Сервисные маршрутизаторы серии ESR
ESR-3200
Пограничный контроллер сессий
ESBC-3200, vESBC

Руководство по эксплуатации
Версия ПО 1.6.0

Содержание

1 Введение ... 6
1.1 Аннотация ... 6
1.2 Целевая аудитория ... 6
1.3 Условные обозначения ... 6
1.4 Примечания и предупреждения ... 7
2 Описание изделий ... 8
2.1 Назначение ... 8
2.2 Функции ... 9
2.2.1 Функции интерфейсов ... 9
2.2.2 Функции при работе с MAC-адресами ... 9
2.2.3 Функции второго уровня сетевой модели OSI ... 10
2.2.4 Функции третьего уровня сетевой модели OSI ... 10
2.2.5 Функции туннелирования трафика ... 11
2.2.6 Функции управления и конфигурирования ... 12
2.2.7 Функции сетевой защиты ... 13
2.3 Основные технические характеристики ... 14
2.4 Конструктивное исполнение ... 16
2.4.1 Конструктивное исполнение ESBC-3200 ... 16
2.4.2 Световая индикация ... 19
2.5 Комплект поставки ... 21
3 Установка и подключение ... 22
3.1 Установка устройства в стойку ... 22
3.2 Установка модулей питания ... 23
3.3 Подключение питающей сети ... 24
3.4 Установка и удаление SFP-трансиверов ... 25
3.4.1 Установка трансивера ... 25
3.4.2 Удаление трансивера ... 25
3.5 Подключение к vESBC ... 25
4 Интерфейсы управления ... 26
4.1 Интерфейс командной строки (CLI) ... 26
4.2 Типы и порядок именования пограничного контроллера сессий ... 26
4.3 Типы и порядок именования туннелей пограничного контроллера сессий ... 29
5 Начальная настройка устройства ... 31
5.1 Заводская конфигурация устройства (только для ESBC-3200) ... 31
5.1.1 Описание заводской конфигурации ... 31
5.2 Подключение и конфигурирование устройства ... 32
5.2.1 Подключение к устройству ... 33
5.2.2 Применение изменения конфигурации ... 33
5.2.3 Базовая настройка устройства ... 34
6 Обновление программного обеспечения ... 38
6.1 Обновление программного обеспечения средствами системы ... 38
6.2 Обновление программного обеспечения из начального загрузчика ... 40
6.3 Обновление вторичного загрузчика (U-Boot) ... 41

7 Рекомендации по безопасной настройке ... 44
7.1 Общие рекомендации ... 44
7.2 Настройка системы логирования событий ... 45
7.2.1 Рекомендации ... 45
7.2.2 Предупреждения ... 45
7.2.3 Пример настройки ... 45
7.3 Настройка политики использования паролей ... 46
7.3.1 Рекомендации ... 46
7.3.2 Пример настройки ... 46
7.4 Настройка политики AAA ... 47
7.4.1 Рекомендации ... 47
7.4.2 Предупреждения ... 47
7.4.3 Пример настройки ... 48
7.5 Настройка удалённого управления ... 49
7.5.1 Рекомендации ... 49
7.5.2 Пример настройки ... 49
7.6 Настройка механизмов защиты от сетевых атак ... 50
7.6.1 Рекомендации ... 50
7.6.2 Пример настройки ... 51
8 Примеры подключения ESBC к сети передачи данных ... 52
8.1 Подключение к разным сетям с использованием двух сетевых интерфейсов ... 52
8.2 Подключение к сети с использованием одного сетевого интерфейса ... 53
8.3 Подключение к сети с использованием нескольких сетевых интерфейсов (резервирование линков) ... 53
8.3.1 Использование протокола LACP для агрегирования сетевых интерфейсов в один логический канал ... 53
8.3.2 Использование моста (Bridge) для терминации на уровне L3 ... 54
8.4 Подключение к нескольким коммутаторам с использованием нескольких сетевых интерфейсов (резервирование линков и узлов сети) ... 55
8.4.1 Использование протокола LACP для агрегирования сетевых интерфейсов в один логический канал ... 55
8.4.2 Использование моста (Bridge) для терминации на уровне L3 ... 55
8.5 Использование кластера (только для ESBC-3200) ... 55
9 Управление ESBC ... 56
9.1 Общие сведения ... 57
9.2 Настройка абонентских интерфейсов ... 58
9.3 Настройка SIP-транков ... 59
9.4 Настройка транковых групп ... 60
9.4.1 Логика работы транковой группы для распределения вызовов на транки, входящие в ее состав ... 63
9.5 Настройка SIP-транспортов ... 65
9.6 Настройка медиаресурсов ... 66
9.7 Настройка таблиц маршрутизации ... 67
9.8 Настройка модификаторов ... 70
9.8.1 Общие модификаторы ... 71
9.8.2 Модификаторы SIP ... 73
9.9 Настройка SIP-профилей ... 98

9.9.1 Контроль доступности направления ... 98
9.9.2 Список причин отбоя для перехода на следующее направление ... 99
9.9.3 Поведение при перенаправлении ... 101
9.9.4 Игнорирование OPTIONS ... 102
9.10 Настройка медиапрофилей ... 102
9.10.1 Управление типом медиаданных и кодеками ... 102
9.10.2 Транскодирование ... 109
9.10.3 Таймаут ожидания RTP-пакетов ... 121
9.10.4 SRTP ... 122
9.10.5 Контроль источника RTP ... 125
9.11 Настройка профилей безопасности ... 126
9.11.1 Общий принцип работы модуля fail2ban ... 126
9.11.2 Фильтрация SIP-флуда ... 127
9.11.3 Блокировка по AOR/User-Agent ... 130
9.11.4 Объединение ошибок по IP-адресу ... 133
9.12 Настройка криптопрофилей ... 135
9.13 Настройка NAT ... 139
9.14 Настройка Public IP ... 140
9.15 Настройка QoS ... 143
9.16 Изменение количества модулей ... 144
9.17 Ограничение входящего трафика ... 146
9.18 Мониторинг ... 150
9.19 Аварии ... 154
9.20 Настройка CDR ... 157
9.21 Работа с логами ... 159
9.22 Примеры настройки ESBC ... 162
9.22.1 Настройка для SIP-абонентов ... 162
9.22.2 Настройка для SIP-транков ... 165
9.22.3 Настройка для SIP-абонентов, использующих WebRTC ... 168
9.22.4 Настройка динамического режима для SIP-транков ... 171
10 Управление интерфейсами ... 175
11 Управление туннелированием ... 175
12 Управление функциями второго уровня (L2) ... 175
13 Управление QoS ... 175
14 Управление маршрутизацией ... 175
15 Управление технологией MPLS ... 175
16 Управление безопасностью ... 175
17 Управление резервированием ... 175
18 Управление кластеризацией ... 176
18.1 Первичная настройка кластера ... 177
18.2 Настройка внешних сетевых интерфейсов ... 178
18.3 Настройка кластерного интерфейса ... 179
18.4 Настройка кластера ... 180
19 Управление удаленным доступом ... 181
20 Управление сервисами ... 181

21 Мониторинг ... 181
22 Управление BRAS (Broadband Remote Access Server) ... 181
23 Управление лицензированием ... 182
23.1 Виды лицензий ESBC ... 182
23.1.1 vESBC ... 182
23.1.2 ESBC-3200 ... 182
23.2 Способы получения лицензии ... 183
23.3 Статусы лицензий ... 183
23.4 ELM ... 183
23.4.1 Алгоритм работы с сервером ELM ... 183
23.4.2 Получение лицензии для vESBC через ELM ... 183
23.4.3 Получение лицензии для ESBC-3200 через ELM ... 185
23.5 Загрузка и активация файловой лицензии ... 186
24 Часто задаваемые вопросы ... 187
25 Приложение A. Packet Flow ... 189
25.1 Порядок обработки входящего/исходящего трафика сетевыми службами пограничного контроллера сессий ESBC ... 189
25.2 Порядок обработки транзитного трафика сетевыми службами пограничного контроллера сессий ESBC ... 191

1 Введение

- Аннотация
- Целевая аудитория
- Условные обозначения
- Примечания и предупреждения

1.1 Аннотация

Производительность, надёжность и безопасность — ключевые приоритеты при организации VoIP-телефонии в корпоративной сети. Необходимо обеспечить не только совместимость оборудования на всех уровнях и его отлаженную работу, но и защиту от различных атак. Игнорирование последнего приводит к взлому VoIP-сети злоумышленниками.

Пограничный контроллер сессий (ESBC) поможет избежать этих проблем. Он используется для сокрытия топологии VoIP-сети, защиты от несанкционированного доступа, а также управления трафиком.

В данном руководстве по эксплуатации изложены назначение, технические характеристики, конструктивное исполнение, порядок установки, рекомендации по начальной настройке и обновлению программного обеспечения пограничного контроллера сессий ESBC (далее ESBC или устройство).

1.2 Целевая аудитория

Данное руководство пользователя предназначено для технического персонала, выполняющего установку, настройку и мониторинг устройств посредством интерфейса командной строки (CLI), а также процедуры по обслуживанию системы и обновлению ПО. Квалификация технического персонала предполагает знание основ работы стеков протоколов TCP/IP, принципов построения Ethernet-сетей.

1.3 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 возможные обязательные параметры. Необходимо выбрать один из параметров.
«,» **«-»**	Данные знаки в описании команды используются для указания диапазонов.
«|»	Данный знак в описании команды обозначает «или».
<Полужирный курсив>	Полужирным курсивом в угловых скобках указываются названия клавиш на клавиатуре.
`Текст в рамке`	В рамках с текстом указаны примеры и результаты выполнения команд. В данном руководстве, помимо примеров для ESBC, могут встречаться примеры для маршрутизаторов ESR. Эти примеры эквивалентны.

1.4 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.

Информация содержит справочные данные об использовании устройства.

2 Описание изделий

- Назначение
- Функции
  - Функции интерфейсов
  - Функции при работе с MAC-адресами
  - Функции второго уровня сетевой модели OSI
  - Функции третьего уровня сетевой модели OSI
  - Функции туннелирования трафика
  - Функции управления и конфигурирования
  - Функции сетевой защиты
- Основные технические характеристики
- Конструктивное исполнение
  - Конструктивное исполнение ESBC-3200
  - Световая индикация
- Комплект поставки

2.1 Назначение

Пограничный контроллер сессий ESBC предназначен для решения задач сопряжения разнородных VoIP-сетей, обеспечивая совместную работу терминалов с различными протоколами сигнализации и наборами используемых кодеков. Кроме того, за счет функциональности Firewall, NAT и проксирования сигнального и медиатрафика он защищает корпоративную сеть от атак и скрывает ее внутреннюю структуру. ESBC всегда устанавливается на границе корпоративной или операторской VoIP-сети и выполняет те функции, которые нецелесообразно возлагать на устройства оператора (например, гибкий коммутатор Softswitch).

Устройства серии ESBC являются высокопроизводительными многоцелевыми сетевыми устройствами, которые объединяют в себе традиционные сетевые функции и комплексный многоуровневый подход к безопасности маршрутизации, что позволяет обеспечить надежную защиту для корпоративной среды.

Устройства поддерживают функции межсетевого экрана для защиты сети организации и своей сетевой инфраструктуры, а также сочетают в себе новейшие средства обеспечения безопасности данных, шифрования, аутентификации и защиты от вторжений.

Устройства содержат в себе средства для программной и аппаратной обработки данных. За счет оптимального распределения функций обработки данных между частями устройства достигается максимальная производительность.

2.2 Функции

2.2.1 Функции интерфейсов

В таблице 1 приведен список функций интерфейсов устройства.

Таблица 1 – Функции интерфейсов устройства

Определение полярности подключения кабеля (Auto MDI/MDIX)	Автоматическое определение типа кабеля – перекрестный кабель или кабель прямого подключения. • MDI (Medium Dependent Interface – прямой) – стандарт кабелей для подключения оконечных устройств; • MDIX (Medium Dependent Interface with Crossover – перекрестный) – стандарт кабелей для подключения концентраторов и коммутаторов.
Поддержка обратного давления (Back pressure)	Метод обратного давления используется на полудуплексных соединениях для регулирования потока данных от встречного устройства путем создания коллизий. Метод позволяет избежать переполнения буферной памяти устройства и потери данных.
Агрегирование каналов (LAG, Link aggregation)	Агрегирование (объединение) каналов позволяет увеличить пропускную способность канала связи и повысить его надежность. Пограничный контроллер сессий поддерживает статическое и динамическое агрегирование каналов. При динамическом агрегировании используется протокол LACP для управления группой каналов.

2.2.2 Функции при работе с MAC-адресами

В таблице 2 приведены функции устройства при работе с MAC-адресами.

Таблица 2 – Функции работы с MAC-адресами

Таблица MAC-адресов	Таблица MAC-адресов устанавливает соответствие между MAC-адресами и интерфейсами устройства и используется для маршрутизации пакетов данных. Пограничные контроллеры сессий имеют таблицу емкостью до 128k MAC-адресов и резервируют определенные MAC-адреса для использования системой.
Режим обучения	MAC-таблица может содержать либо статические адреса, либо адреса, изученные при прохождении пакетов данных через устройство. Изучение происходит за счет регистрации MAC-адресов отправителей пакетов с привязкой их к портам и VLAN. Впоследствии эти данные используются для маршрутизации встречных пакетов. Время хранения зарегистрированных MAC-адресов ограничено, его продолжительность может настраиваться администратором. Если MAC-адрес получателя, указанный в принятом устройством пакете, отсутствует в таблице, то такой пакет отправляется далее как широковещательный в пределах L2-сегмента сети.

2.2.3 Функции второго уровня сетевой модели OSI

В таблице 3 приведены функции и особенности второго уровня (уровень 2 OSI).

Таблица 3 – Описание функций второго уровня (уровень 2 OSI)

Поддержка VLAN	VLAN (Virtual Local Area Network) – это средство разделения сети на изолированные сегменты на уровне L2. Использование VLAN позволяет повысить устойчивость работы крупных сетей за счет деления их на более мелкие сети, изолировать разнородный трафик данных между собой и решить многие другие задачи. Пограничные контроллеры сессий поддерживают различные способы организации VLAN: • VLAN на базе меток пакетов данных, в соответствии с IEEE 802.1Q; • VLAN на базе портов устройства (port-based); • VLAN на базе использования правил классификации данных (policy-based).
Протокол связующего дерева (Spanning Tree Protocol)	Задачей протокола Spanning Tree является исключение избыточных сетевых соединений и приведение топологии сети к древовидной. Основные применения протокола связаны с предотвращением зацикливания сетевого трафика и с организацией резервных каналов связи.

2.2.4 Функции третьего уровня сетевой модели OSI

В таблице 4 приведены функции третьего уровня (уровень 3 OSI).

Таблица 4 – Описание функций третьего уровня (Layer 3)

Статические IP-маршруты	Администратор пограничного контроллера сессий имеет возможность добавлять и удалять статические записи в таблицу маршрутизации.
Динамическая маршрутизация	Протоколы динамической маршрутизации позволяют устройству обмениваться маршрутной информацией с соседними пограничными контроллерами сессий и автоматически составлять таблицу маршрутов. Пограничный контроллер сессий поддерживает следующие протоколы: RIPv2, RIPng, OSPFv2, OSPFv3, IS-IS, BGP.
Таблица ARP	ARP (Address Resolution Protocol) – протокол для выяснения соответствия адресов сетевого и канального уровней. Таблица ARP содержит информацию об изученном соответствии. Соответствие устанавливается на основе анализа ответов от сетевых устройств, адреса устройств запрашиваются с помощью широковещательных пакетов.
Клиент DHCP	Протокол DHCP (Dynamic Host Configuration Protocol) даёт возможность автоматизировать управление сетевыми устройствами. Клиент DHCP позволяет пограничному контроллеру сессий получать сетевой адрес и дополнительные параметры от внешнего DHCP-сервера. Как правило, этот способ используется для получения сетевых настроек оператора публичной сети (WAN).

Сервер DHCP	Сервер DHCP предназначен для автоматизации и централизации конфигурирования сетевых устройств. Размещение DHCP-сервера на пограничном контроллере сессий позволяет получить законченное решение для поддержки локальной сети. DHCP-сервер, входящий в состав пограничного контроллера сессий, позволяет назначать IP-адреса сетевым устройствам и передавать дополнительные сетевые параметры – адреса серверов, адреса шлюзов сети и другие необходимые параметры.
DHCP Relay	Функция DHCP Relay предназначена для перенаправления широковещательных DHCP Discover-пакетов из одного широковещательного домена в одноадресные (unicast) DHCP Discover-пакеты в другом широковещательном домене.
Трансляция сетевых адресов (NAT, Network Address Translation)	Трансляция сетевых адресов – это механизм, который позволяет преобразовывать IP-адреса и номера портов транзитных пакетов. Функция NAT позволяет использовать меньшее количество IP-адресов, транслируя несколько IP-адресов внутренней сети в один внешний публичный IP-адрес. Использование NAT позволяет увеличить защищённость локальной сети за счёт скрытия её внутренней структуры. Пограничные контроллеры сессий поддерживают следующие варианты NAT: • Source NAT (SNAT) – выполняется замена адреса, а также номера порта источника при прохождении пакета в одну сторону и обратной замене адреса назначения в ответном пакете; • Destination NAT (DNAT) – когда обращения извне транслируются пограничным контроллером сессий на компьютер пользователя в локальной сети, имеющий внутренний адрес и потому недоступный извне сети непосредственно (без NAT).

2.2.5 Функции туннелирования трафика

Таблица 5 – Функции туннелирования трафика

Протоколы туннелирования	Туннелирование – это способ преобразования пакетов данных при передаче их по сети, при котором происходит замена, модификация или добавление нового сетевого заголовка пакета. Такой способ может быть использован для согласования транспортных протоколов при прохождении данных через транзитную сеть, для создания защищенных соединений, при которых туннелированные данные подвергаются шифрованию. Пограничные контроллеры сессий поддерживают следующие виды туннелей: • GRE – инкапсуляция IP-пакета/Ethernet-кадра в другой IP-пакет с добавлением GRE (General Routing Encapsulation)-заголовка; • IPv4-IPv4 – туннель, использующий инкапсуляцию исходных IP-пакетов в IP-пакеты с другими сетевыми параметрами; • L2TPv3 – туннель для передачи L2-трафика с помощью IP-пакетов; • IPsec – туннель с шифрованием передаваемых данных; • L2TP, PPTP, PPPoE, OpenVPN, WireGuard – туннели, использующиеся для организации удаленного доступа клиент-сервер.

2.2.6 Функции управления и конфигурирования

Таблица 6 – Основные функции управления и конфигурирования

Загрузка и выгрузка файла настройки	Параметры устройства сохраняются в файле настройки, который содержит данные конфигурации как всей системы в целом, так и определенного порта устройства. Для передачи файлов могут использоваться протоколы TFTP, FTP, SCP.
Интерфейс командной строки (CLI)	Управление посредством CLI осуществляется локально через последовательный порт RS-232 либо удаленно через Telnet, SSH. Интерфейс командной строки консоли (CLI) является промышленным стандартом. Интерпретатор CLI предоставляет список команд и ключевых слов для помощи пользователю и сокращению объема вводимых данных.
Syslog	Протокол Syslog обеспечивает передачу информационных сообщений о происходящих в системе событиях и ведение журнала событий.
Сетевые утилиты ping, traceroute	Утилиты ping и traceroute – предназначены для проверки доступности сетевых устройств и для определения маршрутов передачи данных в IP-сетях.
Управление контролируемым доступом – уровни привилегий	Пограничные контроллеры сессий поддерживают управление уровнем доступа пользователей к системе. Уровни доступа позволяют управлять зонами ответственности администраторов устройств. Уровни доступа нумеруются от 1 до 15, уровень 15 соответствует полному доступу к управлению устройством.
Аутентификация	Аутентификация – это процедура проверки подлинности пользователя. Пограничные контроллеры сессий поддерживают следующие методы аутентификации: • локальная – для аутентификации используется локальная база данных пользователей, хранящаяся на самом устройстве; • групповая – база данных пользователей хранится на сервере аутентификации. Для взаимодействия с сервером используются протоколы RADIUS и TACACS.
Сервер SSH/ сервер Telnet	Функции сервера SSH и Telnet позволяют установить соединение с устройством для управления им.
Автоматическое восстановление конфигурации	Устройство поддерживает автоматическую систему восстановления конфигурации, которая предотвращает ситуации потери удаленного доступа к устройству после смены конфигурации. Если в течение заданного времени после изменения конфигурации не было введено подтверждение – произойдет автоматический откат конфигурации до предыдущего использовавшегося состояния.

2.2.7 Функции сетевой защиты

В таблице 7 приведены функции сетевой защиты, выполняемые устройством.

Таблица 7 − Функции сетевой защиты

Зоны безопасности	Все интерфейсы пограничного контроллера сессий распределяются по зонам безопасности. Для каждой пары зон настраиваются правила, определяющие возможность или невозможность прохождения данных между зонами, правила фильтрации трафика данных.
Фильтрация данных	Для каждой пары зон безопасности составляется набор правил, которые позволяют управлять фильтрацией данных, проходящих через пограничные контроллеры сессий. Командный интерфейс устройства предоставляет средства для детальной настройки правил классификации трафика и для назначения результирующего решения о пропуске трафика.

2.3 Основные технические характеристики

Основные технические параметры пограничного контроллера сессий приведены в таблице 8.

