c public 192.168.1.30 \
1.3.6.1.2.1.4.22.1.2
snmpwalk -v2c -c public 192.168.1.30 \
1.3.6.1.2.1.17.7.1.2.2.1
```



1. Значение таблицы pNetToMediaPhysAddress отображает IP-адрес и MAC-адрес VLAN.

2. Значение таблицы dot1qTrFdbEntry — отображает статус и идентификационный номер порта, с которого доступно устройство.

## 15.2 Контроль протокола DHCP и опции 82

### Включение/выключение DHCP-сервера на коммутаторе

MIB: rldhcp.mib

Используемые таблицы: rldhcpRelayInterfaceListTable — 1.3.6.1.4.1.89.38.29

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.38.30.0 i {true(1), false(2)}
```

### Пример включения DHCP-сервера на коммутаторе

Команда CLI:

```
ip dhcp server
```

Команда SNMP:

```
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.89.38.30.0 i 1
```

### Просмотр записей таблицы dhcp snooping

MIB: rIBridgeSecurity.mib

Используемые таблицы: rllpDhcpSnoopEntry — 1.3.6.1.4.1.89.112.1.11.1

```
snmpwalk -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.112.1.11.1
```

### Пример просмотра таблицы dhcp snooping

Команда CLI:

```
Show ip dhcp snooping binding
```

Команда SNMP:

```
snmpwalk -v2c -c public 192.168.1.30 \
1.3.6.1.4.1.89.112.1.11.1
```

### Включение/выключение dhcp snooping глобально

**MIB:** rlbridge-security.mib

**Используемые таблицы:** rllpDhcpSnoopEnable — 1.3.6.1.4.1.89.112.1.2

```
snmpset -v2c -c <community> <IP address> \  
1.3.6.1.4.1.89.112.1.2.0 i {enable(1), disable(2)}
```

#### **Пример глобального включения dhcp snooping**

Команда CLI:

```
ip dhcp snooping
```

Команда SNMP:

```
snmpset -v2c -c private 192.168.1.30 \  
1.3.6.1.4.1.89.112.1.2.0 i 1
```

### Включение/выключение dhcp snooping во vlan

**MIB:** rlbridge-security.mib

**Используемые таблицы:** rllpDhcpSnoopEnableVlanTable — 1.3.6.1.4.1.89.112.1.12

```
snmpset -v2c -c <community> <IP address> \  
1.3.6.1.4.1.89.112.1.12.1.2.{vlan id} i {createAndGo(4), destroy(6)}
```

#### **Пример включения dhcp snooping в vlan 622**

Команда CLI:

```
ip dhcp snooping vlan 622
```

Команда SNMP:

```
snmpset -v2c -c private 192.168.1.30 \  
1.3.6.1.4.1.89.112.1.12.1.2.622 i 4
```

### Настройка IP DHCP information option

**MIB:** rlbridgesecurity.mib

**Используемые таблицы:** rllpDhcpOpt82InsertionEnable — 1.3.6.1.4.1.89.112.1.8

```
snmpset -v2c -c <community> <IP address> \  
1.3.6.1.4.1.89.112.1.8.0 i {enable(1), disable(2)}
```

#### **Пример**

Команда CLI:

```
ip dhcp information option
```

Команда SNMP:

```
snmpset -v2c -c public 192.168.1.30 \  
1.3.6.1.4.1.89.112.1.8.0 i 1
```

### Настройка доверенного порта DHCP

**MIB:** rlbridge-security.mib

**Используемые таблицы:** rllpDhcpSnoopTrustedPortTable — 1.3.6.1.4.1.89.112.1.13

```
snmpset -v2c -c <community> <IP address>
1.3.6.1.4.1.89.112.1.13.1.2.{ifIndex} i {createAndGo(4), destroy(6)}
```

### Пример настройки доверенного интерфейса TenGigabitEthernet 1/0/23

Команда CLI:

```
interface TenGigabitEthernet 1/0/23
ip dhcp snooping trust
```

Команда SNMP:

```
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.89.112.1.13.1.2.23 i 4
```

### Настройка DHCP relay во VLAN

**MIB:** rldhcp.mib

**Используемые таблицы:**

```
rlDhcpRelayInterfaceListVlanId1To1024 — 1.3.6.1.4.1.89.38.29.1.3
rlDhcpRelayInterfaceListVlanId1025To2048 — 1.3.6.1.4.1.89.38.29.1.4
rlDhcpRelayInterfaceListVlanId2049To3072 — 1.3.6.1.4.1.89.38.29.1.5
rlDhcpRelayInterfaceListVlanId3073To4094 — 1.3.6.1.4.1.89.38.29.1.6
```

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.38.29.1.3.1 x {битовая маска}
```

### Пример настройки IP DHCP relay enable на vlan 1

Команда CLI:

```
Interface vlan 1
ip dhcp relay enable
```

Команда SNMP:

```
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.89.38.29.1.3.1 x 800000000000
```

### Пример настройки IP DHCP relay enable на 1026 vlan

Команда CLI:

```
Interface vlan 1026
ip dhcp relay enable
```

Команда SNMP:

```
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.89.38.29.1.4.1 x 400000000000
```



Пример расчета битовой маски можно посмотреть в разделе «Приложение А. Методика расчета битовой маски».

## 15.3 Защита IP-адреса клиента (IP source Guard)

### Включение/отключение ip source guard глобально

**MIB:** rlbridge-security.mib

**Используемые таблицы:** rlIpSourceGuardEnable — 1.3.6.1.4.1.89.112.2.2

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.112.2.2.0 i {enable(1), disable(2)}
```

#### **Пример глобального включения ip source guard**

Команда CLI:

```
ip source-guard
```

Команда SNMP:

```
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.89.112.2.2.0 i 1
```

#### **Создание статической привязки ip source guard**

**MIB:** rbridge-security.mib

**Используемые таблицы:** rllpDhcpSnoopStaticTable — 1.3.6.1.4.1.89.112.1.10

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.112.1.10.1.3.{vlan id}.{MAC in DEC. Каждый байт MAC-адреса
отделяется от предыдущего точкой} a {ip address (DEC)} \
1.3.6.1.4.1.89.112.1.10.1.4.{vlan id}.{MAC in DEC. Каждый байт MAC-адреса
отделяется от предыдущего точкой} i {ifIndex} \
1.3.6.1.4.1.89.112.1.10.1.5.{vlan id}.{MAC in DEC. Каждый байт MAC-адреса
отделяется от предыдущего точкой} i {createAndGo(4), destroy(6)}
```

#### **Пример привязки MAC-адреса 00:11:22:33:44:55 к IP 192.168.1.34, vlan 622, интерфейсу TenGigabitEthernet 1/0/23**

Команда CLI:

```
ip source-guard binding 00:11:22:33:44:55 622 192.168.1.34 TenGigabitEthernet
1/0/23
```

Команда SNMP:

```
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.89.112.1.10.1.3.622.0.17.34.51.68.85 a 192.168.1.34 \
1.3.6.1.4.1.89.112.1.10.1.4.622.0.17.34.51.68.85 i 23 \
1.3.6.1.4.1.89.112.1.10.1.5.622.0.17.34.51.68.85 i 4
```

#### **Включение/выключение ip source guard на порту**

**MIB:** rbridge-security.mib

**Используемые таблицы:** rllpSourceGuardPortTable — 1.3.6.1.4.1.89.112.2.5

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.112.2.5.1.2.<ifIndex> i {createAndGo(4), destroy(6)}
```

#### **Пример включения ip source guard на интерфейсе TenGigabitEthernet 1/0/23**

Команда CLI:

```
interface TenGigabitEthernet 1/0/23
ip source-guard
```

Команда SNMP:

```
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.89.112.2.5.1.2.23 i 4
```

## 15.4 Контроль протокола ARP (ARP Inspection)

### Включение/выключение ARP Inspection глобально

**MIB:** rlbridge-security.mib

**Используемые таблицы:** rllpArpInspectEnable — 1.3.6.1.4.1.89.112.3.2

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.112.3.2.0 i {enable(1), disable (2)}
```

#### **Пример глобального включения arp inspection**

Команда CLI:

```
ip arp inspection
```

Команда SNMP:

```
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.89.112.3.2.0 i 1
```

### Включение/выключение ARP Inspection во VLAN

**MIB:** rlbridge-security.mib

**Используемые таблицы:** rllpArpInspectEnableVlanTable — 1.3.6.1.4.1.89.112.3.6

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.112.3.6.1.3.{vlan id} i {createAndGo(4), destroy(6)}
```

#### **Пример включения arp inspection в vlan 622**

Команда CLI:

```
ip arp inspection vlan 622
```

Команда SNMP:

```
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.89.112.3.6.1.3.622 i 4
```

### Настройка доверенного порта ARP Inspection

**MIB:** rlbridge-security.mib

**Используемые таблицы:** rllpArpInspectTrustedPortRowStatus — 1.3.6.1.4.1.89.112.3.7.1.2

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.112.3.7.1.2.{ifIndex} i {createAndGo(4), destroy(6)}
```

#### **Пример настройки доверенного интерфейса TenGigabitEthernet 1/0/23**

Команда CLI:

```
interface TenGigabitEthernet 1/0/23
ip arp inspection trust
```

Команда SNMP:

```
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.89.112.3.7.1.2.23 i 4
```

### Привязка ip arp inspection list к vlan

**MIB:** rlbridge-security.mib

**Используемые таблицы:** rllpArpInspectAssignedListName — 1.3.6.1.4.1.89.112.3.6.1.2

```
snmpset -v2c -c <community> <IP address> \  
1.3.6.1.4.1.89.112.3.6.1.2.{vlan id} s {list name}
```

#### **Пример привязки листа с именем test к vlan 622**

**Команда CLI:**

```
ip arp inspection list assign 100 test
```

**Команда SNMP:**

```
snmpset -v2c -c private 192.168.1.30 \  
1.3.6.1.4.1.89.112.3.6.1.2.622 s test
```

## **15.5 Проверка подлинности клиента на основе порта (802.1x)**

### Включение аутентификации 802.1x на коммутаторе

**MIB:** dot1xPaeSystem.mib

**Используемые таблицы:** dot1xPaeSystemAuthControl — 1.0.8802.1.1.1.1.1

```
snmpset -v2c -c <community> <IP address> \  
1.0.8802.1.1.1.1.1.1.0 i {enabled(1), disabled(2)}
```

#### **Пример включения 802.1x**

**Команда CLI:**

```
dot1x system-auth-control
```

**Команда SNMP:**

```
snmpset -v2c -c private 192.168.1.30 \  
1.0.8802.1.1.1.1.1.1.0 i 1
```

### Включение периодической повторной проверки подлинности (переаутентификации) клиента

**MIB:** draft-ietf-bridge-8021x.mib

**Используемые таблицы:** dot1xAuthReAuthEnabled — 1.0.8802.1.1.1.2.1.1.13

```
snmpset -v2c -c <community> <IP address> \  
1.0.8802.1.1.1.2.1.1.13.{ifIndex} i {true(1), false(2)}
```

#### **Пример включения периодической повторной проверки подлинности клиента на интерфейсе TenGigabitEthernet 1/0/23**

**Команда CLI:**

```
interface tengigabitethernet 1/0/23  
dot1x reauthentication
```

**Команда SNMP:**

```
snmpset -v2c -c private 192.168.1.30 \  
1.0.8802.1.1.1.2.1.1.13.23 i 1
```



### Установка периода между повторными проверками подлинности

**MIB:** draft-ietf-bridge-8021x.mib

**Используемые таблицы:** dot1xAuthConfigTable — 1.0.8802.1.1.1.2.1.1.12

```
snmpset -v2c -c <community> <IP address> \
1.0.8802.1.1.1.2.1.1.12.{ifIndex} u {size 300-4294967295}
```

**Пример установки периода в 300 с между повторными проверками на интерфейсе TenGigabitEthernet 1/0/23**

**Команда CLI:**  

```
interface tengigabitethernet 1/0/23
dot1x timeout reauth-period 300
```

**Команда SNMP:**  

```
snmpset -v2c -c private 192.168.1.30 \
1.0.8802.1.1.1.2.1.1.12.23 u 300
```

### Настройка режимов аутентификации 802.1x на интерфейсе

**MIB:** draft-ietf-bridge-8021x.mib

**Используемые таблицы:** dot1xAuthConfigTable — 1.0.8802.1.1.1.2.1.1.6

```
snmpset -v2c -c <community> <IP address> \
1.0.8802.1.1.1.2.1.1.6.{ifIndex} i {force-Unauthorized(1), auto(2), force-
Authorized(3)}
```

**Пример настройки аутентификации 802.1x в режиме auto на интерфейсе TenGigabitEthernet 1/0/23**

**Команда CLI:**  

```
interface tengigabitethernet 1/0/23
dot1x port-control auto
```

**Команда SNMP:**  

```
snmpset -v2c -c private 192.168.1.30 \
1.0.8802.1.1.1.2.1.1.6.23 i 2
```

### Включение аутентификации, основанной на MAC-адресах пользователей

**MIB:** radlan-dot1x-mib.mib

**Используемые таблицы:** rldot1xAuthenticationPortTable — 1.3.6.1.4.1.89.95.10.1.1

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.95.10.1.1.{ifIndex} i {destroy(1), mac-and-802.1x(2), mac-
only(3)}
```

**Пример включения аутентификации, основанной только на MAC-адресах на интерфейсе TenGigabitEthernet 1/0/23**

**Команда CLI:**  

```
interface tengigabitethernet 1/0/23
dot1x authentication mac
```

**Команда SNMP:**  

```
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.89.95.10.1.1.23 i 3
```

## Разрешение наличия одного/нескольких клиентов на авторизованном порту 802.1X

**MIB:** rllInterfaces.mib

**Используемые таблицы:** swIfTable — 1.3.6.1.4.1.89.43.1.1.30

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.43.1.1.30.{ifIndex} i {single(1), none(2), multi-sessions(3)}
```

### **Пример разрешения наличия нескольких клиентов на интерфейсе TenGigabitEthernet 1/0/23**

**Команда CLI:**

```
interface TenGigabitEthernet 1/0/23
dot1x host-mode multi-sessions
```

**Команда SNMP:**

```
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.89.43.1.1.30.23 i 3
```

## Включение одного или двух методов проверки подлинности, авторизации и учета (AAA) для использования на интерфейсах IEEE 802.1x

**MIB:** rlaaa.mib

**Используемые таблицы:** rIAAAEapMethodListTable — 1.3.6.1.4.1.89.97.1

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.97.1.1.1.7.{“default” in DEC, каждая буква отделяется от
следующей точкой} s {authentication list} \1.3.6.1.4.1.89.97.1.1.2.7.{“default”
in DEC, каждая буква отделяется от следующей точкой} i {Deny(0), radius(1),
none(2)} \
1.3.6.1.4.1.89.97.1.1.3.7.{“default” in DEC, каждая буква отделяется от
следующей точкой} i {Deny(0), radius(1), none(2)} \
1.3.6.1.4.1.89.97.1.1.7.7.{“default” in DEC, каждая буква отделяется от
следующей точкой} i 1
```

### **Пример включения списка RADIUS-серверов для аутентификации пользователя**

**Команда CLI:**

```
aaa authentication dot1x default radius none
```

**Команда SNMP:**

```
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.89.97.1.1.1.7.100.101.102.97.117.108.116 s default \
1.3.6.1.4.1.89.97.1.1.2.7.100.101.102.97.117.108.116 i 1 \
1.3.6.1.4.1.89.97.1.1.3.7.100.101.102.97.117.108.116 i 2 \
1.3.6.1.4.1.89.97.1.1.7.7.100.101.102.97.117.108.116 i 1
```



**1) Для того, чтобы вернуться к настройкам по умолчанию, достаточно значения поменять на Deny(0).**

**2) Default переводится из ASCII в HEX с помощью таблицы, которую можно найти по ссылке <https://ru.wikipedia.org/wiki/ASCII>.**

## Добавление указанного сервера в список используемых RADIUS-серверов

**MIB:** rlaaa.mib

**Используемые таблицы:** rIRadiusServerInetTable — 1.3.6.1.4.1.89.80.8

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.80.8.1.2.1.4.{ip address (DEC)}.{default UDP port 1812}.{default
UDP port 1813} x "{ip address(HEX)}" \
1.3.6.1.4.1.89.80.8.1.1.1.4.{ip address (DEC)}.{default UDP port 1812}.{default
UDP port 1813} i {ipv4(1), ipv6(2), ipv4z(3)} \
1.3.6.1.4.1.89.80.8.1.3.1.4.{ip address(DEC)}.{default UDP port 1812}.{default
UDP port 1813} i {default UDP port 1812} \
1.3.6.1.4.1.89.80.8.1.4.1.4.{ip address(DEC)}.{default UDP port 1812}.{default
UDP port 1813} i {default UDP port 1813} \
1.3.6.1.4.1.89.80.8.1.9.1.4.{ip address (DEC)}.{default UDP port 1812}.{default
UDP port 1813} s "#{encoding key}" \
1.3.6.1.4.1.89.80.8.1.13.1.4.{ip address (DEC)}.{default UDP port 1812}.{default
UDP port 1813} i {createAndGo(4), destroy(6)}
```

### Пример

**Команда CLI:**

```
radius-server host 192.168.1.10 encrypted key da90833f59be
```

**Команда SNMP:**