Таблица 8 – Основные технические характеристики

Общие параметры	
Интерфейсы	12 × 1000BASE-X/10GBASE-R/25GBASE-R 1 × Консольный порт RS-232 (RJ-45) 1 × Порт OOB 1 × USB 2.0 1 × Слот для microSD-карты
Типы оптических трансиверов	1000BASE-X SFP 10GBASE-R SFP+ 25GBASE-R SFP28
Дуплексный и полудуплексный режимы интерфейсов	• дуплексный и полудуплексный режимы для электрических портов • дуплексный режим для оптических портов
Скорость передачи данных	• оптические интерфейсы 1/10/25 Гбит/с
Количество VPN-туннелей	500
Количество статических маршрутов	11k
Максимальное количество конкурентных сессий	8,5M
Таблица VLAN	4094
Количество маршрутов BGPv4/ BGPv6	5M
Количество маршрутов OSPFv2/OSPFv3/IS-IS	500k
Количество маршрутов RIP/ RIPng	10k
Размер базы FIB	1,7M
VRF	32
Количество L3-интерфейсов	4000

Соответствие стандартам	IEEE 802.3 10BASE-T Ethernet IEEE 802.3u 100BASE-T Fast Ethernet IEEE 802.3ab 1000BASE-T Gigabit Ethernet IEEE 802.3z Fiber Gigabit Ethernet IEEE 802.3cc 25GBASE-LR Ethernet IEEE 802.3by 25GBASE-SR Ethernet ANSI/IEEE 802.3 автоопределение скорости IEEE 802.3x контроль потоков данных IEEE 802.3ad объединение каналов LACP IEEE 802.1Q виртуальные локальные сети VLAN IEEE 802.1v, IEEE 802.3ac, IEEE 802.3ae, IEEE 802.1D, IEEE 802.1w, IEEE 802.1s
Управление	
Локальное управление	CLI
Удаленное управление	Telnet, SSH
Физические характеристики и условия окружающей среды	
	Сеть переменного тока: 100–240 В, 50–60 Гц Сеть постоянного тока: 36–72 В Варианты питания: • один источник питания постоянного или переменного тока; • два источника питания постоянного или переменного тока с возможностью горячей замены.
Максимальная потребляемая мощность	118 Вт
Масса	5,3 кг
Габаритные размеры (Ш × В × Г)	430 × 44 × 330 мм
Интервал рабочих температур	от -10 до +45 °С
Интервал температуры хранения	от -40 до +70 °С
Относительная влажность при эксплуатации (без образования конденсата)	не более 80 %

Относительная влажность при хранении (без образования конденсата)	от 10 до 95 %
Срок службы	не менее 15 лет

2.4 Конструктивное исполнение

В данном разделе описано конструктивное исполнение устройства. Представлены изображения передней, задней и боковых панелей устройства. Описаны разъемы, светодиодные индикаторы и органы управления.

2.4.1 Конструктивное исполнение ESBC-3200

Устройство выполнено в металлическом корпусе с возможностью установки в 19” конструктив, высота корпуса 1U.

Передняя панель устройства ESBC-3200

Внешний вид передней панели ESBC-3200 показан на рисунке 1.

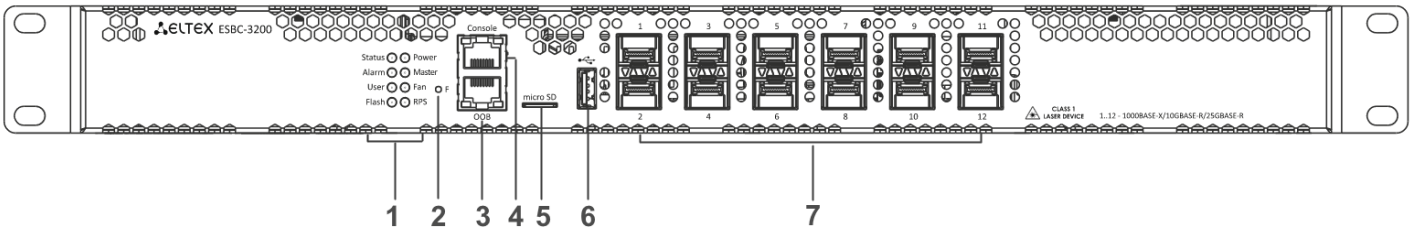


Рисунок 1 – Передняя панель ESBC-3200

Таблица 9 – Описание разъемов, индикаторов и органов управления передней панели ESBC-3200

№	Элемент передней панели	Описание
1	Status	Индикатор текущего состояния устройства.
	Alarm	Индикатор наличия и уровня аварии устройства.
	User	Индикатор, использующийся для пользовательских сценариев, которые можно назначить в режиме конфигурирования устройства.
	Flash	Индикатор активности обмена с накопителем данных – microSD-картой или USB Flash.
	Power	Индикатор питания устройства.
	Master	Индикатор работы устройства в failover-режимах (не используется в текущей версии).
	Fan	Индикатор аварии вентиляторов.

№	Элемент передней панели	Описание
	RPS	Индикатор резервного источника электропитания.
2	F	Функциональная кнопка для перезагрузки устройства и сброса к заводским настройкам: • при удержании кнопки менее 10 секунд происходит перезагрузка устройства; • при удержании кнопки более 10 секунд происходит перезагрузка устройства и сброс к заводским настройкам.
3	OOB	Ethernet-порт используется только для обновления программного обеспечения через вторичный загрузчик U-Boot. Данный порт не используется для удаленного доступа и управления. Данный интерфейс не может участвовать в маршрутизации транзитного трафика.
4	Console	Консольный порт RS-232 (RJ-45) для локального управления устройством.
5	microSD	Разъем для установки microSD-карт памяти.
6	USB1	Порт USB 2.0 для подключения USB-устройств.
7	[1 .. 12]	Слоты для установки трансиверов 25G SFP28/10G SFP+/1G SFP.

Задняя панель устройства ESBC-3200

Внешний вид задней панели ESBC-3200 приведен на рисунке 2.

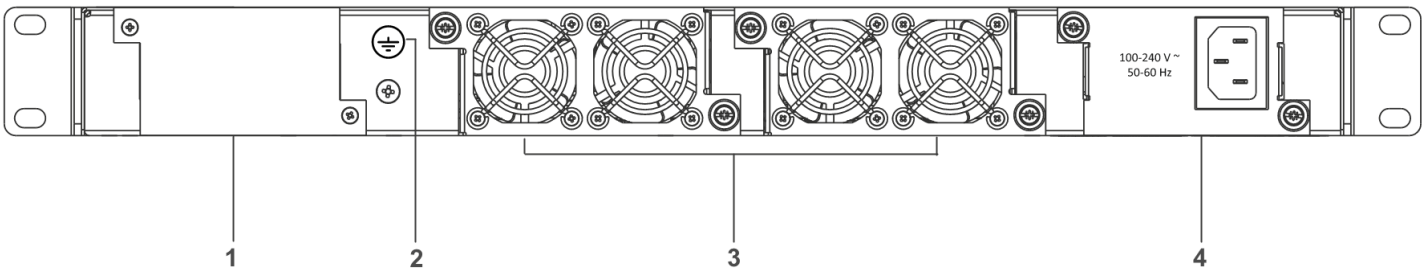


Рисунок 2 – Задняя панель ESBC-3200

Таблица 10 – Описание разъемов задней панели ESBC-3200

№	Описание
1	Основной источник питания.
2	Клемма для заземления устройства.
3	Съемные вентиляционные модули с возможностью горячей замены.
4	Место для установки резервного источника питания.

Боковые панели устройства ESBC-3200

Внешний вид боковых панелей приведен на рисунках ниже.

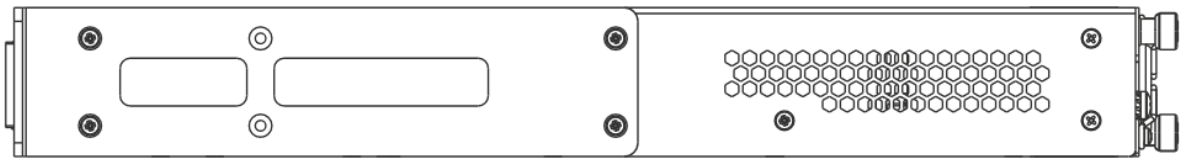


Рисунок 3 – Правая боковая панель ESBC-3200

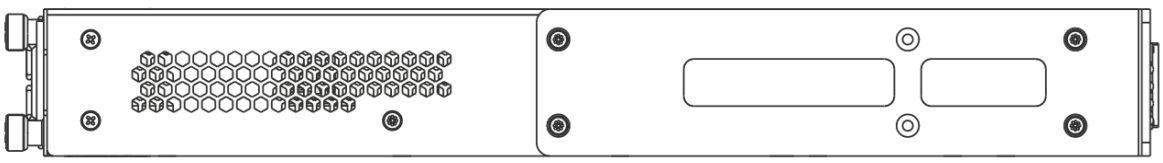


Рисунок 4 – Левая боковая панель ESBC-3200

На боковых панелях устройства расположены вентиляционные решетки, которые служат для отвода тепла. Не закрывайте вентиляционные отверстия посторонними предметами. Это может привести к перегреву компонентов устройства и вызвать нарушения в его работе. Рекомендации по установке устройства расположены в разделе Установка и подключение.

2.4.2 Световая индикация

Световая индикация ESBC-3200

Состояние интерфейсов Ethernet отображается двумя светодиодными индикаторами – *LINK/ACT* зеленого цвета и *SPEED* янтарного цвета. Расположение светодиодных индикаторов интерфейсов показано ниже на рисунках 5 и 6. Значения световой индикации описаны в таблицах 11 и 12 соответственно.

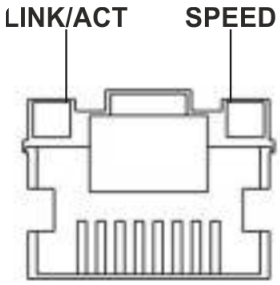


Рисунок 5 – Расположение индикаторов разъема RJ-45

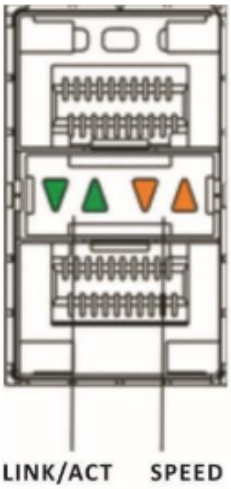


Рисунок 6 – Расположение индикаторов состояния SFP/SFP+/SFP28-интерфейсов

Таблица 11 – Световая индикация состояния RJ-45 интерфейсов

Свечение индикатора SPEED	Свечение индикатора LINK/ACT	Состояние интерфейса Ethernet
Выключен	Выключен	Порт выключен или соединение не установлено.
Выключен	Горит постоянно зелёным	Установлено соединение на скорости 10 или 100 Мбит/с.
Горит постоянно янтарным	Горит постоянно зелёным	Установлено соединение на скорости 1000 Мбит/с.
X	Мигает	Идет передача данных.

Таблица 12 – Световая индикация состояния состояния SFP/SFP+/SFP28-интерфейсов

Свечение индикатора SPEED	**Свечение индикатора LINK/ ACT**	**Состояние интерфейса Ethernet**
Выключен	Выключен	Порт выключен или соединение не установлено.
Выключен	Горит постоянно зелёным	Установлено соединение на скорости 1 Гбит/с.
Горит постоянно янтарным	Горит постоянно зелёным	Установлено соединение на скорости 10 Гбит/с.
X	Мигает	Идет передача данных.

В следующей таблице приведено описание состояний системных индикаторов устройства и их значений.

Таблица 13 – Состояния системных индикаторов

Название индикатора	**Функция индикатора**	**Состояние индикатора**	**Состояние устройства**
Status	Индикатор текущего состояния устройства.	Зеленый	Устройство работает нормально.
		Мигает зеленым	Устройство находится в состоянии загрузки ПО.
Alarm	Индикатор наличия и уровня аварии устройства.	-	-
User	Индикатор пользовательских сценариев.	-	-
Flash	Индикатор активности обмена с накопителем данных: SD-картой или USB Flash.	Зеленый	Выполнение операций чтения/ записи по команде «сору».
Power	Индикатор питания устройства.	Зеленый	Питание устройства в норме. Основной источник питания, если он установлен, работает нормально.
		Красный	Неработоспособность основного источника питания, авария или отсутствие первичной сети.
		Выключен	Отказ внутренних источников питания устройства.
Master	Индикатор работы устройства в failover-режимах.	-	-
Fan	Состояние вентилятора охлаждения.	Выключен	Все вентиляторы исправны.

Название индикатора	Функция индикатора	Состояние индикатора	Состояние устройства
		Красный	Отказ одного или более вентиляторов. Причиной возникновения аварии может быть неработоспособность хотя бы одного из вентиляторов – остановка или пониженная частота оборотов.
RPS	Режим работы резервного источника питания.	Зеленый	Резервный источник установлен и исправен.
		Выключен	Резервный источник не установлен.
		Красный	Отсутствие первичного питания резервного источника или его неисправность.

2.5 Комплект поставки

В базовый комплект поставки ESBC-3200 входят:

- пограничный контроллер сессий ESBC-3200;
- консольный кабель;
- комплект для крепления устройства в стойку 19”;
- сертификат соответствия;
- паспорт;
- документация (опционально);
- памятка о документации.

⚠ По заказу покупателя для ESBC-3200 в комплект поставки может быть включен модуль питания (PM160-220/12).

⚠ По заказу покупателя в комплект поставки могут быть включены SFP/SFP+-трансиверы.

3 Установка и подключение

- Установка устройства в стойку
- Установка модулей питания
- Подключение питающей сети
- Установка и удаление SFP-трансиверов
  - Установка трансивера
  - Удаление трансивера
- Подключение к vESBC

В данном разделе описаны процедуры установки пограничного контроллера сессий в стойку и подключения к питающей сети.

3.1 Установка устройства в стойку

Для установки устройства в стойку:

1. Выберите необходимое положение кронштейна (рисунок 7). Совместите четыре отверстия кронштейна с четырьмя отверстиями на боковой панели устройства. С помощью отвертки прикрепите кронштейн винтами к корпусу.
2. Повторите шаг 1 для другой боковой панели устройства.
3. Совместите отверстия кронштейнов с отверстиями на передних вертикальных направляющих стойки. Используйте отверстия в направляющих на одном уровне с обеих сторон стойки, для того чтобы устройство располагалось горизонтально.
4. С помощью отвертки прикрепите устройство к стойке винтами.

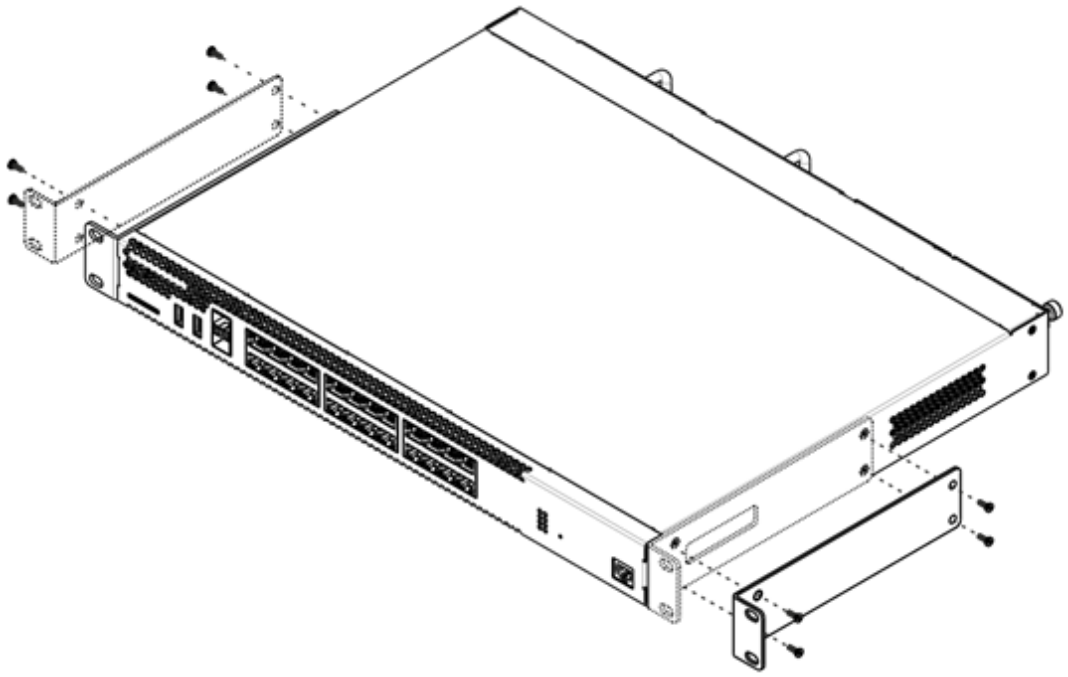


Рисунок 7 – Крепление кронштейнов к ESBC-3200

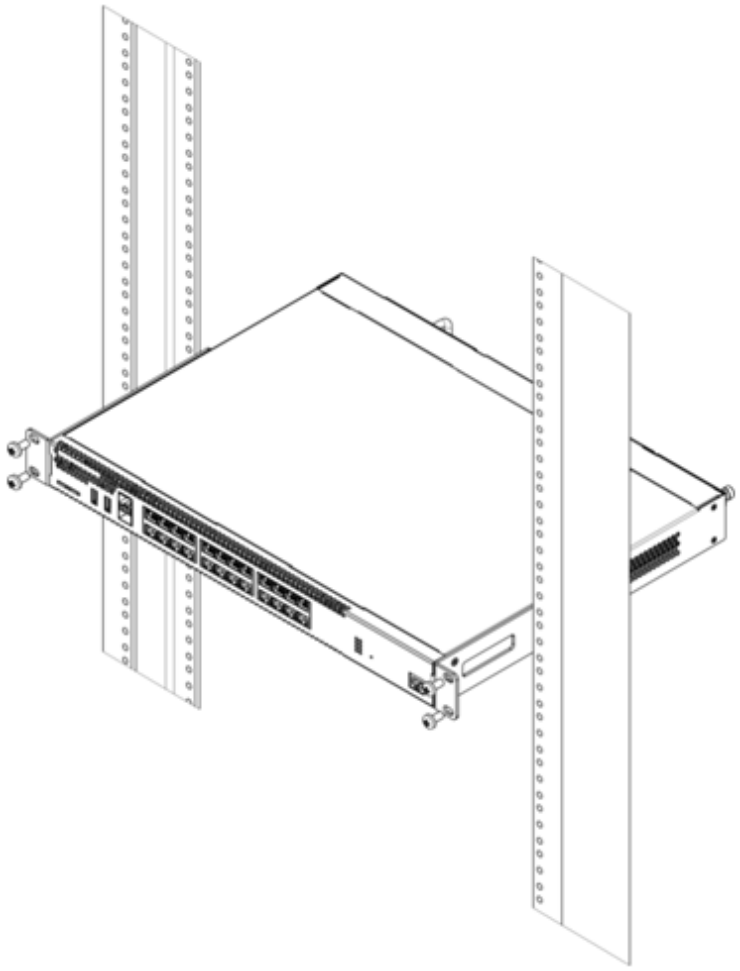


Рисунок 8 – Установка ESBC-3200 в стойку

Вентиляция устройства организована по схеме фронт-тыл. На передней и боковых панелях устройства расположены вентиляционные отверстия, с задней стороны устройства расположены вентиляционные модули. Не закрывайте входные и выходные вентиляционные отверстия посторонними предметами во избежание перегрева компонентов устройства и нарушения его работы.

3.2 Установка модулей питания

Пограничные контроллеры сессий ESBC-3200 могут работать с одним или двумя модулями питания. Установка второго модуля питания необходима в случае использования устройства в условиях, требующих повышенной надежности.

Места для установки модулей питания с электрической точки зрения равноценны. С точки зрения использования устройства, модуль питания может находиться в основном и резервном разъеме, информация о приоритетности находится в таблице Описание разъемов задней панели ESBC-3200. Модули питания могут устанавливаться и извлекаться без выключения устройства. При установке или извлечении дополнительного модуля питания пограничный контроллер сессий продолжает работу без перезапуска.

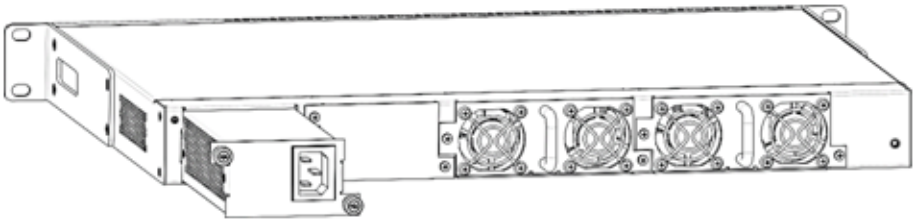


Рисунок 9 – Установка модулей питания

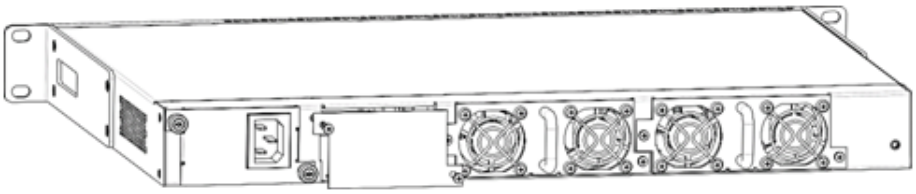


Рисунок 10 – Установка заглушки

Индикация аварии модуля питания может быть вызвана не только отказом модуля, но и отсутствием первичного питания.

Состояние модулей питания может быть проверено по индикации на передней панели устройства (см. раздел Световая индикация) или по диагностике, доступной через интерфейсы управления.

3.3 Подключение питающей сети

1. Прежде чем к устройству будет подключена питающая сеть, необходимо заземлить корпус устройства через заземляющий винт М4. Заземление необходимо выполнять изолированным многожильным проводом. Устройство заземления и сечение заземляющего провода должны соответствовать требованиями Правил устройства электроустановок (ПУЭ).
2. Если предполагается подключение компьютера или иного оборудования к консольному порту пограничного контроллера сессий, это оборудование также должно быть надежно заземлено.
3. Подключите к устройству кабель питания. В зависимости от комплектации устройства, питание может осуществляться от сети переменного тока либо от сети постоянного тока. При подключении сети переменного тока следует использовать кабель, входящий в комплект устройства. Для подключения к сети постоянного тока используйте провод сечением не менее 1 мм^2.
4. Включите питание устройства и убедитесь в отсутствии аварий по состоянию индикаторов на передней панели.

3.4 Установка и удаление SFP-трансиверов

⚠ Установка оптических модулей может производиться как при выключенном, так и при включенном устройстве.

3.4.1 Установка трансивера

1. Вставьте верхний SFP-модуль в слот открытой частью разъема вниз, а нижний SFP-модуль – открытой частью разъема вверх.