```
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.89.80.8.1.2.1.4.192.168.1.10.1812.1813 x "c0a8010a" \
1.3.6.1.4.1.89.80.8.1.1.1.4.192.168.1.10.1812.1813 i 1 \
1.3.6.1.4.1.89.80.8.1.3.1.4.192.168.1.10.1812.1813 i 1812 \
1.3.6.1.4.1.89.80.8.1.4.1.4.192.168.1.10.1812.1813 i 1813 \
1.3.6.1.4.1.89.80.8.1.9.1.4.192.168.1.10.1812.1813 s "#da90833f59be" \
1.3.6.1.4.1.89.80.8.1.13.1.4.192.168.1.10.1812.1813 i 4
```

## 15.6 Механизм обнаружения петель (loopback-detection)

### Глобальное включение loopback-detection

**MIB:** rllbd.mib

**Используемые таблицы:** rllbdEnable — 1.3.6.1.4.1.89.127.1

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.127.1.0 i { true(1), false(2) }
```

### Пример глобального включения loopback-detection

**Команда CLI:**

```
loopback-detection enable
```

**Команда SNMP:**

```
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.89.127.1.0 i 1
```

### Изменение интервала loopback-detection

**MIB:** rllbd.mib

**Используемые таблицы:** rllbdDetectionInterval — 1.3.6.1.4.1.89.127.2

```
snmpset -v2c -c <community> <IP address> \  
1.3.6.1.4.1.89.127.2.0 i { seconds 1-60 }
```

#### **Пример изменения интервала loopback-фреймов на 23 секунды**

Команда CLI:

```
loopback-detection interval 23
```

Команда SNMP:

```
snmpset -v2c -c private 192.168.1.30 \  
1.3.6.1.4.1.89.127.2.0 i 23
```

### Изменение режима работы loopback-detection

**MIB:** rllbd.mib

**Используемые таблицы:** rllbdMode — 1.3.6.1.4.1.89.127.3

```
snmpset -v2c -c <community> <IP address> \  
1.3.6.1.4.1.89.127.3.0 i {source-mac-addr(1),base-mac-addr(2), multicast-mac-  
addr(3),broadcast-mac-addr (4) }
```

#### **Пример изменения режима работы loopback-detection на source-mac-addr**

Команда CLI:

```
loopback-detection mode src-mac-addr
```

Команда SNMP:

```
snmpset -v2c -c private 192.168.1.30 \  
1.3.6.1.4.1.89.127.3.0 i 1
```

### Включение/отключение loopback-detection на интерфейсах

**MIB:** rllbd.mib

**Используемые таблицы:** rllbdPortAdminStatus — 1.3.6.1.4.1.89.127.4.1.1

```
snmpset -v2c -c <community> <IP address> \  
1.3.6.1.4.1.89.127.4.1.1.{ifindex} i { enable(1), disable(2) }
```

#### **Пример включения loopback-detection на интерфейсе TenGigabitethernet 1/0/23**

Команда CLI:

```
interface TenGigabitethernet 1/0/23  
loopback-detection enable
```

Команда SNMP:

```
snmpset -v2c -c private 192.168.1.30 \  
1.3.6.1.4.1.89.127.4.1.1.23 i 1
```

### Просмотр рабочего состояния loopback-detection на интерфейсе

**MIB:** rllbd.mib

**Используемые таблицы:** rllbdPortOperStatus — 1.3.6.1.4.1.89.127.4.1.2

```
snmpwalk -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.127.4.1.2.{ifindex}
```

#### **Пример просмотра состояния loopback-detection на интерфейсе TenGigabitEthernet 1/0/23**

Команда CLI:

```
show loopback-detection TenGigabitEthernet 1/0/23
```

Команда SNMP:

```
snmpwalk -v2c -c public 192.168.1.30 \
1.3.6.1.4.1.89.127.4.1.2.23
```



**При использовании SNMP-команды:**

- 1 — состояние inactive,**
- 2 — состояние active,**
- 3 — loopdetected.**

### Просмотр заблокированных VLAN в режиме vlan-based

**MIB:** rllbd.mib

**Используемые таблицы:** eltMesLdb — 1.3.6.1.4.1.35265.1.23.127

```
snmpwalk -v2c -c <community> <IP address> \
1.3.6.1.4.1.35265.1.23.127.4.1.3.{ifindex}.{vlan}
```

#### **Пример просмотра состояния vlan 2 на порту TenGigabitEthernet 1/0/23**

Команда CLI:

```
show loopback-detection TenGigabitEthernet 1/0/2
```

Команда SNMP:

```
snmpwalk -v2c -c public 192.168.1.30 \
1.3.6.1.4.1.35265.1.23.127.4.1.3.23.622
```



**Возможные состояния:**

- 1 — active,**
- 2 — blocked.**

## **15.7 Контроль широковещательного шторма (storm-control)**

### Настройка storm-control на интерфейсе

**MIB:** RADLAN-MIB

**Используемые таблицы:** rlStormCtrl — 1.3.6.1.4.1.89.77

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.77.12.1.2.{ifindex}.{broadcast(1),multicastRegistered(2),multicastUnregistered(3), multicastAll(4), unknownUnicast(5)} u {rate} \
1.3.6.1.4.1.89.77.12.1.3.{ifindex}.{broadcast(1),multicastRegistered(2),multicastUnregistered(3),multicastAll(4), unknownUnicast(5)} I
kiloBitsPerSecond(1),precentaged(2)} \
1.3.6.1.4.1.89.77.12.1.4.{ifindex}.{broadcast(1),multicastRegistered(2),multicastUnregistered(3), multicastAll(4), unknownUnicast(5)} i
{none(1),trap(2), shutdown(3),trapAndShutdown(4)}
```

#### **Пример включения storm-control для broadcast-трафика на интерфейсе TenGigabitEthernet 1/0/23**

##### **Команда CLI:**

```
interface TenGigabitEthernet 1/0/23
storm-control broadcast kbps 10000 trap shutdown
```

##### **Команда SNMP:**

```
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.89.77.12.1.3.23.1 i 1 \
1.3.6.1.4.1.89.77.12.1.2.23.1 u 1000 \
1.3.6.1.4.1.89.77.12.1.4.23.1 i
```

#### **Пример отключения storm-control для broadcast-трафика на интерфейсе TenGigabitEthernet 1/0/23**

##### **Команда CLI:**

```
interface TenGigabitEthernet 1/0/23
no storm-control broadcast
```

##### **Команда SNMP:**

```
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.89.77.12.1.2.23.1 u 0
```

#### **Включить/выключить storm-control для unknown unicast-трафика**

**MIB:** radlan-stormctrl.mib

**Используемые таблицы:** rlStormCtrlRateLimCfgTable — 1.3.6.1.4.1.89.77.12

```
snmpset -v2c -c <community> <IP address> \
iso.3.6.1.4.1.89.77.12.1.2.{ifIndex}.5 u {Kbps,отключить (0)}
```

#### **Пример включения контроля неизвестного одноадресного трафика до 50 кбит/с**

##### **Команда CLI:**

```
interface TenGigabitEthernet 1/0/23
storm-control unicast Kbps 50
```

##### **Команда SNMP:**

```
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.89.77.12.1.2.23.5 u 50
```

## 16 КОНФИГУРИРОВАНИЕ IP И MAC АСР (СПИСКИ КОНТРОЛЯ ДОСТУПА)

### Создание mac access-list

**MIB:** qosclimib.mib

**Используемые таблицы:** rIQosAcITable — 1.3.6.1.4.1.89.88.7

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.88.7.1.2.{index-of-acl} s "{name-of-acl}" \
1.3.6.1.4.1.89.88.7.1.3.{index-of-acl} i {type-of-acl: mac(1), ip (2)} \
1.3.6.1.4.1.89.88.7.1.4.{index-of-acl} i {createAndGo(4), destroy(6)}
```

### **Пример создания MAC ACL с индексом 207**

**Команда CLI:**

```
mac access-list extended 7-mac
```

**Команда SNMP:**

```
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.89.88.7.1.2.207 s "7-mac" \
1.3.6.1.4.1.89.88.7.1.3.207 i 1 \
1.3.6.1.4.1.89.88.7.1.4.207 i 4
```

### Создание ip access-list (ACL)

**MIB:** qosclimib.mib

**Используемые таблицы:** rIQosAcITable — 1.3.6.1.4.1.89.88.7

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.88.7.1.2.{index-of-acl} s "{name-of-acl}" \
1.3.6.1.4.1.89.88.7.1.3.{index-of-acl} i {type-of-acl: mac(1), ip (2)} \
1.3.6.1.4.1.89.88.7.1.4.{index-of-acl} i {createAndGo(4), destroy(6)}
```

### **Пример создания IP ACL с индексом 107**

**Команда CLI:**

```
ip access-list extended 7-ip
```

**Команда SNMP:**

```
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.89.88.7.1.2.107 s "7-ip" \
1.3.6.1.4.1.89.88.7.1.3.107 i 2 \
1.3.6.1.4.1.89.88.7.1.4.107 i 4
```



**Пример заполнения ACL правилами подробно рассмотрен в разделе «Приложение Б. Пример создания типового IP ACL».**

### Привязка IP или MAC ACL к порту

**MIB:** qosclimib.mib

**Используемые таблицы:**

rIQosIfAcIIn — 1.3.6.1.4.1.89.88.13.1.14

rIQosIfPolicyMapStatus — 1.3.6.1.4.1.89.88.13.1.13

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.88.13.1.14.{ifIndex}.2 i {Index-of-acl} \
1.3.6.1.4.1.89.88.13.1.13.{ifIndex}.2 i 1
```

#### Пример назначения правила с индексом 107 (название ACL 7-ip) на порт TenGigabitEthernet 1/0/23

##### Команда CLI:

```
interface TenGigabitEthernet 1/0/23
service-acl input 7-ip
```

##### Команда SNMP:

```
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.89.88.13.1.14.23.2 i 107 \
1.3.6.1.4.1.89.88.13.1.13.23.2 i 1
```



Для удаления ACL с порта достаточно индекс ACL заменить на 0.

```
snmpset -c -v2c private 192.168.1.30 1.3.6.1.4.1.89.88.13.1.14.50.2 i 0
1.3.6.1.4.1.89.88.13.1.13.50.2 i 1
```

#### Привязка IP и MAC ACL к порту

**MIB:** qosclimib.mib

##### Используемые таблицы:

rlQosIfAcIIn — 1.3.6.1.4.1.89.88.13.1.14

rlQosIfIpv6AcIIn — 1.3.6.1.4.1.89.88.13.1.201.3.6.1.4.1.89.88.13.1.20

rlQosIfPolicyMapStatus — 1.3.6.1.4.1.89.88.13.1.13

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.88.13.1.14.{Ifindex}.2 i {Index-of-mac-acl} \
1.3.6.1.4.1.89.88.13.1.20.{Ifindex}.2 i {Index-of-ip-acl} \
1.3.6.1.4.1.89.88.13.1.13.{ifIndex}.2 i 1
```

#### Пример назначения правила с индексом 107 и 207 (название ACL 7-ip для IP ACL и 7-mac для MAC ACL) на порт TenGigabitEthernet 1/0/23 (Ifindex 23)

##### Команда CLI:

```
interface TenGigabitEthernet 1/0/23
service-acl input 7-mac 7-ip
```

##### Команда SNMP:

```
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.89.88.13.1.14.23.2 i 207 \
1.3.6.1.4.1.89.88.13.1.20.23.2 i 107 \
1.3.6.1.4.1.89.88.13.1.13.23.2 i 1
```



Для удаления ACL с порта достаточно индекс IP и MAC ACL заменить на 0.

```
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.89.88.13.1.14.23.2 i 0 \
1.3.6.1.4.1.89.88.13.1.20.23.2 i 0 \
1.3.6.1.4.1.89.88.13.1.13.23.2 i 1
```



## Создание policy-тар и привязка к нему ACL

**MIB:** qosclimib.mib

### **Используемые таблицы:**

rlQosClassMapTable — 1.3.6.1.4.1.89.88.9

rlQosPolicyMapTable — 1.3.6.1.4.1.89.88.11

rlQosPolicyClassPriorityRefTable — 1.3.6.1.4.1.89.88.39

**Схема:** создание policy-тар проводится в несколько запросов

#### **1. Создаем class и назначаем ему свойства**

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.88.9.1.2.{index-of-class} s "{name-of-class-map}" \
1.3.6.1.4.1.89.88.9.1.3.{index-of-class} i {matchAll (1)} \
1.3.6.1.4.1.89.88.9.1.7.{index-of-class} i {index-of-acl} \
1.3.6.1.4.1.89.88.9.1.9.{index-of-class} i {Mark vlan disable (1), enable(2)} \
1.3.6.1.4.1.89.88.9.1.13.{index-of-class} i {create and go(4),destroy(6)}
```

#### **2. Создаем policy-тар и включаем его**

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.88.11.1.2.{index-of-policy-map} s {name-of-policy-map} \
1.3.6.1.4.1.89.88.11.1.3.{index-of-policy-map} i {createAndGo(4), destroy(6)}
```

#### **3. Привязываем class-тар к policy-тар**

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.88.39.1.2.1.{index-of-class} i {index-of-class} \
1.3.6.1.4.1.89.88.39.1.3.1.{index-of-class} i {index-of-policy-map}
```

#### **4. Создаем ограничение скорости для class-тар**

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.88.10.1.2.{Number-of-class-in-policy} s {Policer-cm-20} \
1.3.6.1.4.1.89.88.10.1.3.{Number-of-class-in-policy} i {single(1),
aggregate(2)} \
1.3.6.1.4.1.89.88.10.1.4.{Number-of-class-in-policy} i {rate} \
1.3.6.1.4.1.89.88.10.1.5.{Number-of-class-in-policy} i {burst} \
1.3.6.1.4.1.89.88.10.1.6.{Number-of-class-in-policy} i {none(1), drop(2),
remark(3)} \
1.3.6.1.4.1.89.88.10.1.8.{Number-of-class-in-policy} i {createAndGo(4),
destroy(6)}
```

#### **5. Привязываем ограничение скорости к class-тар**

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.88.9.1.6.{index-of-class} i {Number-of-class-in-policy}
```

#### **6. Задаем значение метки трафику DSCP, cos или указываем выходную очередь**

**1.3.6.1.4.1.89.88.233**

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.88.9.1.4.{index-of-class} i {setDSCP(3), setQueue(4), setCos(5)} \
1.3.6.1.4.1.89.88.9.1.5.{index-of-class} i {Mark value of DSCP/queue/cos(DEC)}
```

**Пример:** IP ACL с index-of-acl = 107 привязывается к class-map с именем test и выставляется метка DSCP = 36(DEC), cos = 4 и queue = 8 для трафика, подпавшего под IP ACL. Class test привязывается к policy-map с именем test1.

**Команда CLI:**

```
qos advanced
 ip access-list extended 7-ip
  permit ip any any
exit

class-map test
 match access-group 7-ip
exit
policy-map test1
 class test
  set dscp 36
  set queue 8
  set cos 4
  police 97000 524288 exceed-action drop
exit
exit
```

**Команда SNMP:**

```
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.89.88.9.1.2.20 s "test" \
1.3.6.1.4.1.89.88.9.1.3.20 i 1 \
1.3.6.1.4.1.89.88.9.1.7.20 i 107 \
1.3.6.1.4.1.89.88.9.1.9.20 i 1 \
1.3.6.1.4.1.89.88.9.1.13.20 i 4

snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.89.88.11.1.2.1 s "test1" \
1.3.6.1.4.1.89.88.11.1.3.1 i 4

snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.89.88.39.1.2.1.20 i 20 \
1.3.6.1.4.1.89.88.39.1.3.1.20 i 1

snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.89.88.10.1.2.1 s "Policer-cm-20" \
1.3.6.1.4.1.89.88.10.1.3.1 i 1 \
1.3.6.1.4.1.89.88.10.1.4.1 u 97000 \
1.3.6.1.4.1.89.88.10.1.5.1 u 524288 \
1.3.6.1.4.1.89.88.10.1.6.1 i 2 \
1.3.6.1.4.1.89.88.10.1.8.1 i 4

snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.89.88.9.1.6.20 i 1

snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.89.88.9.1.4.20 i 3 \
1.3.6.1.4.1.89.88.9.1.5.20 i 36

snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.89.88.9.1.4.20 i 4 \
1.3.6.1.4.1.89.88.9.1.5.20 i 8

snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.89.88.9.1.4.20 i 5 \
1.3.6.1.4.1.89.88.9.1.5.20 i 4
```

### Назначение Policy-map на порт

**MIB:** qosclimib.mib

**Используемые таблицы:** rlQosIfPolicyMapPointerIn — 1.3.6.1.4.1.89.88.13.1.3

```
snmpset -v2c -c <community> <IP address> \  
1.3.6.1.4.1.89.88.13.1.3.{Ifindex}.2 i {Index-of-policy-map}
```

### **Пример назначения policy-map с индексом 1 на порт TenGigabitEthernet 1/0/23**

**Команда CLI:**

```
interface TenGigabitEthernet 1/0/23  
service-policy input test1
```

**Команда SNMP:**

```
snmpset -v2c -c private 192.168.1.30 \  
1.3.6.1.4.1.89.88.13.1.3.23.2 i 1
```

## 17 КОНФИГУРАЦИЯ ЗАЩИТЫ ОТ DOS-АТАК

### Включение security-suite

**MIB:** rISecuritySuiteMib

**Используемые таблицы:** rISecuritySuiteGlobalEnable — 1.3.6.1.4.1.89.120.1