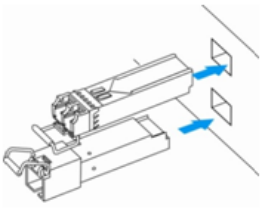


Рисунок 11 – Установка SFP-трансиверов

2. Надавите на модуль по направлению внутрь корпуса устройства до появления характерного щелчка фиксации модуля.

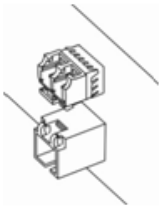


Рисунок 12 – Установленные SFP-трансиверы

3.4.2 Удаление трансивера

1. Откиньте рукоятку модуля, это приведет к разблокированию удерживающей защелки.

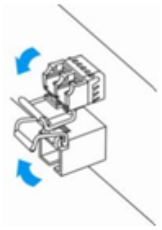


Рисунок 13 – Открытие защелки SFP-трансиверов

2. Извлеките модуль из слота.

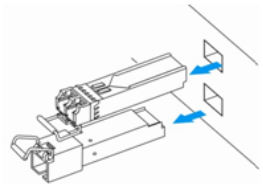


Рисунок 14 – Извлечение SFP-трансиверов

3.5 Подключение к vESBC

Для получения информации об установке и подключении к vESBC перейдите в раздел документации vESBC. Руководство по установке и настройке. Версия 1.6.0.

4 Интерфейсы управления

- Интерфейс командной строки (CLI)
- Типы и порядок именования пограничного контроллера сессий
- Типы и порядок именования туннелей пограничного контроллера сессий

Настройка и мониторинг устройства могут осуществляться через различные интерфейсы управления.

Для доступа к устройству может использоваться сетевое подключение по протоколам Telnet и SSH или прямое подключение через консольный порт, соответствующий спецификации RS-232. При доступе по протоколам Telnet, SSH и при подключении через консольный порт для управления устройством используется интерфейс командной строки.

Только для ESBC-3200:
Заводская конфигурация содержит описание доверенной зоны trusted и IP-адрес для доступа к управлению устройством – 192.168.1.1/24. В доверенную зону входят интерфейсы: Twentyfivegigabitethernet 1/0/3-12.
Для ESBC-3200 и vESBC:
В заводской конфигурации по умолчанию создан пользователь «admin» с паролем «password».
Протоколы семейства STP (STP, RSTP, VSTP) отключены.
Заводскую конфигурацию можно сбросить командой *copy system:default-config system:candidate-config*. После сброса необходимо настроить ESBC с помощью консольного порта.

При использовании любого из перечисленных интерфейсов управления действуют единые принципы работы с конфигурацией. Должна соблюдаться определенная, описанная здесь, последовательность изменения и применения конфигурации, позволяющая защитить устройство от некорректного конфигурирования.

4.1 Интерфейс командной строки (CLI)

Интерфейс командной строки (Command Line Interface, CLI) – интерфейс, предназначенный для управления, просмотра состояния и мониторинга устройства. Для работы потребуется любая установленная на ПК программа, поддерживающая работу по протоколам Telnet, SSH или прямое подключение через консольный порт (например, HyperTerminal).

Интерфейс командной строки обеспечивает авторизацию пользователей и ограничивает их доступ к командам на основании уровня доступа, заданного администратором.

В системе может быть создано необходимое количество пользователей, права доступа задаются индивидуально для каждого из них.

Для обеспечения безопасности командного интерфейса все команды разделены на две категории – привилегированные и непривилегированные. К привилегированным в основном относятся команды конфигурирования. К непривилегированным – команды мониторинга.

Система позволяет нескольким пользователям одновременно подключаться к устройству.

4.2 Типы и порядок именования пограничного контроллера сессий

При работе пограничного контроллера сессий используются сетевые интерфейсы различного типа и назначения. Система именования позволяет однозначно адресовать интерфейсы по их функциональному назначению и местоположению в системе. Далее в таблице приведен перечень типов интерфейсов.

Таблица 14 – Типы и порядок именования интерфейсов пограничного контроллера сессий

Тип интерфейса	Обозначение
Физические интерфейсы	Обозначение физического интерфейса включает в себя его тип и идентификатор. Идентификатор физических интерфейсов имеет вид **<UNIT>/<SLOT>/<PORT>**, где: • **<UNIT>** – номер устройства в группе устройств, • **<SLOT>** – номер модуля в составе устройства или 0 при отсутствии деления устройства на модули, • **<PORT>** – порядковый номер порта.
Порты 1 Гбит/с	**gigabitethernet <UNIT>/<SLOT>/<PORT>** Пример обозначения: **gigabitethernet 1/0/12** ⚠ Допускается использовать сокращенное наименование, например gi1/0/12.
Порты 10 Гбит/с	**tengigabitethernet <UNIT>/<SLOT>/<PORT>** Пример обозначения: **tengigabitethernet 1/0/2** ⚠ Допускается использовать сокращенное наименование, например te1/0/2.
Порты 25 Гбит/с	**twentyfivegigabitethernet <UNIT>/<SLOT>/<PORT>** Пример обозначения: **twentyfivegigabitethernet 1/0/2** ⚠ Допускается использовать сокращенное наименование, например twe1/0/2.
Порты 40 Гбит/с	**fortygigabitethernet <UNIT>/<SLOT>/<PORT>** Пример обозначения: **fortygigabitethernet 1/0/2** ⚠ Допускается использовать сокращенное наименование, например fo1/0/2.
Группы агрегации каналов	Обозначение группы агрегации каналов включает в себя его тип и порядковый номер интерфейса: **port-channel <CHANNEL_ID>** Пример обозначения: **port-channel 6** ⚠ Допускается использовать сокращенное наименование, например, po1.

Тип интерфейса	Обозначение
Саб-интерфейсы	Обозначение саб-интерфейса образуется из обозначения базового интерфейса и идентификатора (VLAN) саб-интерфейса, разделенных точкой. Примеры обозначений: • **gigabitethernet 1/0/12.100** • **tengigabitethernet 1/0/2.123** • **twentyfivegigabitethernet 1/0/2.200** • **fortygigabitethernet 1/0/2.1024** • **port-channel 1.6** ⚠ Идентификатор саб-интерфейса может принимать значения [2..4094].
Q-in-Q интерфейсы	Обозначение Q-in-Q интерфейса образуется из обозначения базового интерфейса, идентификатора сервисного VLAN и идентификатора пользовательского VLAN, разделенных точкой. Примеры обозначений: • **gigabitethernet 1/0/12.100.10** • **tengigabitethernet 1/0/2.45.12** • **twentyfivegigabitethernet 1/0/2.100.200** • **fortygigabitethernet 1/0/2.408.507** • **port-channel 1.6.34** ⚠ Идентификатор сервисного и пользовательского VLAN может принимать значения [1..4094].
Логические интерфейсы	Обозначение логического интерфейса является порядковым номером интерфейса: Примеры обозначений: • **loopback 4** • **bridge 60** • **service-port 1**

⚠ 1. Количество интерфейсов каждого типа зависит от модели пограничного контроллера сессий..
2. Текущая версия ПО поддерживает кластеризацию устройств единой модели. Номер unit в группе устройств может принимать значение 1 или 2.
3. Некоторые команды поддерживают одновременную работу с группой интерфейсов. Для указания группы интерфейсов может быть использовано перечисление через запятую или указание диапазона идентификаторов через дефис «-».
Примеры указания групп интерфейсов:

```
interface gigabitethernet 1/0/1, gigabitethernet 1/0/5
interface tengigabitethernet 1/0/1-2
interface twentyfivegigabitethernet 1/0/3-4
interface fortygigabitethernet 1/0/1-2
interface gi1/0/1-3,gi1/0/7,te1/0/1,fo1/0/1
```

4.3 Типы и порядок именования туннелей пограничного контроллера сессий

При работе пограничного контроллера сессий используются сетевые туннели различного типа и назначения. Система именования позволяет однозначно адресовать туннели по их функциональному назначению. Далее в таблице приведен перечень типов туннелей.

Таблица 15 – Типы и порядок именования туннелей пограничного контроллера сессий

Тип туннеля	Обозначение
L2TP-туннель	Обозначение L2TP-туннеля состоит из обозначения типа и порядкового номера туннеля: **l2tp <L2TP_ID>** Пример обозначения: **l2tp 1**
L2TPv3-туннель	Обозначение L2TPv3-туннеля состоит из обозначения типа и порядкового номера туннеля: **l2tpv3 <L2TPV3_ID>** Пример обозначения: **l2tpv3 1**
GRE-туннель	Обозначение GRE-туннеля состоит из обозначения типа и порядкового номера туннеля: **gre <GRE_ID>** Пример обозначения: **gre 1**
SoftGRE-туннель	Обозначение SoftGRE-туннеля состоит из обозначения типа, порядкового номера туннеля и, опционально, VLAN ID виртуального интерфейса: **softgre <GRE_ID>[.<VLAN>]** Примеры обозначения: **softgre 1, softgre 1.10**
IPv4-over-IPv4-туннель	Обозначение IPv4-over-IPv4-туннеля состоит из обозначения типа и порядкового номера туннеля: **ip4ip4 <IPIP_ID>** Пример обозначения: **ip4ip4 1**
IPsec-туннель	Обозначение виртуального IPsec-туннеля состоит из обозначения типа и порядкового номера туннеля: **vti <VTI_ID>** Пример обозначения: **vti 1**
Логический туннель (туннель между VRF)	Обозначение логического туннеля состоит из обозначения типа и порядкового номера туннеля: **lt <LT_ID>** Пример обозначения: **lt 1**
PPPoE-туннель	Обозначение PPPoE-туннеля состоит из обозначения типа и порядкового номера туннеля: **pppoe <PPPOE_ID>** Пример обозначения: **pppoe 1**

Тип туннеля	Обозначение
OpenVPN-туннель	Обозначение OpenVPN-туннеля состоит из обозначения типа и порядкового номера туннеля: **openvpn <OPENVPN_ID>** Пример обозначения: **openvpn 1**
PPTP-туннель	Обозначение PPTP-туннеля состоит из обозначения типа и порядкового номера туннеля: **pptp <PPTP_ID>** Пример обозначения: **pptp 1**

⚠ Количество туннелей каждого типа зависит от модели и ПО пограничного контроллера сессий.

5 Начальная настройка устройства

- Заводская конфигурация устройства (только для ESBC-3200)
  - Описание заводской конфигурации
- Подключение и конфигурирование устройства
  - Подключение к устройству
    - Подключение по локальной сети Ethernet
    - Подключение через консольный порт RS-232
  - Применение изменения конфигурации
  - Базовая настройка устройства
    - Изменение пароля пользователя «admin»
    - Создание новых пользователей
    - Назначение имени устройства
    - Настройка параметров публичной сети
    - Настройка удаленного доступа к устройству

5.1 Заводская конфигурация устройства (только для ESBC-3200)

При отгрузке устройства клиенту на пограничном контроллере сессий будет загружена заводская конфигурация, которая включает минимально необходимые базовые настройки. Заводская конфигурация позволяет использовать пограничный контроллер сессий в качестве шлюза с функцией SNAT без необходимости применять дополнительные настройки. Кроме того, заводская конфигурация содержит настройки, позволяющие получить сетевой доступ к устройству для выполнения расширенного конфигурирования.

Заводскую конфигурацию можно сбросить командой *copy system:default-config system:candidate-config*. После сброса необходимо настроить ESBC-3200 с помощью консольного порта.

5.1.1 Описание заводской конфигурации

Для подключения к сетям в конфигурации описаны 2 зоны безопасности с наименованиями «Trusted» для локальной сети и «Untrusted» для публичной сети. Все интерфейсы разделены между двух зон безопасности:

1. **Зона «Untrusted»** предназначена для подключения к публичной сети (WAN). В этой зоне открыты порты DHCP-протокола для получения динамического IP-адреса от провайдера. Все входящие соединения из данной зоны на пограничный контроллер сессий запрещены.
   В данную зону безопасности входят интерфейсы:
   - для ESBC-3200: Twentyfivegigabitethernet 1/0/1-2.

   Интерфейсы зоны объединены в один L2-сегмент через сетевой мост *Bridge 2*.
2. **Зона «Trusted»** предназначена для подключения к локальной сети (LAN). В этой зоне открыты порты протоколов Telnet и SSH для удаленного доступа, ICMP-протокола для проверки доступности пограничного контроллера сессий, DHCP-протокола для получения клиентами IP-адресов от устройства. Исходящие соединения из данной зоны в зону «Untrusted» разрешены.
   В данную зону безопасности входят интерфейсы:
   - для ESBC-3200: Twentyfivegigabitethernet 1/0/3-12.

   Интерфейсы зоны объединены в один L2-сегмент через сетевой мост *Bridge 1*.

На интерфейсе *Bridge 2* включен DHCP-клиент для получения динамического IP-адреса от провайдера. На интерфейсе *Bridge 1* сконфигурирован статический IP-адрес 192.168.1.1/24. Созданный IP-интерфейс выступает в качестве шлюза для клиентов локальной сети. Для клиентов локальной сети настроен DHCP-пул адресов 192.168.1.2-192.168.1.254 с маской 255.255.255.0. Для получения клиентами локальной сети доступа к Internet на пограничном контроллере сессий включен сервис Source NAT.

Политики зон безопасности настроены следующим образом (см. таблицу 16).

Таблица 16 – Описание политик зон безопасности

Зона, из которой идет трафик	Зона, в которую идет трафик	Тип трафика	Действие
Trusted	Untrusted	TCP, UDP, ICMP	разрешен
Trusted	Trusted	TCP, UDP, ICMP	разрешен
Trusted	self	TCP/22 (SSH), ICMP, UDP/67 (DHCP Server), UDP/123 (NTP)	разрешен
Untrusted	self	UDP/68 (DHCP Client)	разрешен

Для обеспечения возможности конфигурирования устройства при первом включении в конфигурации устройства создана учётная запись администратора "admin" с паролем "password".
Пользователю будет предложено изменить пароль администратора при начальном конфигурировании устройства.

Для сетевого доступа к управлению пограничным контроллером сессий при первом включении в конфигурации задан статический IP-адрес на интерфейсе Bridge 1 – 192.168.1.1/24.

5.2 Подключение и конфигурирование устройства

Пограничные контроллеры сессий ESBC-3200 предназначены для выполнения функций пограничного шлюза и обеспечения безопасности сети пользователя при подключении ее к публичным сетям передачи данных.

Базовая настройка устройства должна включать:

- назначение IP-адресов (статических или динамических) интерфейсам, участвующим в маршрутизации данных;
- создание зон безопасности и распределение интерфейсов по зонам;
- создание политик, регулирующих прохождение данных между зонами;
- настройка сервисов, сопутствующих маршрутизации данных (NAT, Firewall и прочие).

Расширенные настройки зависят от требований конкретной схемы применения устройства и легко могут быть добавлены или изменены с помощью имеющихся интерфейсов управления.

5.2.1 Подключение к устройству

Ниже описаны предусмотренные способы подключения к устройству.

Подключение по локальной сети Ethernet

⚠ При первоначальном старте устройство загружается с заводской конфигурацией. Описание заводской конфигурации приведено в разделе Заводская конфигурация устройства данного руководства.

Подключите сетевой кабель передачи данных (патч-корд) к любому порту, входящему в зону «***Trusted***», и к компьютеру, предназначенному для управления.

В заводской конфигурации пограничного контроллера сессий активирован DHCP-сервер с пулом IP-адресов в подсети **192.168.1.0/24**.

При подключении сетевого интерфейса управляющего компьютера он должен получить сетевой адрес от сервера.

Если IP-адрес не получен по какой-либо причине, то следует назначить адрес интерфейса вручную, используя любой адрес, кроме 192.168.1.1, в подсети 192.168.1.0/24.

Подключение через консольный порт RS-232

При помощи кабеля RJ-45/DBF9, который входит в комплект поставки устройства, соедините порт **«Console»** пограничного контроллера сессий с портом RS-232 компьютера.

Запустите терминальную программу (например, HyperTerminal или Minicom) и создайте новое подключение. Должен быть использован режим эмуляции терминала VT100.

Выполните следующие настройки интерфейса RS-232:

```
Скорость: 115200 бит/с
Биты данных: 8 бит
Четность: нет
Стоповые биты: 1
Управление потоком: нет
```

5.2.2 Применение изменения конфигурации

Любые изменения, внесенные в конфигурацию, вступят в действие только после применения команды:

```
esbc# commit
Configuration has been successfully committed
```

После применения данной команды запускается таймер "отката" конфигурации. Для остановки таймера и механизма "отката" используется команда:

```
esbc# confirm
Configuration has been successfully confirmed
```

Значение таймера "отката" по умолчанию – 600 секунд. Для изменения данного таймера используется команда:

```
esbc(config)# system config-confirm timeout <TIME>
```

- <TIME> – интервал времени ожидания подтверждения конфигурации, принимает значение в секундах [120..86400].

5.2.3 Базовая настройка устройства

Процедура настройки устройств при первом включении состоит из следующих этапов:

- Изменение пароля пользователя «admin».
- Создание новых пользователей.
- Назначение имени устройства (Hostname).
- Установка параметров подключения к публичной сети в соответствии с требованиями провайдера.
- Настройка удаленного доступа к пограничному контроллеру сессий.
- Применение базовых настроек.

Изменение пароля пользователя «admin»

Для защищенного входа в систему необходимо сменить пароль привилегированного пользователя «admin».

⚠ Учетная запись techsupport необходима для удаленного обслуживания сервисным центром;
Учетная запись remote – аутентификация RADIUS, TACACS+, LDAP;
Удалить пользователей admin, techsupport, remote нельзя. Можно только сменить пароль и уровень привилегий.

⛔ Если информация о пользователе «admin» не отображается в конфигурации, значит параметры данного пользователя настроены по умолчанию (пароль «password», уровень привилегий 15).

Имя пользователя и пароль вводится при входе в систему во время сеансов администрирования устройства.

Для изменения пароля пользователя «admin» используются следующие команды:

```
esbc# configure
esbc(config)# username admin
esbc(config-user)# password <new-password>
esbc(config-user)# exit
```

Создание новых пользователей

Для создания нового пользователя системы или настройки любого из параметров (имени пользователя, пароля, уровня привилегий) используются команды:

```
esbc(config)# username <name>
esbc(config-user)# password <password>
esbc(config-user)# privilege <privilege>
esbc(config-user)# exit
```

⚠ Уровни привилегий 1–9 разрешают доступ к устройству и просмотр его оперативного состояния, но запрещают настройку. Уровни привилегий 10–14 разрешают как доступ, так и настройку большей части функций устройства. Уровень привилегий 15 разрешает как доступ, так и настройку всех функций устройства.

Пример команд для создания пользователя **«fedor»** с паролем **«12345678»** и уровнем привилегий **15** и создания пользователя **«ivan»** с паролем **«password»** и уровнем привилегий **1**:

```
esbc# configure
esbc(config)# username fedor
esbc(config-user)# password 12345678
esbc(config-user)# privilege 15
esbc(config-user)# exit
esbc(config)# username ivan
esbc(config-user)# password password
esbc(config-user)# privilege 1
esbc(config-user)# exit
```

Назначение имени устройства

Для назначения имени устройства используются следующие команды:

```
esbc# configure
esbc(config)# hostname <new-name>
```

После применения конфигурации приглашение командной строки изменится на значение, заданное параметром **<new-name>**.

Настройка параметров публичной сети

Для настройки сетевого интерфейса пограничного контроллера сессий в публичной сети необходимо назначить устройству параметры, определённые провайдером сети – IP-адрес, маска подсети и адрес шлюза по умолчанию.

Пример команд настройки статического IP-адреса для саб-интерфейса **Gigabit Ethernet 1/0/2.150** для доступа к устройству через **VLAN 150**.

Параметры интерфейса:

- IP-адрес – 192.168.16.144;
- Маска подсети – 255.255.255.0;
- IP-адрес шлюза по умолчанию – 192.168.16.1.

```
esbc# configure
esbc(config)# interface gigabitethernet 1/0/2.150
esbc(config-if-sub)# ip address 192.168.16.144/24
esbc(config-if-sub)# exit
esbc(config)# ip route 0.0.0.0/0 192.168.16.1
```

Для того чтобы убедиться, что адрес был назначен интерфейсу, после применения конфигурации введите следующую команду:

```
esbc# show ip interfaces
IP address                  Interface       Admin   Link    Type      Precedence
--------------------------  -------------   -----   -----   -------   -----------
192.168.16.144/24           gi1/0/2.150     Up      Up      static    primary
```

Провайдер может использовать динамически назначаемые адреса в своей сети. Для получения IP-адреса может использоваться протокол DHCP, если в сети присутствует сервер DHCP.

Пример настройки, предназначенной для получения динамического IP-адреса от DHCP-сервера на интерфейсе **Gigabit Ethernet 1/0/10**:

```
esbc# configure
esbc(config)# interface gigabitethernet 1/0/10
esbc(config-if)# ip address dhcp
esbc(config-if)# exit
```

Для того чтобы убедиться, что адрес был назначен интерфейсу, введите следующую команду после применения конфигурации:

```
esbc# show ip interfaces
IP address                    Interface    Admin   Link    Type      Precedence
---------------------------   ----------   -----   -----   -------   -----------
192.168.11.5/25               gi1/0/10     Up      Up      DHCP      --
```

Настройка удаленного доступа к устройству

В заводской конфигурации разрешен удаленный доступ к устройству по протоколам Telnet или SSH из зоны **«trusted»**. Для того чтобы разрешить удаленный доступ к устройству из других зон, например, из публичной сети, необходимо создать соответствующие правила в firewall.

При конфигурировании доступа к устройству правила создаются для пары зон:

- **source-zone** – зона, из которой будет осуществляться удаленный доступ;
- **self** – зона, в которой находится интерфейс управления устройством.