```
snmpset -v2c -c <community> <IP address> 1.3.6.1.4.1.89.120.1.0 i {enable-  
global-rules-only (1), enable-all-rules-types (2), disable (3)}
```

#### **Пример включения класса команд security-suite для всех правил**

Команда CLI:

```
security-suite enable
```

Команда SNMP:

```
snmpset -v2c -c private 192.168.1.30 \  
1.3.6.1.4.1.89.120.1.0 i 2
```

### Настройка режима работы security suite

**MIB:** rISecuritySuiteMib

**Используемые таблицы:** rISecuritySuiteSynProtectionMode — 1.3.6.1.4.1.89.120.10

```
snmpset -v2c -c <community> <IP address> 1.3.6.1.4.1.89.120.10.0 i {disabled  
(1), report (2), block (3)}
```

#### **Пример включения режима report**

Команда CLI:

```
security-suite syn protection mode report
```

Команда SNMP:

```
snmpset -v2c -c private 192.168.1.30 \  
1.3.6.1.4.1.89.120.10.0 i 2
```

### Выключить защиту от TCP-пакетов с одновременно установленными SYN- и FIN- флагами

**MIB:** rISecuritySuiteMib

**Используемые таблицы:** rISecuritySuiteDenySynFinTcp — 1.3.6.1.4.1.89.120.9

```
snmpset -v2c -c <community> <IP address> 1.3.6.1.4.1.89.120.9.0 i {(deny (1),  
permit (2)}
```

#### **Пример включения режима report**

Команда CLI:

```
security-suite deny syn-fin
```

Команда SNMP:

```
snmpset -v2c -c private 192.168.1.30 \  
1.3.6.1.4.1.89.120.9.0 i 2
```

## 18 КАЧЕСТВО ОБСЛУЖИВАНИЯ — QOS

### 18.1 Настройка QoS

#### Ограничение исходящей скорости на Ethernet-портах

**MIB:** qosclimib.mib

**Используемые таблицы:** rlQosIfPolicyEntry — 1.3.6.1.4.1.89.88.13.1

```
snmpset -v2c -c <community> <IP address> \
  1.3.6.1.4.1.89.88.13.1.6.{ifindex порта}.2 i {disable(1),enable
  (1)} \
  1.3.6.1.4.1.89.88.13.1.7.{ifindex порта}.2 i {traffic-shape} \
  1.3.6.1.4.1.89.88.13.1.8.{ifindex порта}.2 i {Burst size in bytes}
```

#### **Пример ограничения исходящей скорости на порту до значения 20 Мбит/с**

**Команда CLI:**

```
interface TenGigabitEthernet 1/0/23
traffic-shape 20480 500000
```

**Команда SNMP:**

```
snmpset -v2c -c private 192.168.1.30 \
  1.3.6.1.4.1.89.88.13.1.6.23.2 i 2 \
  1.3.6.1.4.1.89.88.13.1.7.23.2 i 20480 \
  1.3.6.1.4.1.89.88.13.1.8.23.2 i 500000
```

#### Ограничение входящей скорости на Ethernet-портах

**MIB:** RADLAN-STORMCTRL-MIB

**Используемые таблицы:** rlStormCtrlRateLimCfgTable — 1.3.6.1.4.1.89.77.12

```
snmpset -v2c -c <community> <IP address> \
  1.3.6.1.4.1.89.77.12.1.2.{ifIndex}.6 u {limit} \
  1.3.6.1.4.1.89.77.12.1.5.{ifIndex}.6 u {Burst size in bytes}
```

#### **Пример ограничения входящей скорости на интерфейсе TenGigabitEthernet 1/0/23 до значения 10 Мбит/с**

**Команда CLI:**

```
interface TenGigabitEthernet 1/0/23
rate-limit 10240 burst 500000
```

**Команда SNMP:**

```
snmpset -v2c -c private 192.168.1.30 \
  1.3.6.1.4.1.89.77.12.1.2.23.6 u 10240 \
  1.3.6.1.4.1.89.77.12.1.5.23.6 u 500000
```



**Для отключения rate-limit на интерфейсе необходимо выполнить (на примере интерфейса TenGigabitEthernet 1/0/23):**

```
snmpset -v2c -c private 192.168.1.30 1.3.6.1.4.1.89.77.12.1.2.23.6 u 0
1.3.6.1.4.1.89.77.12.1.5.23.6 u 128000
```

### Создание профиля qos tail-drop и расширение дескрипторов для очередей

**MIB:** eltQosTailDropMIB.mib

**Используемые таблицы:** eltQosTailDropProfileQueueTable — 1.3.6.1.4.1.35265.1.23.12.1.1.1

```
snmpset -v2c -c <community> <IP address> \  
1.3.6.1.4.1.35265.1.23.12.1.1.1.4.{Номер профиля (1-4)}.{номер очереди(1-8)}  
i {size (0-11480)}
```

#### **Пример**

**Команда CLI:**

```
qos tail-drop profile 2  
queue 1 limit 900
```

**Команда SNMP:**

```
snmpset -v2c -c private 192.168.1.30 \  
1.3.6.1.4.1.35265.1.23.12.1.1.1.4.2.1 i 900
```



Чтобы вернуться к настройкам по умолчанию достаточно установить значение параметра равным 12.

### Установка размера пакетного разделяемого пула для порта

**MIB:** eltQosTailDropMIB.mib

**Используемые таблицы:** eltQosTailDropProfileTable — 1.3.6.1.4.1.35265.1.23.12.1.1.4

```
snmpset -v2c -c <community> <IP address> \  
1.3.6.1.4.1.35265.1.23.12.1.1.4.1.2{номер профиля (1-4)} i {size (0-11480)}
```

#### **Пример**

**Команда CLI:**

```
qos tail-drop profile 2  
port-limit 900
```

**Команда SNMP:**

```
snmpset -v2c -c private 192.168.1.30 \  
1.3.6.1.4.1.35265.1.23.12.1.1.4.1.2.2 i 900
```

### Назначение созданного профиля на интерфейс

**MIB:** eltQosTailDropMIB.mib

**Используемые таблицы:** eltQosTailDropIfConfigTable — 1.3.6.1.4.1.35265.1.23.12.1.1.2

```
snmpset -v2c -c <community> <IP address> \  
1.3.6.1.4.1.35265.1.23.12.1.1.2.1.1.{IfIndex} i {номер профиля (1-4)}
```

#### **Пример**

**Команда CLI:**

```
interface TenGigabitEthernet 1/0/23  
qos tail-drop profile 2
```

**Команда SNMP:**

```
snmpset -v2c -c private 192.168.1.30 \  
1.3.6.1.4.1.35265.1.23.12.1.1.2.1.1.23 i 2
```

**Просмотр отображения глобальных лимитов, дескрипторов, буферов****MIB:** ELTEX-MES-QOS-TAIL-DROP-MIB**Используемые таблицы:** eltQosTailDropConfigTable — 1.3.6.1.4.1.35265.1.23.12.1.1.3

```
snmpwalk -v2c -c <community> <ip address> \  
1.3.6.1.4.1.35265.1.23.12.1.1.3
```

**Пример****Команда CLI:**

```
show qos tail-drop
```

**Команда SNMP:**

```
snmpwalk -v2c -c public 192.168.1.30 \  
1.3.6.1.4.1.35265.1.23.12.1.1.3
```

**Просмотр таблицы вывода текущих аллоцированных ресурсов qos (лимитов, дескрипторов, буферов)****MIB:** ELTEX-MES-QOS-TAIL-DROP-MIB**Используемые таблицы:** eltQosTailDropStatusTable — 1.3.6.1.4.1.35265.1.23.12.1.2.1

```
snmpwalk -v2c -c <community> <IP address> \  
1.3.6.1.4.1.35265.1.23.12.1.2.1
```

**Пример****Команда CLI:**

```
show qos tail-drop
```

**Команда SNMP:**

```
snmpwalk -v2c -c public 192.168.1.30 \  
1.3.6.1.4.1.35265.1.23.12.1.2.1
```

**Просмотр Tail Drop счетчиков по очередям****MIB:** RADLAN-COPY-MIB**Используемые таблицы:** eltMesCountersMIB — 1.3.6.1.4.1.35265.1.23.1.8

```
snmpwalk -v2c -c <community> <IP address> \  
1.3.6.1.4.1.35265.1.23.1.8.1.2.1.1.1.7.{ifIndex}.{1-8}.0
```

**Пример просмотра счетчиков для первой очереди****Команда CLI:**