Для создания разрешающего правила используются следующие команды:

```
esbc# configure
esbc(config)# security zone-pair <source-zone> self
esbc(config-zone-pair)# rule <number>
esbc(config-zone-rule)# action permit
esbc(config-zone-rule)# match protocol tcp
esbc(config-zone-rule)# match source-address <network object-group>
esbc(config-zone-rule)# match destination-address <network object-group>
esbc(config-zone-rule)# match destination-port <service object-group>
esbc(config-zone-rule)# enable
esbc(config-zone-rule)# exit
esbc(config-zone-pair)# exit
```

Пример команд для разрешения пользователям из зоны **«untrusted»** с IP-адресами **132.16.0.5-132.16.0.10** подключаться к устройству с IP-адресом **40.13.1.22** по протоколу SSH:

```
esbc# configure
esbc(config)# object-group network clients
esbc(config-addr-set)# ip address-range 132.16.0.5-132.16.0.10
esbc(config-addr-set)# exit
esbc(config)# object-group network gateway
esbc(config-addr-set)# ip address-range 40.13.1.22
esbc(config-addr-set)# exit
esbc(config)# object-group service ssh
esbc(config-port-set)# port-range 22
esbc(config-port-set)# exit
esbc(config)# security zone-pair untrusted self
esbc(config-zone-pair)# rule 10
esbc(config-zone-rule)# action permit
esbc(config-zone-rule)# match protocol tcp
esbc(config-zone-rule)# match source-address clients
esbc(config-zone-rule)# match destination-address gateway
esbc(config-zone-rule)# match destination-port ssh
esbc(config-zone-rule)# enable
esbc(config-zone-rule)# exit
esbc(config-zone-pair)# exit
```

6 Обновление программного обеспечения

- Обновление программного обеспечения средствами системы
- Обновление программного обеспечения из начального загрузчика
- Обновление вторичного загрузчика (U-Boot)

6.1 Обновление программного обеспечения средствами системы

Для обновления программного обеспечения понадобится один из следующих серверов: TFTP, FTP, SCP. На сервер должны быть помещены файлы программного обеспечения устройства, полученные от производителя.
На устройстве хранятся две копии программного обеспечения. Для обеспечения надежности процедуры обновления программного обеспечения доступна для обновления только копия, которая не была использована для последнего старта устройства.

При обновлении программного обеспечения конфигурация пограничного контроллера сессий конвертируется в соответствии с новой версией.
При загрузке пограничного контроллера сессий с более старой версией программного обеспечения, чем загруженная ранее, конфигурация не конвертируется и впоследствии удаляется.

Обновление ПО с более ранних версий можно произвести, воспользовавшись инструкцией, приведенной в разделе Обновление программного обеспечения из начального загрузчика.

Обновление программного обеспечения на устройстве, работающем под управлением операционной системы, выполняется в следующем порядке.

1. Подготовьте для работы выбранный сервер. Должен быть известен адрес сервера, на сервере должен быть размещен дистрибутивный файл программного обеспечения.
2. Пограничный контроллер сессий должен быть подготовлен к работе в соответствии с требованиями документации. Конфигурация устройства должна позволять обмениваться данными по протоколам TFTP/FTP/SCP и ICMP с сервером. При этом должна быть учтена принадлежность сервера к зонам безопасности пограничного контроллера сессий.
3. Подключитесь к устройству локально через консольный порт Console или удаленно, используя протоколы Telnet или SSH.
   Проверьте доступность сервера для пограничного контроллера сессий, используя команду *ping* на устройстве. Если сервер не доступен – проверьте правильность настроек пограничного контроллера сессий и состояние сетевых интерфейсов сервера.
4. Для обновления программного обеспечения устройства введите следующую команду. В качестве параметра <*server*> должен быть указан IP-адрес используемого сервера. Для обновления с FTP или SCP-сервера потребуется ввести имя пользователя (параметр <*user*>) и пароль (параметр <*password*>). В качестве параметра <*file_name*> укажите имя файла программного обеспечения, помещенного на сервер (при использовании SCP нужно указать полный путь – параметр <*folder*>).
   После ввода команды пограничный контроллер сессий скопирует файл во внутреннюю память, проверит целостность данных и сохранит его в энергонезависимую память устройства.
   TFTP:

```
esbc# copy tftp://<server>:/<file_name> system:firmware
```

FTP:

```
esbc# copy ftp://[<user>[:<password>]@]<server>:/<file_name> system:firmware
```

SCP:

```
esbc# copy scp://[<user>[:<password>]@]<server>://<folder>/<file_name>system:firmware
```

SFTP:

```
esbc# copy sftp://[<user>[:<password>]@]<server>:/<file_name> system:firmware
```

Для примера обновите основное ПО через SCP:

```
esbc# copy scp://adm:password123@192.168.16.168://home/tftp/firmware system:firmware
```

5. Для того чтобы устройство работало под управлением новой версии программного обеспечения, необходимо произвести переключение активного образа. С помощью команды *show bootvar* следует выяснить номер образа, содержащего обновленное ПО.

```
esbc# show bootvar
Image   Version                      Date                  Status         After reboot
-----   --------------------------   -------------------   ------------   ------------
1       1.33.0 build                 date 31/03/2025 time  Not Active
        15[ed4770d074]               16:28:01

2       1.33.0 build                 date 31/03/2025 time  Active         *
        16[ed4770d074]               17:41:10
```

Для выбора образа используйте команду:

```
esbc# boot system image-[1|2]|inactive
```

6. Для обновления вторичного загрузчика (U-Boot) введите следующую команду. В качестве параметра *<server>* должен быть указан IP-адрес используемого сервера. Для обновления с FTP или SCP-сервера потребуется ввести имя пользователя (параметр *<user>*) и пароль (параметр *<password>*). В качестве параметра *<file_name>* укажите имя файла вторичного загрузчика, помещенного на сервер (при использовании SCP нужно указать полный путь – параметр *<folder>*). После ввода команды пограничный контроллер сессий скопирует файл во внутреннюю память, проверит целостность данных и сохранит его в энергонезависимую память устройства.
TFTP:

```
esbc# copy tftp://<server>:/<file_name> system:boot-2
```

FTP:

```
esbc# copy ftp://<server>:/<file_name> system:boot-2
```

SCP:

```
esbc# copy scp://[<user>[:<password>]@]<server>://<folder>/<file_name>system:boot-2
```

SFTP:

```
esbc# copy sftp://<server>:/<file_name> system:boot-2
```

6.2 Обновление программного обеспечения из начального загрузчика

Программное обеспечение пограничного контроллера сессий можно обновить из начального загрузчика следующим образом:

1. Остановите загрузку после окончания инициализации пограничного контроллера сессий загрузчиком U-Boot, нажав клавишу ***<Esc>***.

```
Configuring PoE...
distribution 1 dest_threshold 0xa drop_timer 0x0
Configuring POE in bypass mode
NAE configuration done!
initializing port 0, type 2.
initializing port 1, type 2.
SMC Endian Test:b81fb81f
nae-0, nae-1
  =======Skip: Load SYS UCORE for old 8xxB1/3xxB0 revision on default.
Hit any key to stop autoboot:  2
```

2. Укажите IP-адрес TFTP-сервера:

```
BRCM.XLP316Lite Rev B0.u-boot# serverip 10.100.100.1
```

Для версии 1.5 и выше:

```
BRCM.XLP316Lite Rev B0.u-boot# serverip 10.100.100.1
```

3. Укажите IP-адрес пограничного контроллера сессий:

```
BRCM.XLP316Lite Rev B0.u-boot# ipaddr 10.100.100.2
```

Для версии 1.5 и выше:

```
BRCM.XLP316Lite Rev B0.u-boot# ipaddr 10.100.100.2
```

4. Укажите имя файла программного обеспечения на TFTP-сервере:

```
BRCM.XLP316Lite Rev B0.u-boot# firmware_file firmware
```

Для версии 1.5 и выше:

```
BRCM.XLP316Lite Rev B0.u-boot# firmware_file firmware
```

5. Можно сохранить окружение командой *saveenv* для будущих обновлений.

6. Запустите процедуру обновления программного обеспечения:

```
BRCM.XLP316Lite Rev B0.u-boot# run tftp_update_image1
Using nae-0-3 device
TFTP from server 10.100.100.1; our IP address is 10.100.100.2
Filename 'esbc3200/firmware'.
Load address: 0xa800000060000000
Loading: TftpStart:TftpTimeoutMsecs = 10000, TftpTimeoutCountMax = 6
#################################################################
#################################################################
#################################################################
#########################
#################################
done
Bytes transferred = 64453909 (3d77d15 hex)
Device 0: MT29F8G08ABBCAH4 ... is now current device

NAND erase: device 0 offset 0x1440000, size 0x6400000
Bad block table found at page 262080, version 0x01
Bad block table found at page 262016, version 0x01
Erasing at 0x7800000 -- 1895825408% complete..
OK

NAND write: device 0 offset 0x1440000, size 0x6400000
104857600 bytes written: OK
```

7. Установите загруженное программное обеспечение в качестве образа для запуска системы и перезагрузите роутер:

```
BRCM.XLP316Lite Rev B0.u-boot# run set_bootpart_1
```

Для версии 1.5 и выше:

```
BRCM.XLP316Lite Rev B0.u-boot# boot_system image1
BRCM.XLP316Lite Rev B0.u-boot# reset
```

6.3 Обновление вторичного загрузчика (U-Boot)

Вторичный загрузчик занимается инициализацией NAND и пограничного контроллера сессий. При обновлении новый файл вторичного загрузчика сохраняется на flash на месте старого.

Для просмотра текущей версии загрузочного файла, работающего на устройстве, введите команду *version* в CLI U-Boot, также версия отображается в процессе загрузки пограничного контроллера сессий:

```
BRCM.XLP316Lite Rev B0.u-boot# version
BRCM.XLP.U-Boot:1.1.0.47 (29/11/2016 - 19:00:24)
```

Процедура обновления ПО:

1. Остановите загрузку устройства после окончания инициализации пограничного контроллера сессий загрузчиком U-Boot, нажав клавишу ***<Esc>***.

```
Configuring PoE...
distribution 1 dest_threshold 0xa drop_timer 0x0
Configuring POE in bypass mode
NAE configuration done!
initializing port 0, type 2.
initializing port 1, type 2.
SMC Endian Test:b81fb81f
nae-0, nae-1
  =======Skip: Load SYS UCORE for old 8xxB1/3xxB0 revision on default.
Hit any key to stop autoboot:  2
```

2. Укажите IP-адрес TFTP-сервера:

```
BRCM.XLP316Lite Rev B0.u-boot# setenv serverip 10.100.100.1
```

   Для версии 1.5 и выше:

```
BRCM.XLP316Lite Rev B0.u-boot# serverip10.100.100.2
```

3. Укажите IP-адрес пограничного контроллера сессий:

```
BRCM.XLP316Lite Rev B0.u-boot# setenv ipaddr 10.100.100.2
```

   Для версии 1.5 и выше:

```
BRCM.XLP316Lite Rev B0.u-boot# ipaddr 10.100.100.2
```

4. Укажите имя файла загрузчика на TFTP-сервере:

```
BRCM.XLP316Lite Rev B0.u-boot# uboot_file u-boot.bin
```

   Для версии 1.5 и выше:

```
BRCM.XLP316Lite Rev B0.u-boot# uboot_file u-boot.bin
```

5. Можно сохранить окружение командой *saveenv* для будущих обновлений.
6. Запустите процедуру обновления программного обеспечения:

```
BRCM.XLP316Lite Rev B0.u-boot# run upd_uboot
```

Для версии 1.5 и выше:

```
BRCM.XLP316LiteRevB0.u-boot# run tftp_update_uboot
Using nae-1 device
TFTP from server 10.100.100.1; our IP address is 10.100.100.2
Filename 'esbc3200/u-boot.bin'.
Load address: 0xa800000078020000
Loading: #################################################################
done
Bytes transferred = 852648 (d02a8 hex)
SF: Detected MX25L12805D with page size 256, total 16777216 bytes
16384 KiB MX25L12805D at 0:0 is now current device
```

7. Перезагрузите пограничный контроллер сессий:

```
BRCM.XLP316Lite Rev B0.u-boot# reset
```

7 Рекомендации по безопасной настройке

- Общие рекомендации
- Настройка системы логирования событий
  - Рекомендации
  - Предупреждения
  - Пример настройки
- Настройка политики использования паролей
  - Рекомендации
  - Пример настройки
- Настройка политики AAA
  - Рекомендации
  - Предупреждения
  - Пример настройки
- Настройка удалённого управления
  - Рекомендации
  - Пример настройки
- Настройка механизмов защиты от сетевых атак
  - Рекомендации
  - Пример настройки

Рекомендации по безопасной настройке носят общий характер и подходят для большинства инсталляций. Настоящие рекомендации в значительной степени повышают безопасность эксплуатации устройства, но не являются исчерпывающими. В зависимости от схемы применения устройства необходимо настраивать и другие параметры безопасности. В некоторых специфических случаях выполнение данных рекомендаций может привести к неработоспособности сети. При настройке устройства стоит в первую очередь следовать техническим требованиям и регламентам сетей, в которых будет эксплуатироваться данное устройство.

7.1 Общие рекомендации

- Рекомендуется всегда отключать неиспользуемые физические интерфейсы с помощью команды *shutdown*. Команда подробно описана в разделе Конфигурирование и мониторинг интерфейсов справочника команд CLI.
- Рекомендуется всегда настраивать синхронизацию системных часов с доверенными источниками сетевого времени (NTP). Алгоритм настройки NTP приведён в разделе Настройка NTP настоящего руководства. Подробная информация о командах для настройки NTP приведена в разделе Управление системными часами справочника команд CLI.
- Рекомендуется отключать NTP broadcast client, включённый по умолчанию в заводской конфигурации.
- Не рекомендуется использовать команду *ip firewall disable*, отключающую межсетевое экранирование. Следует всегда назначать интерфейсам соответствующие зоны безопасности и настраивать корректные правила межсетевого экрана. Алгоритм настройки межсетевого экрана приведён в разделе Конфигурирование Firewall настоящего руководства. Подробная информация о командах для настройки межсетевого экрана приведена в разделе Управление Firewall справочника команд CLI.

⚠ Для предачи сигнального (SIP) и медиа (RTP) трафика не требуется конфигурирование дополнительных правил и зон Firewal. Правила будут созданы автоматически при конфигурировании SIP-транспортов, SIP-транков и абонентских интерфейсов. Подробная информация о настройке находится в разделе Управление ESBC.

7.2 Настройка системы логирования событий

Алгоритмы настройки системы логирования событий приведены в подразделе «Настройка Syslog» раздела Мониторинг настоящего руководства.

Подробная информация о командах для настройки системы логирования событий приведена в разделе Управление SYSLOG справочника команд CLI.

7.2.1 Рекомендации

- Рекомендуется настроить хранение сообщений о событиях в файл syslog на устройстве и передачу этих событий на внешний syslog-сервер.
- Рекомендуется ограничивать размер syslog-файла на устройстве.
- Рекомендуется настраивать ротацию syslog-файлов на устройстве.
- Рекомендуется включать нумерацию сообщений syslog.

7.2.2 Предупреждения

- Данные, хранящиеся в файловой системе **tmpsys:syslog**, не сохраняются при перезагрузке устройства. Этот тип файловой системы рекомендуется использовать для хранения оперативных логов.
- Не рекомендуется использовать файловую систему **flash:syslog** для хранения логов, так как это может привести к преждевременному выходу из строя устройства esbc.

7.2.3 Пример настройки

Задача:

Настроить хранение сообщений о событиях уровня info и выше в файл syslog на устройстве и настроить передачу этих событий на внешний syslog-сервер. Ограничить файл размером 512 Кбайт. Включить ротацию 3 файлов. Включить нумерацию сообщений syslog.

Решение:

Настраиваем хранение syslog-сообщений в файле:

```
esbc(config)# syslog file tmpsys:syslog/default
esbc((config-syslog-file)# severity info
esbc((config-syslog-file)# exit
```

Настраиваем ограничение размера и ротацию файлов:

```
esbc(config)# syslog max-files 3
esbc(config)# syslog file-size 512
```

Настраиваем передачу сообщений на внешний сервер:

```
esbc(config)# syslog host mylog
esbc(config-syslog-host)# remote-address 92.168.1.2
esbc(config-syslog-host)# transport udp
esbc(config-syslog-host)# port 514
esbc(config-syslog-host)# severity info
esbc(config-syslog-host)# exit
```

Включаем нумерацию сообщений syslog:

```
esbc(config)# syslog sequence-numbers
```

7.3 Настройка политики использования паролей

Алгоритмы настройки политики AAA приведены в разделе Настройка AAA настоящего руководства.

Подробная информация о командах для настройки политики использования паролей приведена в разделе Настройка AAA справочника команд CLI.

7.3.1 Рекомендации

- Рекомендуется всегда включать требования на смену пароля по умолчанию пользователя admin.
- Рекомендуется ограничивать время жизни паролей и запрещать повторно использовать, как минимум, предыдущий пароль.
- Рекомендуется выставлять требования минимальной длины пароля больше 8 символов.
- Рекомендуется выставлять требования на использование строчных и прописных букв, цифр и спецсимволов.

7.3.2 Пример настройки

Задача:

- Настроить парольную политику с обязательным требованием смены пароля по умолчанию, временем действия пароля 1 месяц и запретом на использование 12 последних паролей.
- Задать минимальную длину пароля 16 символов, максимальную — 24 символа.
- Пароль должен содержать не менее 3 прописных букв, не менее 5 строчных букв, не менее 4 цифр и не менее 2 спецсимволов. Пароль в обязательном порядке должен содержать все 4 типа символов.

Решение:

Включаем запрос на смену пароля по умолчанию для пользователя admin:

```
esbc(config)# security passwords default-expired
```

Устанавливаем время жизни пароля 30 дней и запрет на использование предыдущих 12 паролей:

```
esbc(config)# security passwords lifetime 30
esbc(config)# security passwords history 12
```

Устанавливаем ограничения на длину пароля:

```
esbc(config)# security passwords min-length 16
esbc(config)# security passwords max-length 24
```

Устанавливаем ограничения по минимальному количеству символов соответствующих типов:

```
esbc(config)# security passwords upper-case 3
esbc(config)# security passwords lower-case 5
esbc(config)# security passwords special-case 2
esbc(config)# security passwords numeric-count 4
esbc(config)# security passwords symbol-types 4
```

7.4 Настройка политики AAA

Алгоритмы настройки политики AAA приведены в разделе Настройка AAA настоящего руководства.

Подробная информация о командах для настройки политики AAA приведена в разделе Настройка AAA справочника команд CLI.

7.4.1 Рекомендации

- Рекомендуется использовать ролевую модель доступа на устройство.
- Рекомендуется использовать персональные учетные записи для аутентификации на устройстве.
- Рекомендуется включать логирование вводимых пользователем команд.
- Рекомендуется использовать несколько методов аутентификации для входа на устройства через консоль, удалённого входа на устройства и повышения привилегий. Оптимальной считается комбинация из аутентификации по одному из протоколов RADIUS/TACACS/LDAP и локальной аутентификации.
- Рекомендуется понизить уровень привилегий встроенной учётной записи **admin** до 1.
- Рекомендуется настроить логирование изменений локальных учётных записей.
- Рекомендуется настроить логирование изменений политики AAA.

7.4.2 Предупреждения

- Встроенную учётную запись admin удалить нельзя.
- Команда **no username admin** не удаляет пользователя **admin**, сбрасывает его конфигурацию в значения по умолчанию. После применения этой команды пользователь **admin** не будет отображаться в конфигурации.
- Команда **no password** для пользователя **admin** также не удаляет пароль пользователя **admin**, а сбрасывает его в значение по умолчанию. После применения этой команды пароль пользователя **admin** перестаёт отображаться в конфигурации и становится ‘password'.
- Перед установкой пользователю admin пониженных привилегий у вас должен быть настроен пользователь с уровнем привилегий 15 или задан ENABLE-пароль.

7.4.3 Пример настройки

Задача:

Настроить политику AAA:

- Для удалённого входа по протоколу SSH использовать аутентификации через RADIUS.
- Для входа через локальную консоль использовать аутентификации через RADIUS, в случае отсутствия связи с RADIUS-серверами использовать локальную аутентификацию.
- Использовать ENABLE-пароль, заданный через RADIUS, в случае отсутствия связи с RADIUS-серверами использовать локальный ENABLE-пароль.
- Установить пользователю admin пониженный уровень привилегий.
- Настроить логирование изменений локальных учётных записей.
- Настроить логирование изменений политик AAA.
- Настроить логирование вводимых команд.

Решение:

Создаем локального пользователя **local-operator** с уровнем привилегий 8:

```
esbc(config)# username local-operator
esbc(config-user)# password Pa$$w0rd1
esbc(config-user)# privilege 8
esbc(config-user)# exit
```

Задаём локальный ENABLE-пароль:

```
esbc(config)# enable password $6e5c4r3e2t!
```

Понижаем привилегии пользователя admin:

```
esbc(config)# username admin
esbc(config-user)# privilege 1
esbc(config-user)# exit
```

Настраиваем связь с двумя RADIUS-серверами, основным 192.168.1.11 и резервным 192.168.2.12:

```
esbc(config)# radius-server host 192.168.1.11
esbc(config-radius-server)# key ascii-text encrypted 8CB5107EA7005AFF
esbc(config-radius-server)# priority 100
esbc(config-radius-server)# exit
esbc(config)# radius-server host 192.168.2.12
esbc(config-radius-server)# key ascii-text encrypted 8CB5107EA7005AFF
esbc(config-radius-server)# priority 150
esbc(config-radius-server)# exit
```

Настраиваем политику AAA:

```
esbc(config)# aaa authentication login CONSOLE radius local
esbc(config)# aaa authentication login SSH radius
esbc(config)# aaa authentication enable default radius enable
esbc(config)# aaa authentication mode break
esbc(config)# line console
esbc(config-line-console)# login authentication CONSOLE
esbc(config-line-console)# exit
esbc(config)# line ssh
esbc(config-line-ssh)# login authentication SSH
esbc(config-line-ssh)# exit
```

Настраиваем логирование:

```
esbc(config)# logging userinfo
esbc(config)# logging aaa
esbc(config)# syslog cli-commands
```

7.5 Настройка удалённого управления

Подробная информация о командах настройки удалённого доступа приведена в разделе Настройка доступа SSH, Telnet справочника команд CLI.