```
show interface TenGigabitEthernet 1/0/23
```

**Команда SNMP:**

```
snmpwalk -v2c -c public 192.168.1.30 \  
1.3.6.1.4.1.35265.1.23.1.8.1.2.1.1.1.7.23.1.0
```

## 18.2 Статистика QoS

### Включение/выключение QoS-статистики

**MIB:** qosclimib.mib

**Используемые таблицы:** eltCountersQosStatisticsEnable — 1.3.6.1.4.1.35265.1.23.1.8.1.1.1.1

```
snmpwalk -v2c -c <community> <IP address> \
1.3.6.1.4.1.35265.1.23.1.8.1.1.1.1.0 i {включить (1), выключить (2)}
```

### **Пример настройки статистики QoS**

**Команда CLI:**

```
qos statistics interface
```

**Команда SNMP:**

```
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.35265.1.23.1.8.1.1.1.1.0 i 1
```

### Просмотр счетчиков QoS-статистики

**MIB:** qosclimib.mib

**Используемые таблицы:** rlInterfaceQueueStatisticsTxPackets — 1.3.6.1.4.1.89.233.2.1.4

```
snmpwalk -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.233.2.1.{Номер счетчика}.{ifIndex}.{Номер очереди}
```

### **Пример снятия показаний счетчика TxPackets на 4 очереди интерфейса TenGigabitEthernet 1/0/23**

**Команда CLI:**

```
show qos statistics interface
```

**Команда SNMP:**

```
snmpwalk -v2c -c public 192.168.1.30 \
1.3.6.1.4.1.89.233.2.1.4.23.4
```



Возможные номера счетчиков:

1. Все счетчики ( )
2. Счетчик Queue(2)
3. Счетчик txpackets(4)
4. Счетчик TxBytes(5)
5. Счетчик droppedpackets(6)
6. Счетчик DroppedBytes(7)

### Пример очистки счетчиков QoS-статистики

**MIB:** qosclimib.mib

**Используемые таблицы:** rlInterfaceQueueStatisticsClear — 1.3.6.1.4.1.89.233.1.0

```
snmpwalk -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.233.1.0 i 1
```



### Пример

Команда CLI:

```
clear qos statistics
```

Команда SNMP:

```
snmpset -v2c -c private 192.168.1.30 \  
1.3.6.1.4.1.89.233.1.0 i 1
```

## 19 МАРШРУТИЗАЦИЯ

### 19.1 Статическая маршрутизация

#### Просмотр таблицы маршрутизации

**MIB:** IP-FORWARD-MIB

**Используемые таблицы:** ipCidrRouteTable — 1.3.6.1.2.1.4.24.4

```
snmpwalk -v2c -c <community> <IP address> \  
1.3.6.1.2.1.4.24.4
```

#### **Пример**

Команда CLI:

```
show ip route
```

Команда SNMP:

```
snmpwalk -v2c -c public 192.168.1.30 \  
1.3.6.1.2.1.4.24.4
```

#### Просмотр статических маршрутов

**MIB:** rlip.mib

**Используемые таблицы:** rIpStaticRouteTable — 1.3.6.1.4.1.89.26.17.1

```
snmpwalk -v2c -c <community> <IP address> \  
1.3.6.1.4.1.89.26.17.1
```

#### **Пример**

Команда CLI:

```
show running-config routing
```

Команда SNMP:

```
snmpwalk -v2c -c public 192.168.1.30 \  
1.3.6.1.4.1.89.26.17.1
```

### 19.2 Динамическая маршрутизация

#### Просмотр соседства OSPF

**MIB:** rlip.mib

**Используемые таблицы:** rIOspfNbrTable — 1.3.6.1.4.1.89.210.11

```
snmpwalk -v2c -c <community> <IP address> \  
1.3.6.1.4.1.89.210.11
```

### Пример

Команда CLI:

```
show ip ospf neighbor
```

Команда SNMP:

```
snmpwalk -v2c -c public 192.168.1.30 \  
1.3.6.1.4.1.89.210.11
```

## 20 КОНФИГУРАЦИЯ VXLAN

### Создание VXLAN-инстанса

**MIB:** ELTEX-EVPN-MIB

**Используемые таблицы:** eltexEvpnVxlanTable — 1.3.6.1.4.1.35265.56.1.1.1,  
eltexEvpnVxlanFirstFreeIndex - 1.3.6.1.4.1.35265.56.1.1.3

```
snmpget -v2c -c <community> <IP address> 1.3.6.1.4.1.35265.56.1.1.3
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.35265.56.1.1.1.1.3.{index} i { adminStatusUp(1),
adminStatusDown(2) } \
1.3.6.1.4.1.35265.56.1.1.1.1.4.{index} i { vni } \
1.3.6.1.4.1.35265.56.1.1.1.1.5.{index} i { vlan } \
1.3.6.1.4.1.35265.56.1.1.1.1.6.{index} s { vxlan_name } \
1.3.6.1.4.1.35265.56.1.1.1.1.2.{index} i 4
```

### **Пример**

**Команда CLI:**

```
vxlan VX105
vni 10105
vlan 105
exit
```

**Команда SNMP:**

```
snmpget -v2c -c private 192.168.1.30 1.3.6.1.4.1.35265.56.1.1.3
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.35265.56.1.1.1.1.3.4 i 1 \
1.3.6.1.4.1.35265.56.1.1.1.1.4.4 i 10105 \
1.3.6.1.4.1.35265.56.1.1.1.1.5.4 i 105 \
1.3.6.1.4.1.35265.56.1.1.1.1.6.4 s "VX105" \
1.3.6.1.4.1.35265.56.1.1.1.1.2.4 i 4
```



**Сначала получаем номер первого свободного индекса, а затем используем его для создания VXLAN-инстанса.**

### Удаление VXLAN-инстанса

**MIB:** ELTEX-EVPN-MIB

**Используемые таблицы:** eltexEvpnVxlanTable — 1.3.6.1.4.1.35265.56.1.1.1

```
snmpset -v2c -c <community> <IP address>
1.3.6.1.4.1.35265.56.1.1.1.1.2.{index} i 6
```

### **Пример удаления VXLAN-инстанса**

**Команда CLI:**

```
no vxlan VX105
```

**Команда SNMP:**

```
snmpset -v2c -c private 192.168.1.30 1.3.6.1.4.1.35265.56.1.1.1.1.2.4 i 6
```

---

### Просмотр VXLAN-инстансов

**MIB:** ELTEX-EVPN-MIB

**Используемые таблицы:** eltexEvpnVxlanTable — 1.3.6.1.4.1.35265.56.1.1.1

```
snmpwalk -v2c -c <community> <IP address>  
1.3.6.1.4.1.35265.56.1.1.1.1
```

### **Пример просмотра VXLAN-инстансов**

**Команда SNMP:**

```
snmpwalk -v2c -c private 192.168.1.30 1.3.6.1.4.1.35265.56.1.1.1.1
```



## ПРИЛОЖЕНИЕ Б. ПРИМЕР СОЗДАНИЯ ТИПОВОГО IP ACL

В данном приложении рассмотрен пример наполнения IP ACL с index-of-acl = 107 правилами вида:

```
ip access-list extended 7-ip
deny udp any bootps any bootpc ace-priority 20
permit igmp any any ace-priority 40
deny ip any 224.0.0.0 15.255.255.255 ace-priority 60
permit ip 37.193.119.7 0.0.0.0 any ace-priority 80
permit ip 10.130.8.3 0.0.0.0 any ace-priority 100
permit ip 192.168.0.0 0.0.0.15 any ace-priority 120
permit ip 37.193.119.7 0.0.0.0 any ace-priority 140
exit
```

### Создание правила deny udp any bootps any bootpc

MIB: qosclimib.mib

#### Используемые таблицы:

rlQosTupleTable — 1.3.6.1.4.1.89.88.5

rlQosAceTidxTable — 1.3.6.1.4.1.89.88.31

**Схема:** создание правила проводится в два запроса.

#### 1. Задаются параметры правила.

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.88.5.1.2.{значение поля 1} i {protocol(1)} \
1.3.6.1.4.1.89.88.5.1.4.{значение поля 1} x {protocol index (HEX)} \
1.3.6.1.4.1.89.88.5.1.3.{значение поля 1} i {Значение в таблице порта для
протокола = 0. Константа для этого правила} \
1.3.6.1.4.1.89.88.5.1.2.{значение поля 2} i {udp-port-src(6)} \
1.3.6.1.4.1.89.88.5.1.3.{значение поля 2} i {Number of source port (DEC)} \
1.3.6.1.4.1.89.88.5.1.4.{значение поля 2} x {source ip(HEX)} \
1.3.6.1.4.1.89.88.5.1.2.{значение поля 3} i { udp-port-dst(6)} \
1.3.6.1.4.1.89.88.5.1.3.{значение поля 3} i {Number of dst port (DEC)} \
1.3.6.1.4.1.89.88.5.1.4.{значение поля 3} x {dst ip(HEX)}
```

#### 2. Привязка правила по индексу (index-of-rule) к ACL по индексу (index-of-acl) как deny.

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.88.31.1.3.{index-of-acl}.{index-of-rule} i {deny(2)} \
1.3.6.1.4.1.89.88.31.1.4.{index-of-acl}.{index-of-rule} i {udp(3)} \
1.3.6.1.4.1.89.88.31.1.5.{index-of-acl}.{index-of-rule} i {значение поля 1} \
1.3.6.1.4.1.89.88.31.1.7.{index-of-acl}.{index-of-rule} i {значение поля 3} \
1.3.6.1.4.1.89.88.31.1.9.{index-of-acl}.{index-of-rule} i {значение поля 2}
```

**Пример добавления правила deny udp any bootps any bootpc в IP ACL 7-ip (т.к. предполагается, что правило первое по счету, то index-of-rule=20)**

#### Команда CLI:

```
ip access-list extended 7-ip
deny udp any bootps any bootpc ace-priority 20
exit
```

#### Команда SNMP:

```
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.89.88.5.1.2.1 i 1 \
```

```
1.3.6.1.4.1.89.88.5.1.4.1 x "0x11 FF" \
1.3.6.1.4.1.89.88.5.1.3.1 i 0 \
1.3.6.1.4.1.89.88.5.1.2.2 i 6 \
1.3.6.1.4.1.89.88.5.1.3.2 i 67 \
1.3.6.1.4.1.89.88.5.1.4.2 x "0x00 00" \
1.3.6.1.4.1.89.88.5.1.2.3 i 7 \
1.3.6.1.4.1.89.88.5.1.3.3 i 68 \
1.3.6.1.4.1.89.88.5.1.4.3 x "0x00 00"

snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.89.88.31.1.3.1.20 i 2 \
1.3.6.1.4.1.89.88.31.1.4.1.20 i 3 \
1.3.6.1.4.1.89.88.31.1.5.1.20 i 1 \
1.3.6.1.4.1.89.88.31.1.7.1.20 i 2 \
1.3.6.1.4.1.89.88.31.1.9.1.20 i 3
```

### Создание правила permit igmp any any

**MIB:** qosclimib.mib

#### **Используемые таблицы:**

rIQoSTupleTable — 1.3.6.1.4.1.89.88.5

rIQoSAceTidxTable — 1.3.6.1.4.1.89.88.31

**Схема:** создание правила проводится в два запроса.

#### 1. Задаются параметры правила.

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.88.5.1.2.{значение поля 4} i {protocol(1)} \
1.3.6.1.4.1.89.88.5.1.4.{значение поля 4} x {protocol index (HEX)}
```

#### 2. Привязка правила по индексу (index-of-rule) к ACL по индексу (index-of-acl) как permit.

```
snmpset -v2c -c <community> <IP address> \
```

```
1.3.6.1.4.1.89.88.31.1.3.{index-of-acl}.{index-of-rule} i {permit (1)} \
1.3.6.1.4.1.89.88.31.1.4.{index-of-acl}.{index-of-rule} i {igmp (8)} \
1.3.6.1.4.1.89.88.31.1.5.{index-of-acl}.{index-of-rule} i {значение поля 4}
```

**Пример добавления правила permit igmp any any в IP ACL 7-ip (т.к. предполагается, что правило второе по счету, то index-of-rule=40)**

#### **Команда CLI:**

```
ip access-list extended 7-ip
 permit igmp any any ace-priority 40
exit
```

#### **Команда SNMP:**

```
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.89.88.5.1.2.4 i 1 \
1.3.6.1.4.1.89.88.5.1.4.4 x "0x02 FF"

snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.89.88.31.1.3.1.40 i 1 \
1.3.6.1.4.1.89.88.31.1.4.1.40 i 8 \
1.3.6.1.4.1.89.88.31.1.5.1.40 i 4
```



### Создание правила deny ip any any any 224.0.0.0 15.255.255.255

**MIB:** qosclimib.mib

**Используемые таблицы:**

rlQosTupleTable — 1.3.6.1.4.1.89.88.5

rlQosAceTidxTable — 1.3.6.1.4.1.89.88.31

**Схема:** создание правила проводится в два запроса.

1. Задаются параметры правила.

```
snmpset -v2c -c <community> <IP address> \
  1.3.6.1.4.1.89.88.5.1.2.{значение поля 5} i {ip-dest(3)} \
  1.3.6.1.4.1.89.88.5.1.4.{значение поля 5} x {dst ip +wildcard mask (HEX)}
```

2. Привязка правила по индексу (index-of-rule) к ACL по индексу (index-of-acl) как deny.

```
snmpset -v2c -c <community> <IP address> \
  1.3.6.1.4.1.89.88.31.1.3.{index-of-acl}.{index-of-rule} i {deny (2)} \
  1.3.6.1.4.1.89.88.31.1.4.{index-of-acl}.{index-of-rule} i {ip (1)} \
  1.3.6.1.4.1.89.88.31.1.5.{index-of-acl}.{index-of-rule} i {значение поля 5}
```

**Пример добавления правила deny ip any any any 224.0.0.0 15.255.255.255 в IP ACL 7-ip (т.к. предполагается, что правило третье по счету, то index-of-rule=60)**

**Команда CLI:**

```
ip access-list extended 7-ip
deny ip any any any 224.0.0.0 15.255.255.255 ace-priority 60
exit
```

**Команда SNMP:**

```
snmpset -v2c -c private 192.168.1.30 \
  1.3.6.1.4.1.89.88.5.1.2.5 i 3 \
  1.3.6.1.4.1.89.88.5.1.4.5 x "0xE0 00 00 00 0F FF FF FF"
```

```
snmpset -v2c -c private 192.168.1.30 \
  1.3.6.1.4.1.89.88.31.1.3.1.60 i 2 \
  1.3.6.1.4.1.89.88.31.1.4.1.60 i 1 \
  1.3.6.1.4.1.89.88.31.1.5.1.60 i 5
```

### Создание правила permit ip any any 37.193.119.7 0.0.0.0 any

**MIB:** qosclimib.mib

**Используемые таблицы:** rlQosTupleTable — 1.3.6.1.4.1.89.88.5, rlQosAceTidxTable — 1.3.6.1.4.1.89.88.31

**Схема:** создание правила проводится в два запроса.

1. Задаются параметры правила.

```
snmpset -v2c -c <community> <IP address> \
  1.3.6.1.4.1.89.88.5.1.2.{значение поля 6} i {ip-source(2)} \
  1.3.6.1.4.1.89.88.5.1.4.{значение поля 6} x {source ip +wildcard mask (HEX)}
```

## 2. Привязка правила по индексу (index-of-rule) к ACL по индексу (index-of-acl) как permit.

```
snmpset -v2c -c <community> <IP address> \
  1.3.6.1.4.1.89.88.31.1.3.{index-of-acl}.{index-of-rule} i {permit (1)} \
  1.3.6.1.4.1.89.88.31.1.4.{index-of-acl}.{index-of-rule} i {ip (1)} \
  1.3.6.1.4.1.89.88.31.1.5.{index-of-acl}.{index-of-rule} i {значение поля 6}
```

**Пример добавления правила permit ip 37.193.119.7 0.0.0.0 any в IP ACL 7-ip (т.к. предполагается, что правило четвертое по счету, то index-of-rule=80)**

**Команда CLI:**

```
ip access-list extended 7-ip
permit ip 37.193.119.7 0.0.0.0 any ace-priority 80
exit
```

**Команда SNMP:**

```
snmpset -v2c -c private 192.168.1.30 \
  1.3.6.1.4.1.89.88.5.1.2.6 i 2 \
  1.3.6.1.4.1.89.88.5.1.4.6 x "0x25 C1 77 07 00 00 00 00"

snmpset -v2c -c private 192.168.1.30 \
  1.3.6.1.4.1.89.88.31.1.3.1.80 i 1 \
  1.3.6.1.4.1.89.88.31.1.4.1.80 i 1 \
  1.3.6.1.4.1.89.88.31.1.6.1.80 i 6
```

### Создание правила permit ip 10.130.8.3 0.0.0.0 any

**MIB:** qosclimib.mib

**Используемые таблицы:**

rIQosTupleTable — 1.3.6.1.4.1.89.88.5

rIQosAceTidxTable — 1.3.6.1.4.1.89.88.31

**Схема:** создание правила проводится в два запроса.

#### 1. Задаются параметры правила.

```
snmpset -v2c -c <community> <IP address> \
  1.3.6.1.4.1.89.88.5.1.2.{значение поля 7} i {ip-source(2)} \
  1.3.6.1.4.1.89.88.5.1.4.{значение поля 7} x {source ip +wildcard mask (HEX)}
```