7.5.1 Рекомендации

- Не рекомендуется включать удалённое управление по протоколу Telnet.
- Рекомендуется использовать криптостойкие алгоритмы аутентификации sha2-512 и отключить все остальные.
- Рекомендуется использовать криптостойкие алгоритмы шифрования aes256ctr и отключить все остальные.
- Рекомендуется использовать криптостойкий алгоритм обмена ключами шифрования dh-group-exchange-sha256 и отключить все остальные.
- Рекомендуется использовать криптостойкий алгоритм верификации Host-Key для SSH rsa и отключить все остальные.
- Рекомендуется разрешить доступ к удалённому управлению устройством только с определённых IP-адресов.
- Перед началом эксплуатации рекомендуется перегенерировать ключи шифрования.

7.5.2 Пример настройки

Задача:

Сгенерировать новые ключи шифрования. Использовать криптостойкие алгоритмы.

Решение:

Отключаем устаревшие и не криптостойкие алгоритмы:

```
esbc(config)# ip ssh server
esbc(config)# ip ssh authentication algorithm md5 disable
esbc(config)# ip ssh authentication algorithm md5-96 disable
esbc(config)# ip ssh authentication algorithm ripemd160 disable
esbc(config)# ip ssh authentication algorithm sha1 disable
esbc(config)# ip ssh authentication algorithm sha1-96 disable
esbc(config)# ip ssh authentication algorithm sha2-256 disable
esbc(config)# ip ssh encryption algorithm 3des disable
esbc(config)# ip ssh encryption algorithm aes128 disable
esbc(config)# ip ssh encryption algorithm aes128ctr disable
esbc(config)# ip ssh encryption algorithm aes192 disable
esbc(config)# ip ssh encryption algorithm aes192ctr disable
esbc(config)# ip ssh encryption algorithm aes256 disable
esbc(config)# ip ssh encryption algorithm arcfour disable
esbc(config)# ip ssh encryption algorithm arcfour128 disable
esbc(config)# ip ssh encryption algorithm arcfour256 disable
esbc(config)# ip ssh encryption algorithm blowfish disable
esbc(config)# ip ssh encryption algorithm cast128 disable
esbc(config)# ip ssh key-exchange algorithm dh-group-exchange-sha1 disable
esbc(config)# ip ssh key-exchange algorithm dh-group1-sha1 disable
esbc(config)# ip ssh key-exchange algorithm dh-group14-sha1 disable
esbc(config)# ip ssh key-exchange algorithm ecdh-sha2-nistp256 disable
esbc(config)# ip ssh key-exchange algorithm ecdh-sha2-nistp384 disable
esbc(config)# ip ssh key-exchange algorithm ecdh-sha2-nistp521 disable
esbc(config)# ip ssh host-key algorithm dsa disable
esbc(config)# ip ssh host-key algorithm ecdsa256 disable
esbc(config)# ip ssh host-key algorithm ecdsa384 disable
esbc(config)# ip ssh host-key algorithm ecdsa521 disable
esbc(config)# ip ssh host-key algorithm ed25519 disable
```

Генерируем новые ключи шифрования:

```
esbc# update ssh-host-key rsa 2048
```

7.6 Настройка механизмов защиты от сетевых атак

Алгоритмы настройки механизмов защиты от сетевых атак приведены в разделе Настройка логирования и защиты от сетевых атак.

Подробная информация о командах для настройки политики использования паролей приведена в разделе Управление логированием и защитой от сетевых атак справочника команд CLI.

7.6.1 Рекомендации

- Рекомендуется всегда включать защиту от ip spoofing.
- Рекомендуется всегда включать защиту от TCP-пакетов с неправильно выставленными флагами.
- Рекомендуется всегда включать защиту от фрагментированных TCP-пакетов с выставленным флагом SYN.
- Рекомендуется всегда включать защиту от фрагментированных ICMP-пакетов.
- Рекомендуется всегда включать защиту ICMP-пакетов большого размера.

- Рекомендуется всегда включать защиту от незарегистрированных IP-протоколов.
- Рекомендуется включать логирование механизма защиты от сетевых атак.

7.6.2 Пример настройки

Задача:

Настроить механизм защиты от сетевых атак в соответствии с рекомендациями.

Решение:

Включаем защиту от ip spoofing и логирование механизма защиты:

```
esbc(config)# ip firewall screen spy-blocking spoofing
esbc(config)# logging firewall screen spy-blocking spoofing
```

Включаем защиту от TCP-пакетов с неправильно выставленными флагами и логирование механизма защиты:

```
esbc(config)# ip firewall screen spy-blocking syn-fin
esbc(config)# logging firewall screen spy-blocking syn-fin
esbc(config)# ip firewall screen spy-blocking fin-no-ack
esbc(config)# logging firewall screen spy-blocking fin-no-ack
esbc(config)# ip firewall screen spy-blocking tcp-no-flag
esbc(config)# logging firewall screen spy-blocking tcp-no-flag
esbc(config)# ip firewall screen spy-blocking tcp-all-flags
esbc(config)# logging firewall screen spy-blocking tcp-all-flags
```

Включаем защиту от фрагментированных ICMP-пакетов и логирование механизма защиты:

```
esbc(config)# ip firewall screen suspicious-packets icmp-fragment
esbc(config)# logging firewall screen suspicious-packets icmp-fragment
```

Включаем защиту от ICMP-пакетов большого размера и логирование механизма защиты:

```
esbc(config)# ip firewall screen suspicious-packets large-icmp
esbc(config)# logging firewall screen suspicious-packets large-icmp
```

Включаем защиту от незарегистрированных IP-протоколов и логирование механизма защиты:

```
esbc(config)# ip firewall screen suspicious-packets unknown-protocols
esbc(config)# logging firewall screen suspicious-packets unknown-protocols
```

8 Примеры подключения ESBC к сети передачи данных

В данном разделе приведены примеры физического подключения ESBC к сети передачи данных. В примерах указан ESBC-3200, но схемы применимы и к vESBC.

После подключения и настройки сетевых интерфейсов (терминации IP-адресов), можно использовать эти интерфейсы для организации SIP-trunk/User-interface между сетями NET 1 и NET 2, в качестве которых, например, могут выступать публичная сеть Internet и локальная сеть предприятия.

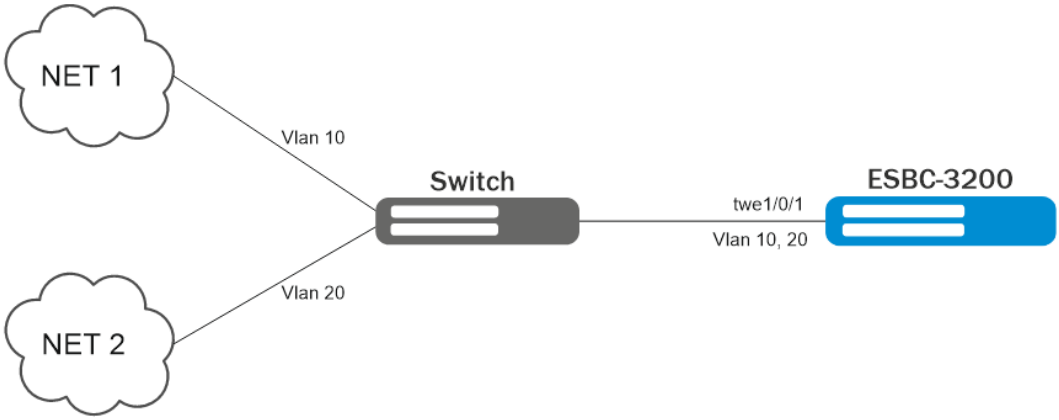
Команды и примеры настройки интерфейсов ESBC приведены в разделах Управление интерфейсами и Управление функциями второго уровня (L2) Руководства по эксплуатации, а также в разделах Управление L2-функциями и Конфигурирование и мониторинг интерфейсов Справочника команд CLI.

- Подключение к разным сетям с использованием двух сетевых интерфейсов
- Подключение к сети с использованием одного сетевого интерфейса
- Подключение к сети с использованием нескольких сетевых интерфейсов (резервирование линков)
  - Использование протокола LACP для агрегирования сетевых интерфейсов в один логический канал
  - Использование моста (Bridge) для терминации на уровне L3
- Подключение к нескольким коммутаторам с использованием нескольких сетевых интерфейсов (резервирование линков и узлов сети)
  - Использование протокола LACP для агрегирования сетевых интерфейсов в один логический канал
  - Использование моста (Bridge) для терминации на уровне L3
- Использование кластера (только для ESBC-3200)

8.1 Подключение к разным сетям с использованием двух сетевых интерфейсов

При подключении к сети с использованием двух сетевых интерфейсов в разных сетях, следует перевести режим работы интерфейсов twe1/0/1 и twe1/0/2 в L3 (*mode routerport*) и назначить на них соответствующие IP-адреса. При использовании VLAN требуется сконфигурировать соответствующие саб-интерфейсы, например twe1/0/1.10 и twe1/0/1.20 и назначить на них соответствующие IP-адреса.

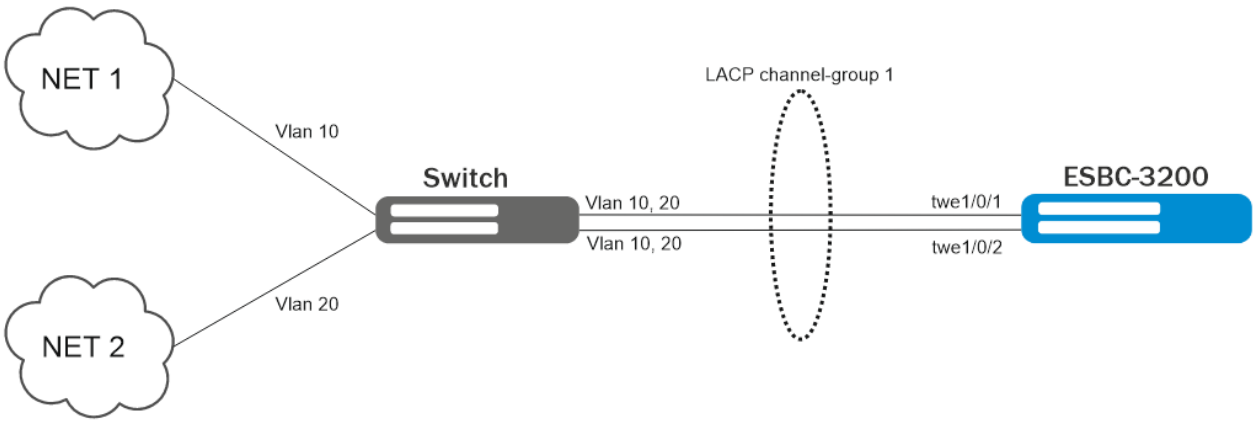
8.2 Подключение к сети с использованием одного сетевого интерфейса



При подключении к сети с использованием одного сетевого интерфейса ESBC-3200, следует перевести режим работы интерфейса twe1/0/1 в L3 (*mode routerport*). Для терминации VLAN 10 и VLAN 20 требуется сконфигурировать два саб-интерфейса twe1/0/1.10 и twe1/0/1.20 и назначить на них соответствующие IP-адреса.

На порту коммутатора (Switch) VLAN 10 и VLAN 20 необходимо передавать с тегами (mode trunk).

8.3 Подключение к сети с использованием нескольких сетевых интерфейсов (резервирование линков)



8.3.1 Использование протокола LACP для агрегирования сетевых интерфейсов в один логический канал

Для агрегации интерфейсов twe1/0/1 и twe1/0/2 необходимо включить их в одну группу (channel-group 1). Для терминации на уровне L3 следует использовать *interface port-channel 1*.

Терминация VLAN 10 и VLAN 20 осуществляется путем конфигурации двух саб-интерфейсов: *port-channel 1.10* и *port-channel 1.20*.

На коммутаторе также необходимо настроить протокол LACP.

8.3.2 Использование моста (Bridge) для терминации на уровне L3

Для терминации на L3 используется интерфейс bridge. Пример настройки:

```
esbc# configure
esbc(config)# bridge 10
esbc(config-bridge)# vlan 10
esbc(config-bridge)# ip address 192.168.1.1/24
esbc(config-bridge)# exit
esbc(config)# bridge 20
esbc(config-bridge)# vlan 20
esbc(config-bridge)# ip address 192.168.2.1/24
esbc(config-bridge)# exit
```

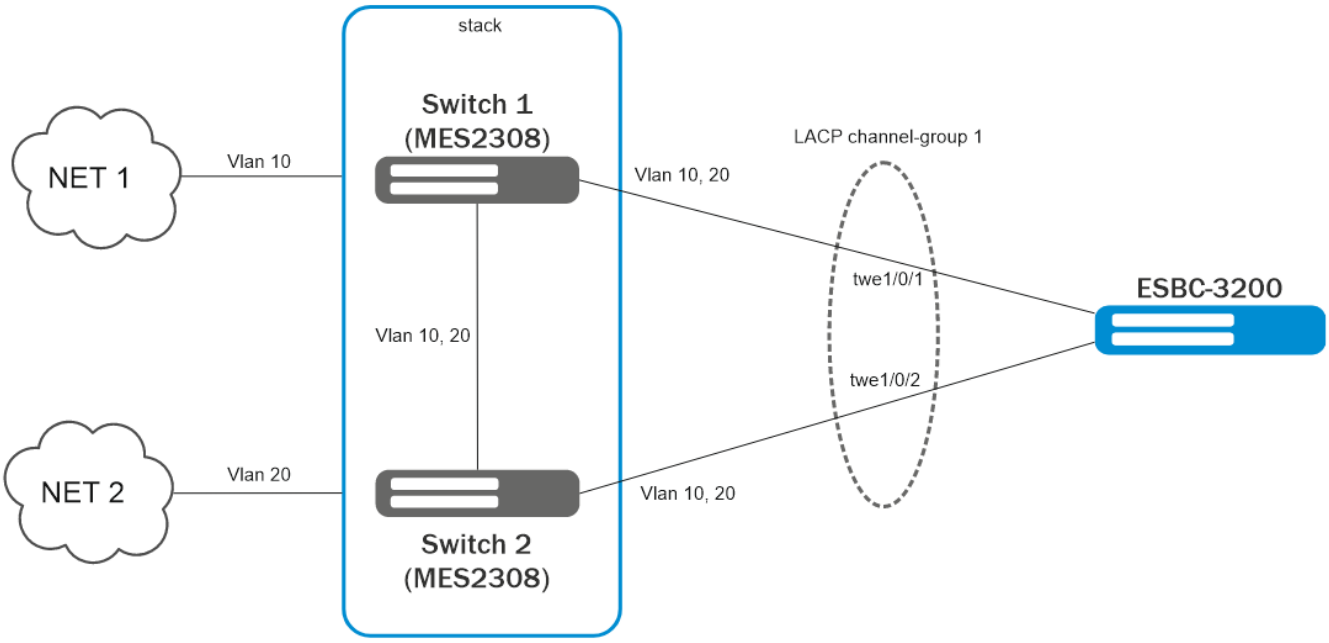
На интерфейсах twe1/0/1 и twe1/0/2 следует использовать режим *switchport* и добавить VLAN 10 и 20:

```
esbc# configure
esbc(config)# interface twentyfivegigabitethernet 1/0/1-2
esbc(config-if-twe)#  mode switchport
esbc(config-if-twe)#  switchport mode trunk
esbc(config-if-twe)#  switchport trunk allowed vlan add 10,20
esbc(config-if-twe)#  exit
```

⚠ При использовании интерфейсов в режиме **switchport** необходимо дополнительно настроить протокол семейства STP для предотвращения образования петель.

❌ Наиболее предпочтительным является подключение, описанное в примере Использование протокола LACP для агрегирования сетевых интерфейсов в один логический канал, т. к. использование моста (Bridge) может увеличить нагрузку на устройство и приводить к образованию петель коммутации.

8.4 Подключение к нескольким коммутаторам с использованием нескольких сетевых интерфейсов (резервирование линков и узлов сети)



8.4.1 Использование протокола LACP для агрегирования сетевых интерфейсов в один логический канал

Подключение и настройка осуществляется аналогично примеру Использование протокола LACP для агрегирования сетевых интерфейсов в один логический канал.

Обязательным требованием для реализации данного подключения является поддержка коммутаторами (Switch 1 и Switch 2) функции стекирования и/или VPC.

8.4.2 Использование моста (Bridge) для терминации на уровне L3

В случае если коммутаторы не поддерживают функции стекирования и VPC, то допускается подключение, описанное в примере Использование моста (Bridge) для терминации на уровне L3, но требуется дополнительная настройка протокола STP таким образом, чтобы в топологии STP был заблокирован один из интерфейсов ESBC с целью исключения прохождения транзитного broadcast-трафика.

В данном примере при обрыве линка между Switch 1 и Switch 2, транзитный трафик в любом случае будет проходить через ESBC, что может повлиять на производительность.

Наиболее предпочтительным является подключение, описанное в примере Использование протокола LACP для агрегирования сетевых интерфейсов в один логический канал, т. к. использование моста (Bridge) может увеличить нагрузку на устройство и приводить к образованию петель коммутации.

8.5 Использование кластера (только для ESBC-3200)

Пример конфигурации кластера приведен в разделе Управление кластеризацией данного руководства по эксплуатации.

9 Управление ESBC

- Общие сведения
- Настройка абонентских интерфейсов
- Настройка SIP-транков
- Настройка транковых групп
  - Логика работы транковой группы для распределения вызовов на транки, входящие в ее состав
- Настройка SIP-транспортов
- Настройка медиаресурсов
- Настройка таблиц маршрутизации
- Настройка модификаторов
  - Общие модификаторы
  - Модификаторы SIP
    - Модификатор добавления заголовка (add)
    - Модификатор передачи заголовка (transit)
    - Модификатор удаления заголовка (no-transit)
    - Модификатор транзита и замены заголовка (replace)
    - Модификатор копирования (copy)
    - Использование системных переменных
- Настройка SIP-профилей
  - Контроль доступности направления
  - Список причин отбоя для перехода на следующее направление
  - Поведение при перенаправлении
  - Игнорирование OPTIONS
- Настройка медиапрофилей
  - Управление типом медиаданных и кодеками
    - Примеры использования медиапрофиля для управления кодеками и типами медиаданных в режиме проксирования
  - Транскодирование
    - Примеры использования медиапрофилей для управления кодеками в режиме транскодирования
  - Таймаут ожидания RTP-пакетов
  - SRTP
  - Контроль источника RTP
- Настройка профилей безопасности
  - Общий принцип работы модуля fail2ban
  - Фильтрация SIP-флуда
    - Фильтрация клиентских приложений
  - Блокировка по AOR/User-Agent
  - Объединение ошибок по IP-адресу
- Настройка криптопрофилей
- Настройка NAT
- Настройка Public IP
- Настройка QoS
- Изменение количества модулей
- Ограничение входящего трафика
- Мониторинг
- Аварии
- Настройка CDR
- Работа с логами
- Примеры настройки ESBC
  - Настройка для SIP-абонентов
  - Настройка для SIP-транков
  - Настройка для SIP-абонентов, использующих WebRTC

- Настройка динамического режима для SIP-транков

9.1 Общие сведения

В данном разделе содержится описание функций пограничного контроллера сессий ESBC и примеры их настройки для обеспечения передачи SIP-сигнализации и медиапотоков RTP между разными направлениями.

Переход в режим конфигурирования функционала ESBC осуществляется следующими командами CLI:

```
vesbc# configure
vesbc(config)# esbc
vesbc(config-esbc)#
```

Максимальное количество объектов конфигурации ESBC каждого типа:

Объект	Количество
sip transport	500
trunk	500
user-interface	500
trunk-group	250
sip profile	1000
route table	500
rule	64 на таблицу route table
condition	64 на правило rule
media profile	1000
media resource	1000
mod-table	500
mod	64 на таблицу mod-table
cause-list	64
crypto profile	64
flood filter	250
security-profile	500

Не рекомендуется использовать максимальное количество объектов конфигурации одновременно, это может повлиять на работоспособность системы.

Расчет максимального количества контактов зарегистрированных абонентов для vESBC:

Объем оперативной памяти (RAM) vESBC, GB	Количество контактов
3	4500
≥4	20000*(объем RAM - 3)

ESBC-3200 поддерживает до 420000 контактов зарегистрированных абонентов.

9.2 Настройка абонентских интерфейсов

Абонентский интерфейс представляет собой направление для приёма и маршрутизации запросов SIP-абонентов. В конфигурации не задаётся адрес и порт удалённой стороны, для аутентификации используется механизм SIP-регистрации. Регистрация на вышестоящем сервере осуществляется через связанный SIP-транк.

Для создания абонентского интерфейса необходимо настроить:

- SIP-транспорт;
- Медиаресурсы;
- Таблицу маршрутизации.

Эти настройки являются обязательными. Описание конфигурирования и базовой схемы применения представлено в разделе Примеры настройки ESBC.

Помимо этого абонентский интерфейс содержит набор следующих настроек:

- Таблица модификаций (для входящих и исходящих сообщений);
- SIP-профиль;
- Медиапрофиль;
- Профиль безопасности;
- Режим работы абонентов за NAT;
- Public IP;
- QoS;
- Ограничение входящего трафика;
- SIP-домен. При настройке домен будет использоваться в host-part URI в заголовках To и From для исходящих сообщений. Во входящих сообщениях будет осуществляться проверка домена в заголовке From.
- Опция "Разрешить вызовы без регистрации". Разрешает принимать входящие сообщения INVITE от незарегистрированных абонентов.

По умолчанию вызовы с абонентского интерфейса без предварительной регистрации запрещены.

Подробное описание параметров всех настроек можно найти в разделе Настройки абонентского интерфейса Справочника команд CLI.

9.3 Настройка SIP-транков

SIP-транк представляет собой интерфейс для подключения к вышестоящему SIP-устройству (IP АТС/ SIP-proxy/Удаленный SSW и др.) или группе вышестоящих устройств при включении динамического режима работы транка. При включении динамического режима работы в конфигурации необходимо задать адрес и порт удалённой стороны или диапазон адресов и портов. Эти параметры используются для идентификации источника запроса.

⚠ На транке запрещена обработка входящих запросов REGISTER.

Для создания SIP-транка необходимо настроить:

- Адрес удалённой стороны (или диапазон адресов для динамического режима);
- Порт удалённой стороны (или диапазон адресов для динамического режима);
- SIP-транспорт;
- Медиаресурсы.

Эти настройки являются обязательными. Описание конфигурирования и базовой схемы применения представлено в разделе Примеры настройки ESBC.

Помимо этого SIP-транк содержит набор следующих настроек:

- Таблица маршрутизации;
- Таблица модификаций (для входящих и исходящих сообщений);
- SIP-профиль;
- Медиапрофиль;
- Профиль безопасности;
- Режим работы за NAT;
- Public IP;
- QoS;
- Ограничение входящего трафика;
- Динамический режим. Используется для подключения к группе вышестоящих SIP-устройств (IP АТС/ SIP-proxy/Удаленный SSW и др.).
- Опция "Доверенная сеть" для переадресаций;
- SIP-домен. При настройке домен будет использоваться в host-part URI в заголовках To и From для исходящих сообщений. Во входящих сообщениях будет осуществляться проверка домена в заголовке From.

Подробное описание параметров всех настроек можно найти в разделе Настройки SIP-транка Справочника команд CLI.

⚠ Создание транков с одинаковым SIP-транспортом и IP:Port разрешено только в случае, если отличается SIP-домен.

9.4 Настройка транковых групп

Транк-группа представляет собой набор транков различного типа (в текущей версии поддерживается только SIP-транк) и алгоритм балансировки нагрузки между ними. В текущей версии балансировка вызовов осуществляется алгоритмом **round-robin**.

Помимо этого группа содержит набор следующих настроек:

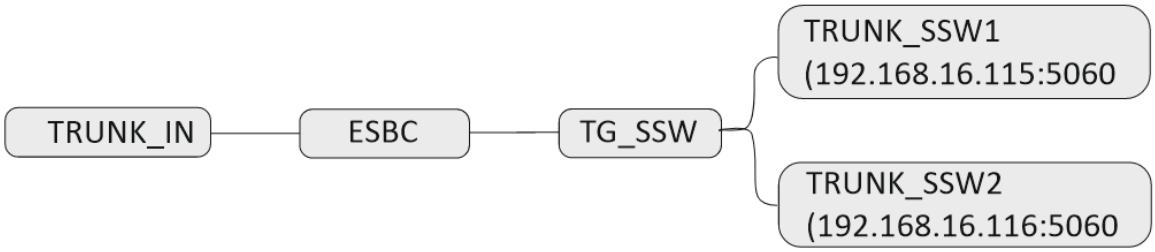
- Таблица маршрутизации;
- Медиапрофиль;
- Медиаресурсы;
- SIP-профиль;
- Профиль безопасности;
- Таблица модификаций (для входящих и исходящих сообщений);
- QoS;
- Public IP;
- Ограничение входящего трафика;
- Опция "Доверенная сеть" для переадресаций.

Логика работы:

Все перечисленные в предыдущем пункте настройки являются общими для всех транков, включенных в состав транковой группы. Это значит, что при отсутствии у транка, входящего в состав транковой группы, какой-либо из перечисленных настроек, будет использоваться настройка из транковой группы. Такой подход позволяет создавать множество транков с минимальным набором настроек, и, объединяя их в транковую группу, производить донастройку через нее. При необходимости можно изменить какие-либо параметры отдельно взятых транков из группы через индивидуальную настройку транков.

Пример работы общих настроек:

Схема:



На ESBC настроена транковая группа TG_SSW, в состав которой входят 2 транка TRUNK_SSW1 и TRUNK_SSW2, также настроен еще один транк TRUNK_IN, который не входит в состав транковой группы. Требуется настроить схему таким образом, чтобы вызовы, которые пришли с TRUNK_IN, маршрутизировались на TG_SSW, и наоборот, вызовы, которые пришли с TRUNK_SSW1 и TRUNK_SSW2, маршрутизировались на TRUNK_IN.

Решение:

1. Создать SIP-транспорт в сторону TRUNK_SSW1 и TRUNK_SSW2:

```
vesbc#
vesbc# configure
vesbc(config)# esbc
vesbc(config-esbc)# sip transport TRANSPORT_SSW
vesbc(config-esbc-sip-transport)# ip address 192.168.16.113
vesbc(config-esbc-sip-transport)# port 5065
```

2. Создать SIP-транспорт в сторону TRUNK_IN:

```
vesbc#
vesbc# configure
vesbc(config)# esbc
vesbc(config-esbc)# sip transport TRANSPORT_TRUNK_IN
vesbc(config-esbc-sip-transport)# ip address 192.168.20.120
vesbc(config-esbc-sip-transport)# port 5067
```

3. Создать медиаресурсы для согласования и передачи голоса на плече TRUNK_IN --- ESBC:

```
vesbc#
vesbc# configure
vesbc(config)# esbc
vesbc(config-esbc)# media resource MEDIA_TRUNK_IN
vesbc(config-esbc-media-resource)# ip address 192.168.20.120
```

4. Создать медиаресурсы для согласования и передачи голоса на плече ESBC --- TG_SSW:

```
vesbc#
vesbc# configure
vesbc(config)# esbc
vesbc(config-esbc)# media resource MEDIA_TG_SSW
vesbc(config-esbc-media-resource)# ip address 192.168.16.113
```

5. Создать SIP-транк в сторону TRUNK_IN:

```
vesbc#
vesbc# configure
vesbc(config)# esbc
vesbc(config-esbc)# trunk sip TRUNK_IN
vesbc(config-esbc-trunk-sip)# sip transport TRANSPORT_TRUNK_IN
vesbc(config-esbc-trunk-sip)# remote address 192.168.20.99
vesbc(config-esbc-trunk-sip)# remote port 5060
vesbc(config-esbc-trunk-sip)# media resource 0 MEDIA_TRUNK_IN
```

6. Создать SIP-транк в сторону TRUNK_SSW1 и TRUNK_SSW2:

```
vesbc#
vesbc# configure
vesbc(config)# esbc
vesbc(config-esbc)# trunk sip TRUNK_SSW1
vesbc(config-esbc-trunk-sip)# sip transport TRANSPORT_SSW
vesbc(config-esbc-trunk-sip)# remote address 192.168.16.115
vesbc(config-esbc-trunk-sip)# remote port 5060
vesbc(config-esbc-trunk-sip)# exit
vesbc(config-esbc)# trunk sip TRUNK_SSW2
vesbc(config-esbc-trunk-sip)# sip transport TRANSPORT_SSW
vesbc(config-esbc-trunk-sip)# remote address 192.168.16.116
vesbc(config-esbc-trunk-sip)# remote port 5060
```

7. Создать транковую группу TG_SSW и добавить туда транки TRUNK_SSW1 и TRUNK_SSW2:

```
vesbc#
vesbc# configure
vesbc(config)# esbc

#Создание и переход в настройки транковой группы TG_SSW:
vesbc(config-esbc)# trunk-group TG_SSW

#Добавление в состав транковой группы транков TRUNK_SSW1 и TRUNK_SSW2:
vesbc(config-esbc-trunk-group)# trunk 0 TRUNK_SSW1
vesbc(config-esbc-trunk-group)# trunk 1 TRUNK_SSW2

#Добавление медиаресурсов:
vesbc(config-esbc-trunk-group)# media resource 0 MEDIA_TG_SSW
```

8. Создать таблицу маршрутизации и добавить туда правила, по которым вызовы, приходящие с транка TRUNK_IN, будут маршрутизироваться в транковую группу TG_SSW:

```
vesbc#
vesbc# configure
vesbc(config)# esbc
vesbc(config-esbc)# route-table TO_TG_SSW
vesbc(config-esbc-route-table)# rule 0
vesbc(config-esbc-route-table-rule)# action direct-to-trunk-group TG_SSW
```

9. Создать таблицу маршрутизации и добавить туда правила, по которым вызовы, приходящие с TG_SSW, будут маршрутизироваться в транк TRUNK_IN:

```
vesbc#
vesbc# configure
vesbc(config)# esbc
vesbc(config-esbc)# route-table TO_TRUNK_IN
vesbc(config-esbc-route-table)# rule 0
vesbc(config-esbc-route-table-rule)# action direct-to-trunk TRUNK_IN
```

10. Привязать созданные таблицы маршрутизации к транку TRUNK_IN и транковой группе TG_SSW:

```
vesbc#
vesbc# configure
vesbc(config)# esbc
vesbc(config-esbc)# trunk sip TRUNK_IN
vesbc(config-esbc-trunk-sip)# route-table TO_TG_SSW
vesbc(config-esbc-trunk-sip)# exit
vesbc(config-esbc)# trunk-group TG_SSW
vesbc(config-esbc-trunk-sip)# route-table TO_TRUNK_IN
```

11. Применить конфигурацию и подтвердить изменения:

```
vesbc# commit
vesbc# confirm
```

На шаге 6 при создании транков, в конфигурацию транков не были добавлены медиаресурсы и таблица маршрутизации. Но эти настройки есть в транковой группе TG_SSW, куда включены оба транка. Поэтому при поступлении вызовов с этих транков они будут маршрутизироваться по таблице маршрутизации, которая привязана к TG_SSW, медиаресурсы для согласования и передачи RTP также будут браться из транковой группы TG_SSW.

В случае если необходимо, чтобы один из транков, входящих в состав транковой группы, при поступлении на него входящих вызовов маршрутизировался по другой таблице маршрутизации или использовал другие медиаресурсы, нужно добавить соответствующие настройки в данный транк. Настройки транковой группы при этом не меняются, т. к. настройки транка в приоритете.

9.4.1 Логика работы транковой группы для распределения вызовов на транки, входящие в ее состав

1. Распределение вызовов без использования алгоритма балансировки:

Все исходящие вызовы, маршрутизируемые через транковую группу, используют первый транк в группе. В случае недоступности транка или при совпадении ответа с маской из списка причин отбоя, вызов будет направлен через следующий транк в группе.

Пример:

На ESBC натроена транковая группа TRUNK_GROUP, в состав которой входят 3 транка (TRUNK_1, TRUNK_2 и TRUNK_3). Приходит вызов и по правилу маршрутизации уходит на эту транковую группу. В результате ESBC совершает попытку вызова в первый транк в составе транковой группы (TRUNK_1), если транк недоступен, то происходит попытка позвонить во второй транк (TRUNK_2). Если попытка вызова также неуспешна, то будет попытка позвонить в последний транк (TRUNK_3). Если попытка также неуспешна, то вызов на первом плече отбивается. Если на каком-то из транков пришел ответ 200OK, то вызов устанавливается.

Все последующие вызовы также будут сначала отправлены в TRUNK_1, и только в случае неудачи будут попытки позвонить в TRUNK_2 и TRUNK_3.

2. Распределение вызовов без использования алгоритма балансировки, но с включенной опцией **pick-once**:

Все исходящие вызовы, маршрутизируемые через транковую группу, используют первый транк в группе. В случае недоступности транка или при совпадении ответа с маской из списка причин отбоя, вызов **НЕ** будет направлен через следующий транк в группе, вызов на первом плече сразу отбивается.

Опцию **pick-once** можно включить в настройках таблицы маршрутизации при выборе действия *direct-to-trunk-group:*

```
vesbc#
vesbc# configure
vesbc(config)# esbc
vesbc(config-esbc)# route-table TO_TG_SSW
vesbc(config-esbc-route-table)# rule 0

#Включение опции pick-once при создании правила маршрутизации на транковую группу TG_SSW:
vesbc(config-esbc-route-table-rule)# action direct-to-trunk-group TG_SSW pick-once
```

3. Распределение вызовов с использованием алгоритма балансировки **round-robin** (опция **pick-once** выключена):

Каждый последующий исходящий вызов, маршрутизируемый через транковую группу, использует следующий транк в группе независимо от результата маршрутизации предыдущего вызова в данную транковую группу. В случае недоступности транка или при совпадении ответа с маской из списка причин отбоя, вызов будет направлен через следующий транк в группе.

Пример:

На ESBC настроена транковая группа TRUNK_GROUP, в состав которой входят 3 транка (TRUNK_1, TRUNK_2 и TRUNK_3). Приходит вызов и по правилу маршрутизации уходит на эту транковую группу. В результате ESBC совершает попытку вызова в первый транк в составе транковой группы (TRUNK_1), если вызов неуспешный (транк недоступен или ответ совпал с маской из списка причин отбоя), то происходит попытка позвонить во второй транк (TRUNK_2). Если попытка вызова также неуспешна, то будет попытка позвонить в последний транк (TRUNK_3). Если попытка также неуспешна, то вызов на первом плече отбивается. Если на каком-то из транков пришел ответ 200OK, то вызов устанавливается.

Второй вызов, который смаршрутизировался на данную транковую группу, сначала уйдет на TRUNK_2. Если вызов неуспешный, то ESBC совершит попытку позвонить в TRUNK_3 и потом в TRUNK_1. Если попытки неуспешны, то вызов на первом плече отбивается. По такому же принципу третий вызов сначала распределится в TRUNK_3, четвертый вызов — в TRUNK_1 и т. д.

Опция балансировки **round-robin** включается в настройках транковой группы:

```
vesbc#
vesbc# configure
vesbc(config)# esbc

#Создание и переход в настройки транковой группы TRUNK_GROUP:
vesbc(config-esbc)# trunk-group TRUNK_GROUP

#Добавление в состав транковой группы транков TRUNK_1, TRUNK_2 и TRUNK_3:
vesbc(config-esbc-trunk-group)# trunk 0 TRUNK_1
vesbc(config-esbc-trunk-group)# trunk 1 TRUNK_2
vesbc(config-esbc-trunk-group)# trunk 2 TRUNK_3

#Активация режима балансировки round-robin на траковой группе:
vesbc(config-esbc-trunk-group)# balancing round-robin
```

4. Распределение вызовов с использованием алгоритма балансировки **round-robin** (опция **pick-once** включена):

Каждый последующий исходящий вызов, маршрутизируемый через транковую группу, использует следующий транк в группе. В случае недоступности транка или при совпадении ответа с маской из списка причин отбоя вызов **НЕ** будет направлен через следующий транк в группе, вызов на первом плече сразу отбивается.

Пример:

В схеме из п. 3 первый вызов распределяется в TRUNK_1, если он отбивается, то первое плечо вызова сразу отбивается, попыток позвонить в TRUNK_2, TRUNK_3 нет. Второй вызов распределяется в TRUNK_2, третий — в TRUNK_3, четвертый — в TRUNK_1 и т. д.

9.5 Настройка SIP-транспортов

SIP-транспорт представляет точку входа/выхода сигнализации, т. е. это IP-адрес и порт, с которого ESBC будет отправлять и на который будет принимать сигнальные сообщения.

> Возможно использование IP-адреса, полученного по DHCP.

Пример:

Требуется, чтобы ESBC для передачи/приема сигнальных сообщений на встречную сторону использовал IP-адрес 192.168.16.113 порт 5065.

Решение:

Перейти к настройкам модуля управления конфигурацией ESBC:

```
vesbc#
vesbc# configure
vesbc(config)# esbc
```

Создать и настроить соответствующим образом SIP-транспорт:

```
#Создание/переход в настройки SIP-транспорта NEW_TRANSPORT:
vesbc(config-esbc)# sip transport NEW_TRANSPORT

#Назначить IP-адрес 192.168.16.113 для использования SIP-транспортом:
vesbc(config-esbc-sip-transport)# ip address 192.168.16.113

#Назначить порт 5065 для использования SIP-транспортом:
vesbc(config-esbc-sip-transport)# port 5065

#Выбрать протокол транспортного уровня, используемый для приема/передачи сообщений SIP:
vesbc(config-esbc-sip-transport)# mode udp-prefer
```

После привязки созданного SIP-транспорта к какому-либо направлению (транку или абонентскому интерфейсу) он будет использоваться для передачи/получения сигнальных SIP-сообщений на выбранных направлениях.

Поддержано несколько режимов работы с протоколами транспортного уровня, конфигурируется командой *mode* из примера выше. Режимы работы следующие:

- *tcp-only* — использовать только TCP-протокол;
- *tcp-prefer* — прием по UDP и TCP. Отправка по TCP. В случае если не удалось установить соединение по TCP, отправка производится по UDP;
- *tls* — использовать tls;
- *udp-only* — использовать только UDP-протокол;
- *udp-prefer* — прием по UDP и TCP. Отправка пакетов более 1300 байт по TCP, менее 1300 байт — по UDP;
- *ws* — использовать WebSocket;
- *wss* — использовать WebSocket Secure.

При использовании типа транспорта tls или wss возможно использование пользовательских сертификатов. Подробнее о пользовательских сертификатах см. в разделе Настройка профилей безопасности.

Пример настройки SIP-абонентов, использующих WebRTC есть в разделе Примеры настройки ESBC.

9.6 Настройка медиаресурсов

Медиаресурсы представляют собой диапазоны UDP-портов и IP-адресов, используемых ESBC для передачи/получения потоков RTP.

Возможно использование IP-адреса, полученного по DHCP.

Пример:

Требуется, чтобы ESBC для передачи медиатрафика использовал IP-адрес 192.168.16.113 и порты с 20000 до 30000.

Решение:

Перейти к настройкам модуля управления конфигурацией ESBC:

```
vesbc#
vesbc# configure
vesbc(config)# esbc
```

Создать и настроить соответствующим образом медиаресурс:

```
#Создание/переход в настройки медиаресурса MEDIA_1:
vesbc(config-esbc)# media resource MEDIA_1

#Назначить IP-адрес 192.168.16.113 для использования в медиаресурсах:
vesbc(config-esbc-media-resource)# ip address 192.168.16.113

#Настроить диапазон UDP-портов с 20000 до 30000 для использовании в медиаресурсах:
vesbc(config-esbc-media-resource)# port-range 20000-30000
```

После привязки созданного медиаресурса к какому-либо направлению (транку, транковой группе или абонентскому интерфейсу), он будет использоваться для передачи/получения потоков RTP на выбранных направлениях.

При использовании одинакового IP-адреса для разных медиаресурсов не допускается пересечение диапазонов портов между этими ресурсами.

9.7 Настройка таблиц маршрутизации

Схематично таблица маршрутизации выглядит следующим образом:

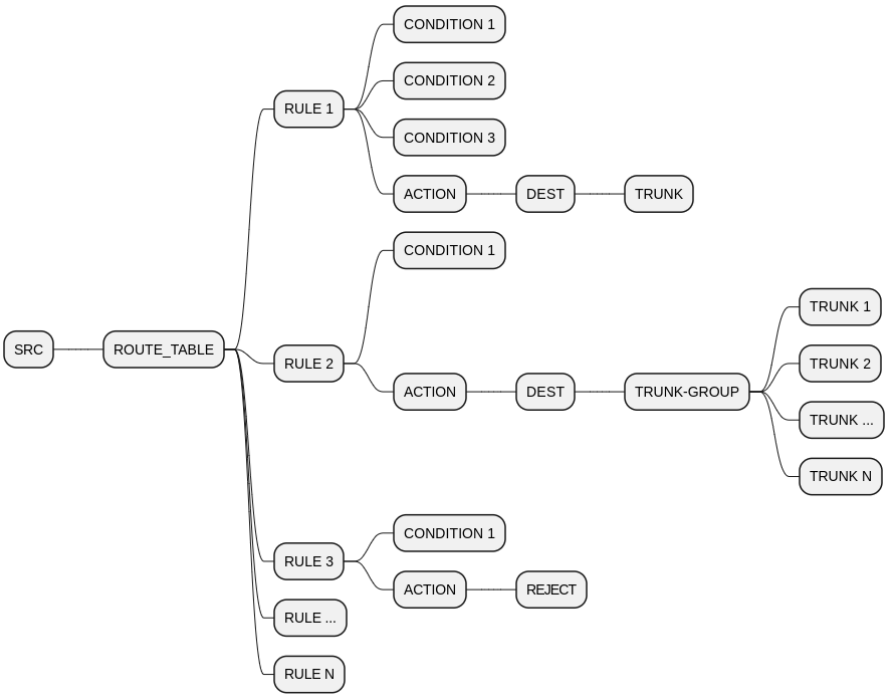


Таблица маршрутизации представляет собой набор правил и действий, по которым обрабатывается входящий вызов, и указывается исходящий транк (или транк-группа) для формирования исходящего вызова.

Таблицы маршрутизации применяются к входящим вызовам и могут быть настроены для транков, транк-групп и абонентских интерфейсов.

Таблица состоит из правил (RULE), правило обязательно должно содержать действие (ACTION), и, опционально, — условия (CONDITION), которые должны быть соблюдены для выполнения данного действия маршрутизации. Если условия отсутствуют, действия совершаются безусловно.
Действие — это операция, результатом которой будет являться конкретное направление.
В текущей версии в качестве направлений могут выступают транки и транк-группы.

Условия маршрутизации:

- безусловная маршрутизация — маршрутизация всех SIP-сообщений без анализа содержимого;
- маршрутизация по CgPN, CdPN — анализируются user-part из заголовков From и To в сообщении SIP;
- SIP-MESSAGE — маршрутизация по наличию любого совпадения в любой части SIP-сообщения.

Правила маршрутизации выбираются по порядку до тех пор, пока второе плечо не будет успешно согласовано, или не будет рассмотрено последнее правило. Если рассматривать на примере вызова, то роутинг будет выполняться до тех пор, тока второе плечо не примет вызов.

В случае маршрутизации на транк-группу действует тот же алгоритм. Т. е. проход осуществляется по всем транкам выбранной группы по порядку до тех пор, пока сессия не согласуется, или не будет выбран последний транк. Если после прохождения по всем транкам выбранной группы не удалось согласовать второе плечо, продолжается выбор оставшихся правил из таблицы маршрутизации.