#### 2. Привязка правила по индексу (index-of-rule) к ACL по индексу (index-of-acl) как permit

```
snmpset -v2c -c <community> <IP address> \
  1.3.6.1.4.1.89.88.31.1.3.{index-of-acl}.{index-of-rule} i {permit (1)} \
  1.3.6.1.4.1.89.88.31.1.4.{index-of-acl}.{index-of-rule} i {ip (1)} \
  1.3.6.1.4.1.89.88.31.1.5.{index-of-acl}.{index-of-rule} i {значение поля 7}
```

**Пример добавления правила permit ip 10.130.8.3 0.0.0.0 any в IP ACL 7-ip (т.к. предполагается, что правило пятое по счету, то index-of-rule=100)**

**Команда CLI:**

```
ip access-list extended 7-ip
  permit ip 10.130.8.3 0.0.0.0 any ace-priority 100
exit
```

**Команда SNMP:**

```
snmpset -v2c -c private 192.168.1.30 \
  1.3.6.1.4.1.89.88.5.1.2.7 i 2 \
```

```
1.3.6.1.4.1.89.88.5.1.4.7 x "0x0A 82 08 03 00 00 00 00"
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.89.88.31.1.3.1.100 i 1 \
1.3.6.1.4.1.89.88.31.1.4.1.100 i 1 \
1.3.6.1.4.1.89.88.31.1.6.1.100 i 7
```

### Создание правила permit ip any any 192.168.0.0 0.0.0.15 any

**MIB:** qosclimib.mib

#### **Используемые таблицы:**

rlQosTupleTable — 1.3.6.1.4.1.89.88.5

rlQosAceTidxTable — 1.3.6.1.4.1.89.88.31

**Схема:** Создание правила проводится в два запроса.

#### 1. Задаются параметры правила.

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.88.5.1.2.{значение поля 8} i {ip-source(2)} \
1.3.6.1.4.1.89.88.5.1.4.{значение поля 8} x {source ip +wildcard mask (HEX)}
```

#### 2. Привязка правила по индексу (index-of-rule) к ACL по индексу (index-of-acl) как permit.

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.88.31.1.3.{index-of-acl}.{index-of-rule} i {permit (1)} \
1.3.6.1.4.1.89.88.31.1.4.{index-of-acl}.{index-of-rule} i {ip (1)} \
1.3.6.1.4.1.89.88.31.1.5.{index-of-acl}.{index-of-rule} i {значение поля 8}
```

**Пример добавления правила permit ip 192.168.0.0 0.0.0.15 any в IP ACL 7-ip (т.к. предполагается, что правило шестое по счету, то index-of-rule=120)**

#### **Команда CLI:**

```
ip access-list extended 7-ip
 permit ip 192.168.0.0 0.0.0.15 any ace-priority 120
exit
```

#### **Команда SNMP:**

```
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.89.88.5.1.2.8 i 2 \
1.3.6.1.4.1.89.88.5.1.4.8 x "0xC0 A8 00 00 00 00 00 0F"

snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.89.88.31.1.3.1.120 i 1 \
1.3.6.1.4.1.89.88.31.1.4.1.120 i 1 \
1.3.6.1.4.1.89.88.31.1.6.1.120 i 8
```

#### 1. Привязка правила по индексу (index-of-rule) к ACL по индексу (index-of-acl) как permit.

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.88.31.1.3.{index-of-acl}.{index-of-rule} i {permit (1)} \
1.3.6.1.4.1.89.88.31.1.4.{index-of-acl}.{index-of-rule} i {ip (1)} \
1.3.6.1.4.1.89.88.31.1.5.{index-of-acl}.{index-of-rule} i {значение поля 9} \
1.3.6.1.4.1.89.88.31.1.6.{index-of-acl}.{index-of-rule} i {значение поля 10}
```

## ПРИЛОЖЕНИЕ В. ПРИМЕР СОЗДАНИЯ, НАПОЛНЕНИЯ И УДАЛЕНИЯ OFFSET-LIST С ПРИВЯЗКОЙ К MAC ACL

В данном приложении рассмотрен пример создания и наполнения MAC ACL с index-of-acl = 207 правилами вида:

```
mac access-list extended 7-mac
offset-list PADO 12 12 00 88 12 13 00 63 12 15 00 07
deny any any offset-list PADO ace-priority 20
```

### Создание mac access-list

**MIB:** qosclimib.mib

**Используемые таблицы:** rIQosAcITable — 1.3.6.1.4.1.89.88.7

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.88.7.1.2.{index-of-acl} s "{name-of-acl}" \
1.3.6.1.4.1.89.88.7.1.3.{index-of-acl} i {type-of-acl: mac(1), ip (2)} \
1.3.6.1.4.1.89.88.7.1.4.{index-of-acl} i {createAndGo(4), destroy(6)}
```

### **Пример создания MAC ACL с индексом 207**

**Команда CLI:**

```
mac access-list extended 7-mac
```

**Команда SNMP:**

```
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.89.88.7.1.2.1 s "7-mac" \
1.3.6.1.4.1.89.88.7.1.3.1 i 1 \
1.3.6.1.4.1.89.88.7.1.4.1 i 4
```

### Создание правила в MAC ACL на основе EtherType

**MIB:** qosclimib.mib

**Используемые таблицы:**

rIQosTupleTable — 1.3.6.1.4.1.89.88.5

rIQosAceTidxTable — 1.3.6.1.4.1.89.88.31

**Схема:** создание правила проводится в два запроса.

1. Задаются параметры правила.

```
snmpset -v2c -c <community> <IP address> \
1.3.6.1.4.1.89.88.5.1.2.{значение поля 1} i {mac-src(10), mac-dest(11),
vlan(12)} \
1.3.6.1.4.1.89.88.5.1.4.{значение поля 1} x {protocol index (HEX)} \
1.3.6.1.4.1.89.88.5.1.3.{значение поля 1} i {Значение в таблице порта для
протокола = 0. Константа для этого правила} \
1.3.6.1.4.1.89.88.5.1.2.{значение поля 2} i {ether-type(17)} \
1.3.6.1.4.1.89.88.5.1.3.{значение поля 2} i {ether-type (DEC)} \
1.3.6.1.4.1.89.88.5.1.4.{значение поля 2} x {Нулевое поле - константа}
```

## 2. Привязка правила по индексу (index-of-rule) к ACL по индексу (index-of-acl) как permit.

```
snmpset -v2c -c <community> <IP address> \
.1.3.6.1.4.1.89.88.31.1.3.{index-of-acl}.{index-of-rule} i {permit(1)} \
.1.3.6.1.4.1.89.88.31.1.4.{index-of-acl}.{index-of-rule} i {mac(5)} \
.1.3.6.1.4.1.89.88.31.1.5.{index-of-acl}.{index-of-rule} i {значение поля 1} \
.1.3.6.1.4.1.89.88.31.1.9.{index-of-acl}.{index-of-rule} i {значение поля 2}
```

**Пример добавления правила permit 00:1f:c6:8b:c6:8a 00:00:00:00:00:00 any 806 0000 в MAC ACL 7-мас (т.к. предполагается, что правило первое по счету, то index-of-rule=20)**

**Команда CLI:**

```
mac access-list extended 7-mac
 permit 00:1f:c6:8b:c6:8a 00:00:00:00:00:00 any 806 0000 ace-priority 20
exit
```

**Команда SNMP:**

```
snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.89.88.5.1.2.1 i 10 \
1.3.6.1.4.1.89.88.5.1.2.2 i 17 \
1.3.6.1.4.1.89.88.5.1.4.1 x "0x001fc68bc68a000000000000" \
1.3.6.1.4.1.89.88.5.1.3.1 i 0 \
1.3.6.1.4.1.89.88.5.1.3.2 i 2054 \
1.3.6.1.4.1.89.88.5.1.4.2 x "0x00 00"

snmpset -v2c -c private 192.168.1.30 \
1.3.6.1.4.1.89.88.31.1.3.1.20 i 1 \
1.3.6.1.4.1.89.88.31.1.4.1.20 i 5 \
1.3.6.1.4.1.89.88.31.1.5.1.20 i 1 \
1.3.6.1.4.1.89.88.31.1.9.1.20 i 2
```

---

## ТЕХНИЧЕСКАЯ ПОДДЕРЖКА

Для получения технической консультации по вопросам эксплуатации оборудования ООО «Предприятие «ЭЛТЕКС» вы можете обратиться в Сервисный центр компании:

Форма обратной связи на сайте: <https://eltex-co.ru/support/>

Servicedesk: <https://servicedesk.eltex-co.ru>

На официальном сайте компании вы можете найти техническую документацию и программное обеспечение для продукции ООО «Предприятие «ЭЛТЕКС», обратиться к базе знаний или оставить интерактивную заявку:

Официальный сайт компании: <https://eltex-co.ru/>

База знаний: <https://docs.eltex-co.ru/display/EKB/Eltex+Knowledge+Base>

Центр загрузок: <https://eltex-co.ru/support/downloads>