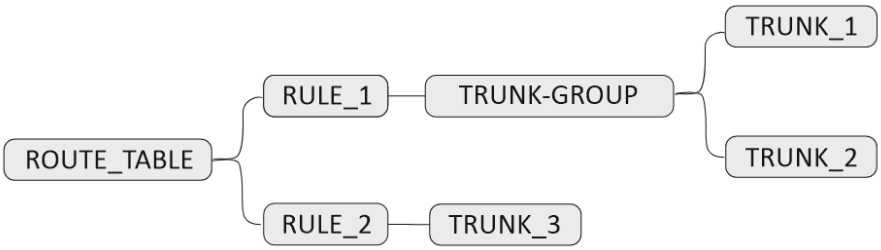
В общем, этот алгоритм можно описать так: **проход по всем направлениям, всех правил маршрутизации, пока сессия не будет согласована, или не будет рассмотрено последнее правило.**

Исключением является правило **Reject** — отбой входящей сессии. Это правило завершает проход по таблице маршрутизации.

Выбор следующего направления будет происходить:

- при внутренних сбоях, до согласования сессии;
- при отбое с встречной стороны, кроме 3xx кодов SIP.

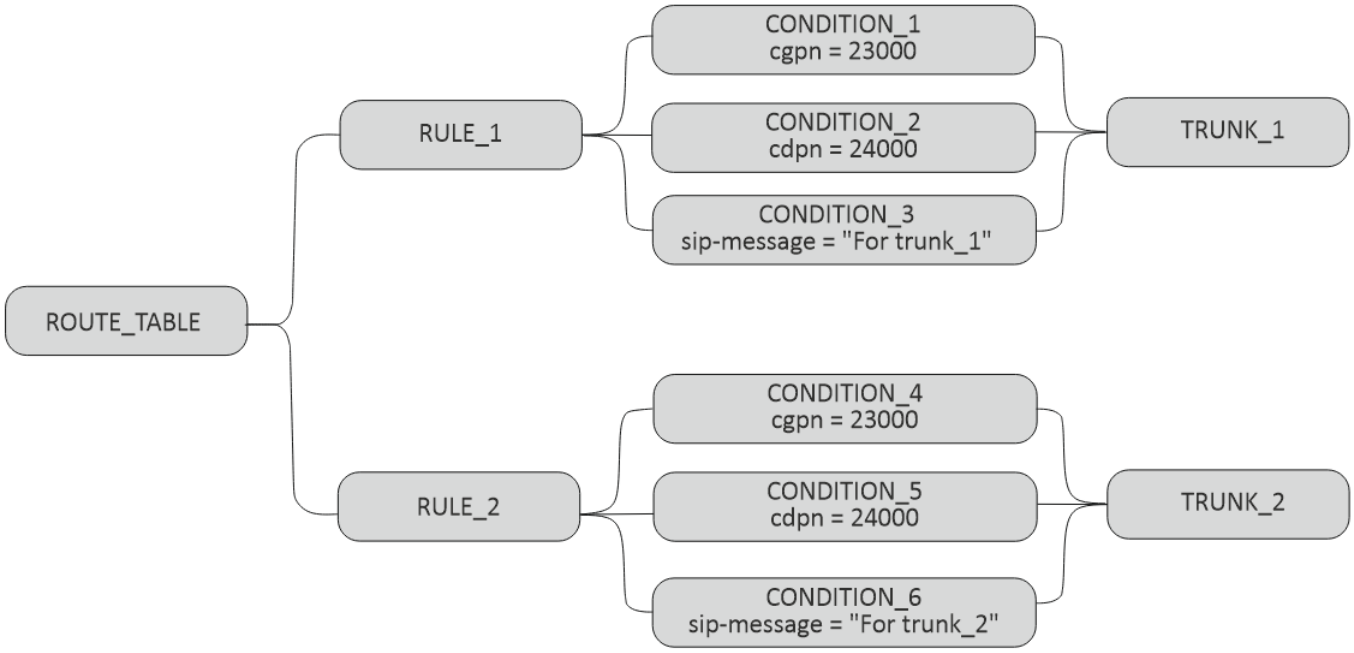
Пример перебора правил:



В таблице маршрутизации два правила, первое направляет вызов в TRUNK_GROUP, второе направляет вызов в TRUNK_3, условия нигде не настроены. Приходит вызов и начинает маршрутизироваться по данной таблице маршрутизации. В результате вызов уходит на TRUNK_GROUP и оттуда в TRUNK_1, в случае если вызов через TRUNK_1 не установился (например, транк недоступен), то маршрутизация продолжает выполняться, вызов отправляется в TRUNK_2. Если попытка вызова в TRUNK_2 также завершилась неудачно, ESBC переходит к RULE_2 и отправляет вызов в TRUNK_3. Если и здесь попытка установить вызов также оказалась неуспешной, то первое плечо отбивается, и вызов завершается, т. к. больше правил в таблице маршрутизации нет. Если попытка установить вызов успешна, то вызов устанавливается.

```
vesbc#
vesbc# configure
vesbc(config)# esbc
vesbc(config-esbc)# route-table ROUTE_TABLE

#Добавление первого правила с действием отправить вызов в транковую группу TRUNK_GROUP:
vesbc(config-esbc-route-table)# rule 0
vesbc(config-esbc-route-table-rule)# action direct-to-trunk-group TRUNK_GROUP
vesbc(config-esbc-route-table-rule)# exit

#Добавление второго правила с действием отправить вызов в транк TRUNK_3:
vesbc(config-esbc-route-table)# rule 1
vesbc(config-esbc-route-table-rule)# action direct-to-trunk TRUNK_3
```

Пример работы условий:



В таблице маршрутизации два правила, у обоих есть условия по CGPN, CDPN и SIP-MESSAGE. Например, приходит вызов, у которого номер А=23000, номер Б=24000, и кастомный заголовок "Trunk: For trunk_1". ESBC заходит в RULE_1 и анализирует условие CONDITION_1, условие истинно, далее происходит анализ условия из CONDITION_2, условие истинно, далее происходит анализ условия из CONDITION_3, условие также истинно. Значит правило RULE_1 подходит для маршрутизации, и вызов отправляется в TRUNK_1.

Рассмотрим вызов с номерами, которые подходят под условия из RULE_2.

Приходит вызов, у которого номер А=23000, номер Б=24000 и кастомный заголовок "Trunk: For trunk_2". ESBC заходит в RULE_1 и анализирует условие CONDITION_1, условие истинно, далее происходит анализ условия из CONDITION_2, условие истинно, далее происходит анализ условия из CONDITION_3, условие ложно. Значит правило не подходит (правило подходит, только если все условия внутри правила истинны). Далее ESBC переходит к RULE_2, анализирует условие CONDITION_4, условие истинно, далее происходит анализ условия из CONDITION_5, условие истинно, далее происходит анализ условия из CONDITION_6, условие также истинно. Значит правило RULE_2 подходит для маршрутизации, и вызов отправляется в TRUNK_2.

Если приходит вызов, который не подходит ни под одно правило, то такой вызов отбивается.

```
vesbc#
vesbc# configure
vesbc(config)# esbc
vesbc(config-esbc)# route-table ROUTE_TABLE

#Добавление первого правила с условиями CONDITION_1, CONDITION_2, CONDITION_3 и действием
отправить вызов в TRUNK_1:
vesbc(config-esbc-route-table)# rule 0
vesbc(config-esbc-route-table-rule)# condition 0 cgpn ^23000$
vesbc(config-esbc-route-table-rule)# condition 1 cdpn ^24000$
vesbc(config-esbc-route-table-rule)# condition 2 sip-message '.*For trunk_1.*'
vesbc(config-esbc-route-table-rule)# action direct-to-trunk TRUNK_1
vesbc(config-esbc-route-table-rule)# exit

#Добавление второго правила с условиями CONDITION_4, CONDITION_5, CONDITION_6 и действием
отправить вызов в TRUNK_2:
vesbc(config-esbc-route-table)# rule 1
vesbc(config-esbc-route-table-rule)# condition 0 cgpn ^23000$
vesbc(config-esbc-route-table-rule)# condition 1 cdpn ^24000$
vesbc(config-esbc-route-table-rule)# condition 2 sip-message '.*For trunk_2.*'
vesbc(config-esbc-route-table-rule)# action direct-to-trunk TRUNK_2
```

Синтаксис для написания условий

Для написания условий можно использовать регулярные выражения PCRE.

9.8 Настройка модификаторов

ESBC поддерживает два типа модификаторов — **common** и **sip**.

Модификаторы **common** позволяют модифицировать CdPN и CgPN без привязки к протоколу сигнализации. В текущей версии ПО поддерживается только протокол SIP. Учитывая это, при использовании модификаторов в транках и абонентских интерфейсов, модификаторами **common** можно изменять user part SIP URI заголовков To и From.

Модификаторы **sip** позволяют модифицировать любые заголовки сообщений SIP.

Таблицы модификаций применяются в транках, транковых группах и абонентских интерфейсах. Их можно подключить, как **out** — тогда правила будут применяться при отправке сообщения или, и как **in** — тогда правила применяются при получении сообщения. Таблица модификаций, используемая для транковой группы, будет использоваться только в том случае, если в транке, входящем в эту транковую группу, не настроена своя таблица.

В таблицах модификации для отбора значений (header pattern, header value, response-pattern, value-pattern, value, replacement и др.) используются регулярные выражения PCRE.

Перед использованием модификаторов рекомендуется ознакомиться с описанием синтаксиса регулярных выражений PCRE.

Допускается использование следующей конструкции при составлении регулярных выражений PCRE для помещения значений в локальные переменные (от 0 до 9) с помощью цифр, экранированных обратной чертой ('\1-9'). '\0' — весь текст:

```
value-pattern '(some)-(value)'
#Значения some и value заносятся в локальные переменные pcre 1 и 2 соответственно
replacement '\2-\1'
#Значения переменных меняются местами
```

Результат замены: value-some

Данные переменные используются в рамках одной модификации. Для использования переменных в разных модификациях одной таблицы модификаций используется модификатор типа **copy**.

> ⚠ При применении на транке/абонентском интерфейсе модификаторов обоих типов одновременно, используется следующий порядок их обработки в зависимости от направления модификации:
> - IN — сначала применяется модификатор sip, затем — модификатор common;
> - OUT — сначала применяется модификатор common, затем — sip.

9.8.1 Общие модификаторы

Пример использования модификатора **common**.

На ESBC настроена следующая конфигурация:

```
  route-table TO_UAS
    rule 0
      action direct-to-trunk UAS
    exit
  exit
  mod-table common COMMON_MOD
    mod 5 cgpn
      value-pattern '2(.+)'
      #Осуществляется выбор номеров, начинающихся с 2. Остальная часть номера сохраняется в
локальную переменную 1
      replacement '8\1'
      #Выполняется замена 2 на 8 и добавляется значение из переменной 1
    exit
    mod 10 cdpn
      value-pattern '23002'
      #Осуществляется выбор номера 23002
      replacement '22222'
      #Выполняется замена номера 23002 на 22222
    exit
  exit
  trunk sip UAC
    remote addr 192.168.80.26
    remote port 5070
    sip transport UAC
    route-table TO_UAS
    mod-table common in COMMON_MOD
    media resource 0 MEDIA
  exit
  trunk sip UAS
    remote addr 192.168.80.26
    remote port 5080
    sip transport UAS
    media resource 0 MEDIA
  exit
exit
```

Схема вызова:

На транк UAC приходит INVITE:

```
INVITE sip:24001@192.168.80.129:5080;line=76196f92c8f42f97c3b78125dd1b842c SIP/2.0
Via: SIP/2.0/UDP 192.168.80.26:5070;rport;branch=z9hG4bK-294378-1-1
From: <sip:24001@192.168.80.26:5070>;tag=1
To: <sip:23002@192.168.80.129:5070>
Call-ID: 1-294378@192.168.80.26
Cseq: 1 INVITE
Contact: <sip:24001@192.168.80.26:5070>
Max-Forwards: 70
Allow: INVITE, ACK, BYE, CANCEL, UPDATE, SUBSCRIBE
Content-Type: application/sdp
Content-Length:   174

[SDP]...
```

В результате применения модификатора **COMMON_MOD** в транке UAC, из транка UAS будет отправлен INVITE:

```
INVITE sip:22222@192.168.80.26:5080 SIP/2.0
Via: SIP/2.0/UDP 192.168.80.129:5080;rport;branch=z9hG4bKPjWDx0A5VQhCqmg7Sf-wS7Huya0dESxrro
Max-Forwards: 70
From: <sip:84001@192.168.80.129>;tag=epoMSc5qF1.Pfc5pcypn80ONBKHCaO-x
To: <sip:22222@192.168.80.26>
Contact: <sip:84001@192.168.80.129:5080>
Call-ID: 326c0035a257a9f76185383b49df705f
CSeq: 9446 INVITE
Allow: PRACK, SUBSCRIBE, NOTIFY, REFER, INVITE, ACK, BYE, CANCEL, UPDATE
Supported: 100rel, replaces
Content-Type: application/sdp
Content-Length:   177

[SDP]...
```

В результате модификации mod 5 cgpn выполнена модификация CgPN 24001 на 84001, в результате mod 10 cdpn — модификация CdPN 23002 на 22222.

⚠ При использовании модификатора CgPN, помимо заголовка From, изменяется user part SIP URI заголовка Contact. При использовании модификатора CdPN, помимо заголовка To, изменяется user part SIP в Request-URI.

⚠ Модификаторы common, настроенные в качестве IN, могут влиять на результат маршрутизации при использовании в route-table условий (condition), т. к. правила route-table обрабатываются после применения модификации. Модификаторы, настроенные в качестве OUT, не влияют на результат маршрутизации.

⛔ Для сообщений REGISTER модификаторы common не применяются.

Описание всех команд для настройки общих модификаторов приведено в разделе Настройки общих модификаторов.

9.8.2 Модификаторы SIP

Данный тип модификации позволяет изменять любые заголовки сообщений SIP.

⚠ Процесс модификации заголовков отличается в зависимости от режима использования модификатора IN или OUT.

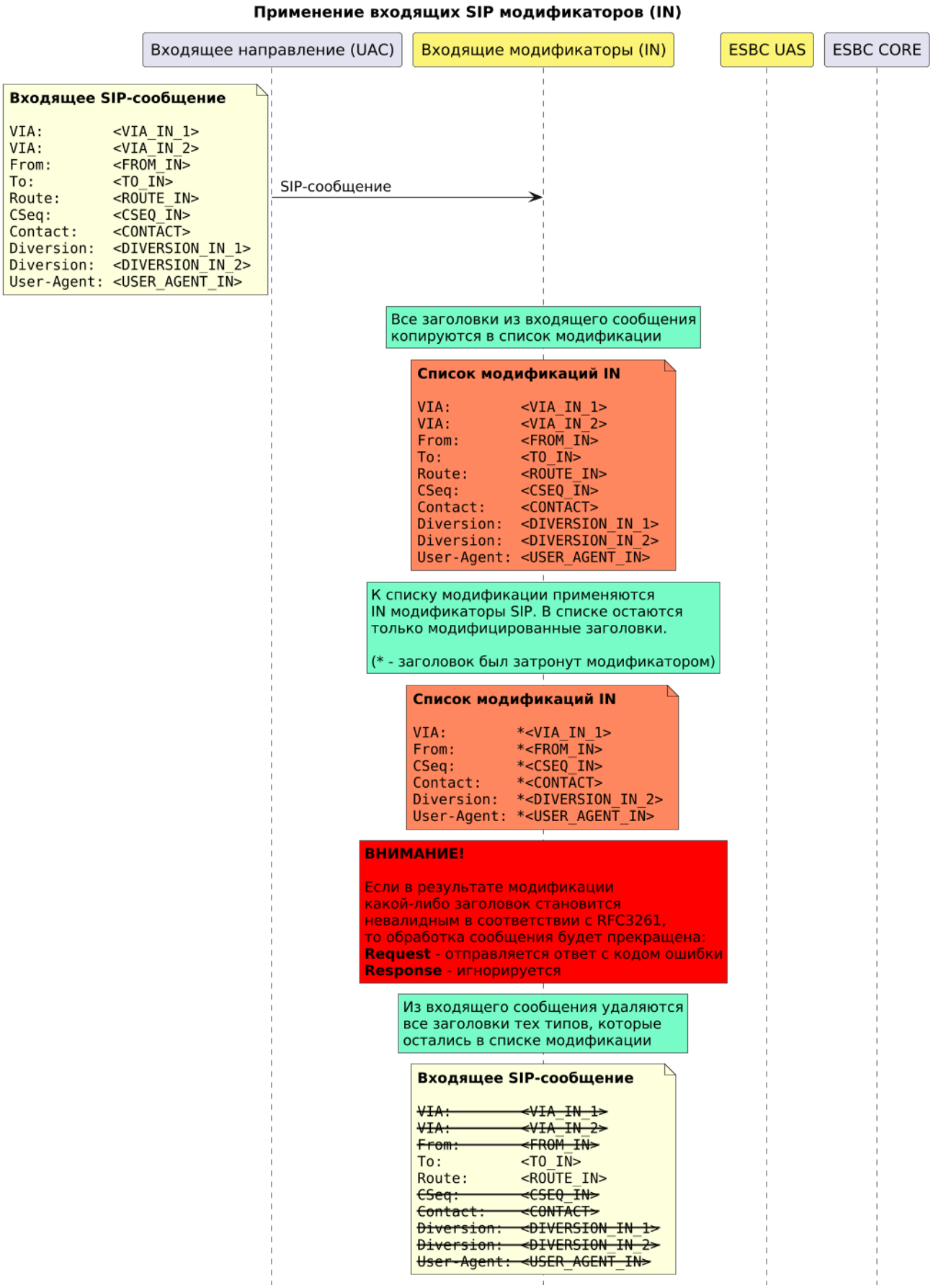
Существуют ограничения на модификацию основных заголовков sip, к которым относятся: Call-ID, From, To, Via, CSeq, Contact, Max-Forwards, Route, Record-Route, Content-Type, Content-Lenght, Require, Supported.

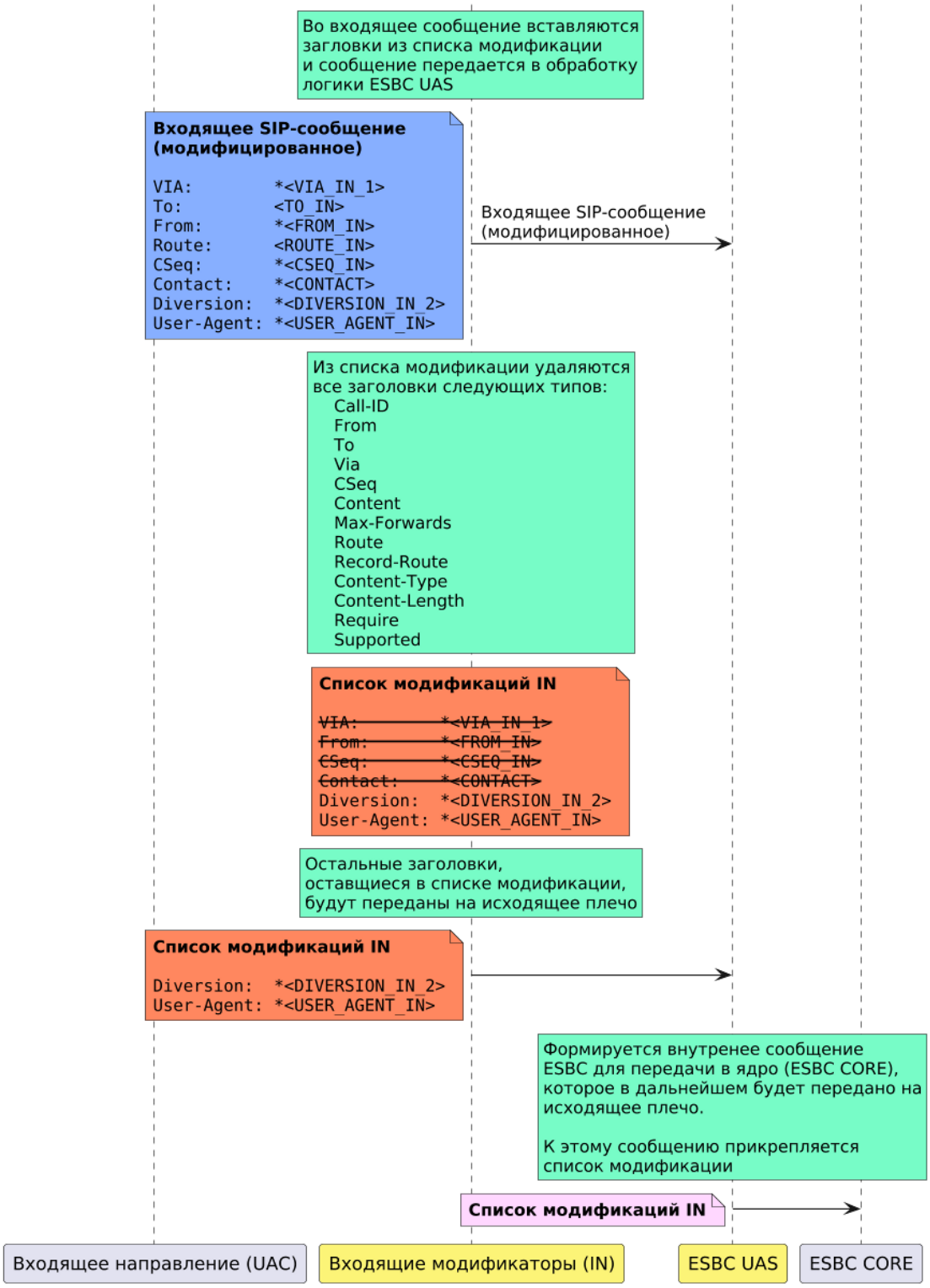
После применения к сообщению модификатора IN и использования модификаций основных заголовков, дальнейшая обработка диалога sip будет осуществляться в соответствии с модифицированным сообщением, т. к. в ядро системы попадает модифицированное сообщение. В результате в ответных сообщениях будут использоваться данные, которые могут отличаться от исходного сообщения. Модификация IN также влияет на дальнейшую маршрутизацию сообщения.

Применение к сообщению модификатора OUT и использования модификаций основных заголовков, изменяет только значения заголовков непосредственно перед отправкой, но не влияет на последующие сообщения в диалоге, т. к. исходное сообщение формируется ядром системы до применения модификаторов OUT.

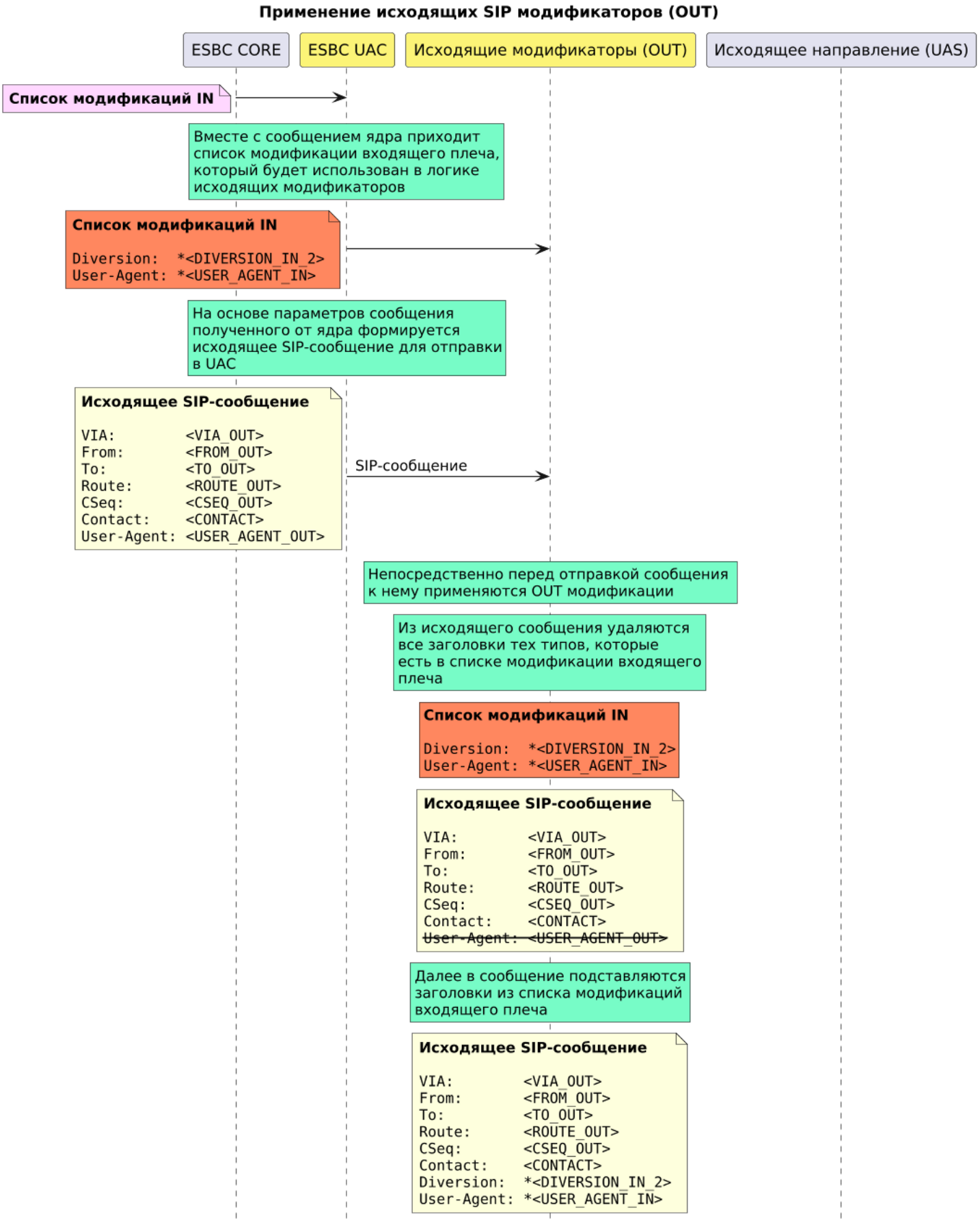
⛔ Применение модификаторов к основным заголовкам SIP может привести к нарушению обработки сообщений.

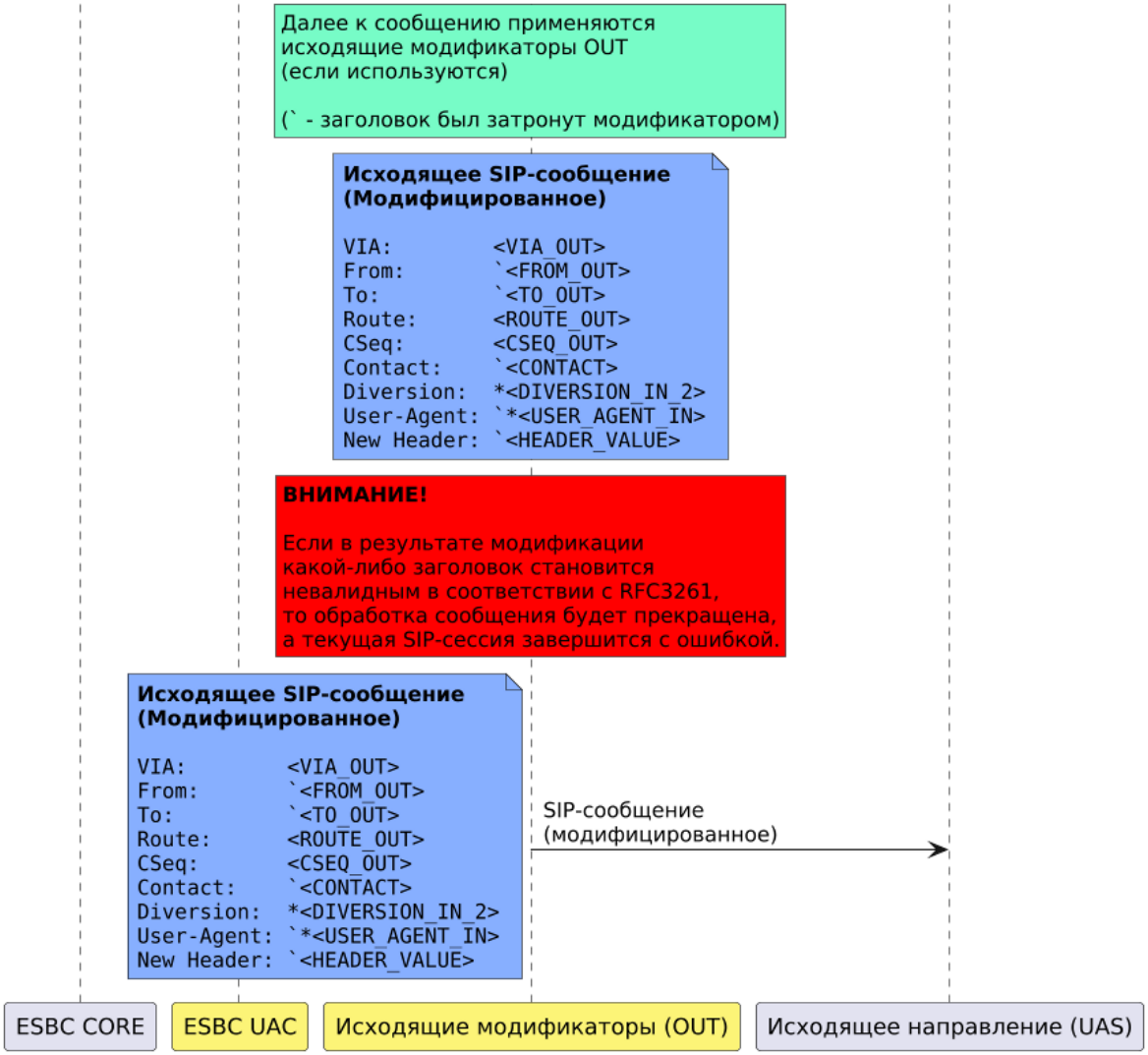
Логика обработки сообщения SIP при использовании IN-модификации:




Во входящее сообщение вставляются загловки из списка модификации и сообщение передается в обработку логики ESBC UAS
Входящее SIP-сообщение (модифицированное)
VIA: *<VIA_IN_1>
To: <TO_IN>
From: *<FROM_IN>
Route: <ROUTE_IN>
CSeq: *<CSEQ_IN>
Contact: *<CONTACT>
Diversion: *<DIVERSION_IN_2>
User-Agent: *<USER_AGENT_IN>
Входящее SIP-сообщение (модифицированное)
Из списка модификации удаляются все заголовки следующих типов:
Call-ID
From
To
Via
CSeq
Content
Max-Forwards
Route
Record-Route
Content-Type
Content-Length
Require
Supported
Список модификаций IN
VIA: *<VIA_IN_1>
From: *<FROM_IN>
CSeq: *<CSEQ_IN>
Contact: *<CONTACT>
Diversion: *<DIVERSION_IN_2>
User-Agent: *<USER_AGENT_IN>
Остальные заголовки, оставщиеся в списке модификации, будут переданы на исходящее плечо
Список модификаций IN
Diversion: *<DIVERSION_IN_2>
User-Agent: *<USER_AGENT_IN>
Формируется внутренее сообщение ESBC для передачи в ядро (ESBC CORE), которое в дальнейшем будет передано на исходящее плечо.
К этому сообщению прикрепляется список модификации
Список модификаций IN
Входящее направление (UAC)
Входящие модификаторы (IN)
ESBC UAS
ESBC CORE

Логика обработки сообщения SIP при использовании OUT-модификации:





Модификаторы SIP позволяют гибко осуществлять выбор требуемого метода (Request) или ответного сообщения (Response) по конкретному коду. Для этого используются команды:

- ***sip method pattern <PATTERN>*** — необходимый метод или несколько методов задается при помощи регулярного выражения PCRE.
- ***sip method type <TYPE>*** — необходимый метод выбирается из списка шести базовых методов стандарта RFC3261 (REGISTER, INVITE, ACK, CANCEL, BYE и OPTIONS).
- ***sip response-pattern <PATTERN>*** — необходимый код ответного сообщения задается при помощи регулярного выражения PCRE.

Команда *sip method type* аналогична команде *sip method pattern* и может использоваться в случае, когда модификацию требуется осуществлять только в одном из шести вышеуказанных методов. При использовании этой команды нет необходимости использовать pattern для написания регулярного выражения, достаточно выбрать метод из списка. Команды *sip method type* и *sip method pattern* являются взаимоисключающими.

При использовании *pattern* имейте в виду, что по умолчанию синтаксис PCRE является регистрозависимым. Поэтому, например, паттерн "sip method pattern INVITE" не эквивалентен паттерну "sip method pattern invite" и отбор метода INVITE **не будет** осуществляться при использовании выражения "sip method pattern invite".

⚠ При настройке модификатора обязательно следует указывать командами выше, для каких методов и кодов ответа он будет применяться. Иначе модификация не будет применена ни к одному сообщению.

Примеры отбора сообщений SIP для модификации:

Требуется применять модификатор, который добавляет заголовок Test_header со значением test_value только в сообщение INVITE:

```
mod-table sip SIP_MOD
  mod 1 add
# Т.к. по условию требуется добавлять заголовок только в сообщения INVITE, можно
воспользоваться командой "sip method type"
    sip method type Invite
    header name Test_header
    header value test_value
  exit
```

Требуется применять модификатор, который добавляет заголовок Test_header со значением test_value только в сообщения INVITE, BYE и в ответы 200 OK:

```
mod-table sip SIP_MOD
  mod 1 add
# Т.к. по условию требуется добавлять заголовок в INVITE и BYE, надо воспользоваться командой
 "sip method pattern"
    sip method pattern INVITE|BYE
# Для добавления заголовка в ответы 200 OK следует добавить команду "sip response-pattern"
    sip response-pattern 200
    header name Test_header
    header value test_value
  exit
```

Требуется применять модификатор, который добавляет заголовок Test_header со значением test_value во все запросы и ответы:

```
mod-table sip SIP_MOD
  mod 1 add
# Т.к. по условию требуется добавлять заголовок во все методы, используется отбор любых
значений
    sip method pattern .+
# Т.к. по условию требуется добавлять заголовок во все ответы, используется отбор любых
значений
    sip response-pattern .+
    header name Test_header
    header value test_value
  exit
```

Требуется применять модификатор, который добавляет заголовок Test_header со значением test_value только в предварительные ответы 100–199:

```
mod-table sip SIP_MOD
  mod 1 add
# Т.к. по условию требуется добавлять заголовок во все ответы от 100 до 199, используется,
например, регулярное выражение '1\d{2}'
    sip response-pattern '1\d{2}'
    header name Test_header
    header value test_value
  exit
```

Поддерживаемые модификации

Поддерживаются следующие типы модификации:

- **add** — добавление заголовка.
- **no-transit** — удаление заголовка. Данная модификация применяется только при использовании в качестве **out** (таблицы **in** всегда удаляют все заголовки, полученные в сообщении из сети).
- **replace** — замена заголовка.
- **transit** — передача заголовка. Данная модификация применяется только при использовании в качестве **in** (таблицы **out** всегда передают все заголовки, полученные с другого плеча).
- **copy** — позволяет скопировать значение или часть значения заголовка в переменную для использования этого значения в модификаторах **add** или **transit** в рамках одной таблицы модификаций (на одном плече вызова).

Порядок применения модификаций в таблице

Модификации в рамках одной таблицы применяются последовательно ко всем заголовкам в порядке, добавленном в конфигурации, т. е. первая модификация применяется ко всем заголовкам, затем вторая модификация применится ко всем заголовкам и т. д.

В результате если какой-либо заголовок был добавлен модификацией add, а затем этот же заголовок был указан в правиле no-transit, то в исходящем сообщении этот заголовок не будет передан.

Пример:

Таблица модификации SIP_MOD используется в качестве OUT:

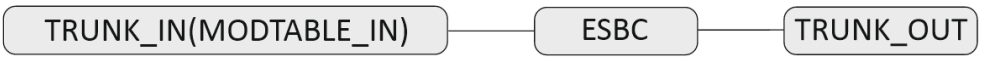
```
mod-table sip SIP_MOD
  mod 1 add
    sip method pattern '.+'
    sip response-pattern '.+'
    header name Test_header
    header value Test_value
  exit
  mod 2 no-transit
    sip header-pattern 'Test_header'
    sip method pattern '.+'
    sip response-pattern '.+'
    value-pattern 'Test_value'
  exit
```

Заголовок Test_header не будет передан.

Описание всех команд для настройки общих модификаторов приведено в разделе Настройки SIP-модификаторов.

Модификатор добавления заголовка (add)

Схема:



В конфигурации есть два транка, настроена маршрутизация. Вызов, который приходит из TRUNK_IN, уходит в TRUNK_OUT. Требуется, чтобы в запросе INVITE, который отправляется в TRUNK_OUT, был добавлен заголовок Test_header со значением example string.

```
vesbc#
vesbc# configure
vesbc(config)# esbc

#Создание таблицы модификаторов MODTABLE_IN:
vesbc(config-esbc)# mod-table sip MODTABLE_IN
vesbc(esbc-mod-table)#

#Добавление в таблицу модификаторов правила на добавление заголовка:
vesbc(esbc-mod-table)# mod 0 add
vesbc(esbc-mod-table-modification)#

#Выбор запроса, в котором будет добавлен заголовок (в данном случае INVITE):
vesbc(esbc-mod-table-modification)# sip method type Invite

#Указать название заголовка, который необходимо вставить (в данном случае Test_header):
vesbc(esbc-mod-table-modification)# header name Test_header

#Указать содержимое заголовка, которое необходимо вставить (в данном случае example string):
vesbc(esbc-mod-table-modification)# header value "example string"

vesbc(esbc-mod-table-modification)# exit
vesbc(esbc-mod-table)# exit

#Привязать таблицу модификаторов к входящему транку TRUNK_IN:
vesbc(config-esbc)# trunk sip TRUNK_IN
vesbc(config-esbc-trunk-sip)# mod-table sip in MODTABLE_IN

#Применить и подтвердить изменения:
vesbc(config-esbc-trunk-sip)# do commit
Configuration has been successfully applied and saved to flash. Commit timer started, changes
will be reverted in 600 seconds.
vesbc(config-esbc-trunk-sip)# do confirm
Configuration has been confirmed. Commit timer canceled.
```

После внесения изменений в конфигурацию с TRUNK_IN приходит следующий INVIITE:

```
INVITE sip:24000@192.168.114.130:5461 SIP/2.0
Via: SIP/2.0/UDP 192.168.114.129:5461;branch=z9hG4bK-372660-1-5
From: "Simple UAC send bye" <sip:24001@192.168.114.130;cpc=priority>;tag=1372660
To: "24000" <sip:24000@192.168.114.130>
Call-ID: 1-372660@192.168.114.129
CSeq: 1 INVITE
Contact: <sip:24001@192.168.114.129:5461>
Max-Forwards: 70
Content-Type: application/sdp
Content-Length: 149

v=0
o=tester 123456 654321 IN IP4 192.168.114.129
s=A conversation
c=IN IP4 192.168.114.129
t=0 0
m=audio 8338 RTP/AVP 8
a=rtpmap:8 PCMA/8000
```

На TRUNK_OUT отправляется уже модифицированный INVITE с добавленным заголовком:

```
INVITE sip:24000@192.168.114.129:5460 SIP/2.0
Via: SIP/2.0/UDP 192.168.114.130:5460;rport;branch=z9hG4bKPj-fvzSQlwN2zoMaGUR5JCLMkjmkBV3Vz1
Max-Forwards: 70
From: "Simple UAC send bye" <sip:24001@192.168.114.130>;tag=l2jkRSMeumV03IdhjPntOt7l0XBKy-Ln
To: "24000" <sip:24000@192.168.114.129>
Contact: <sip:24001@192.168.114.130:5460;transport=udp>
Call-ID: P-W.2oee.2vJw0JoaFbNkRDvnxY40FoP
CSeq: 30738 INVITE
Allow: PRACK, SUBSCRIBE, NOTIFY, REFER, INVITE, ACK, BYE, CANCEL, UPDATE
Supported: timer, 100rel, replaces
Session-Expires: 1800
Min-SE: 90

#Добавленный через таблицу модификаторов заголовок:
Test_header: example string
Content-Type: application/sdp
Content-Length: 157

v=0
o=tester 3927594021 3927594021 IN IP4 192.168.114.130
s=A conversation
c=IN IP4 192.168.114.130
t=0 0
m=audio 8062 RTP/AVP 8
a=rtpmap:8 PCMA/8000
```

Модификатор передачи заголовка (transit)

Схема:

В конфигурации настроено два транка и настроена маршрутизация из транка TRUNK_IN в TRUNK_OUT. Требуется передать заголовок "User-Agent" из входящего INVITE, только если в нем указано значение "TestUA".

⚠ По умолчанию все необязательные и пользовательские заголовки удаляются на входящем плече и не передаются на исходящее плечо.

Решение:

Создаем таблицу модификации MODTABLE_IN:

```
vesbc#
vesbc# configure
vesbc(config)# esbc
#Создаем модификаторов MODTABLE_IN:
vesbc(config-esbc)# mod-table sip MODTABLE_IN
vesbc(esbc-mod-table)#
#Добавляем в таблицу модификаторов правила для транзита заголовков:
vesbc(esbc-mod-table)# mod 0 transit
vesbc(esbc-mod-table-modification)#
#Выбираем метод, в котором будет осуществляться поиск заголовка (по условиям задачи - INVITE):
vesbc(esbc-mod-table-modification)# sip method type Invite
#Указываем паттерн для выбора названия заголовка, который необходимо передавать (по условиям
задачи - User-Agent):
vesbc(esbc-mod-table-modification)# sip header-pattern User-Agent
#Указать содержимое заголовка, при совпадении с которым заголовок будет передан (по условиям
задачи - TestUA):
vesbc(esbc-mod-table-modification)# value-pattern TestUA
vesbc(esbc-mod-table-modification)# exit
vesbc(esbc-mod-table)# exit
```

Используем созданную таблицу в качестве IN для транка TRUNK_IN:

```
vesbc(config-esbc)# trunk sip TRUNK_IN
vesbc(config-esbc-trunk-sip)# mod-table sip in MODTABLE_IN
vesbc(config-esbc-trunk-sip)# do commit
vesbc(config-esbc-trunk-sip)# do confirm
```

В результате, когда в транке TRUNK_IN будет получен INVITE, содержащий заголовок "User-Agent" со значением "TestUA", в исходящем INVITE также будет присутствовать этот заголовок:

```
# INVITE полученный в TRUNK_IN:
INVITE sip:23002@192.168.23.199:5070 SIP/2.0
Via: SIP/2.0/UDP 192.168.23.200:5070;rport;branch=z9hG4bK-1763439-1-1
From: sipp <sip:24001@192.168.23.200:5070>;tag=1
To: sut <sip:23002@192.168.23.199:5070>
Call-ID: 1-1763439@192.168.23.200
Cseq: 1 INVITE
Contact: <sip:24001@192.168.23.200:5070>
Max-Forwards: 70
Subject: Performance Test
Allow: INVITE, ACK, BYE, CANCEL, PRACK, REGISTER, INFO, REFER, NOTIFY, OPTIONS, SUBSCRIBE,
MESSAGE, UPDATE, PUBLISH
Content-Type: application/sdp
User-Agent: TestUA
Content-Length:   240

[SDP]

# INVITE отправленный с TRUNK_OUT:
INVITE sip:23002@192.168.80.26:5080 SIP/2.0
Via: SIP/2.0/UDP 192.168.80.129:5080;rport;branch=z9hG4bKPj88572ec4-0af0-4323-abc6-fe1db1ea6e37
Max-Forwards: 70
From: "sipp" <sip:24001@192.168.80.129>;tag=7da4c833-38da-4523-89d7-adc88b581397
To: "sut" <sip:23002@192.168.80.26>
Contact: <sip:24001@192.168.80.129:5080;transport=udp>
Call-ID: d21bdedb499dbfe992a939a27255c536
CSeq: 6199 INVITE
Allow: INVITE, ACK, BYE, CANCEL, PRACK, REGISTER, INFO, REFER, NOTIFY, OPTIONS, SUBSCRIBE,
MESSAGE, UPDATE
Supported: 100rel, replaces, ice
User-Agent: TestUA
Content-Type: application/sdp
Content-Length:   241

[SDP]
```

⚠ Данный модификатор сработает только в случае, когда значение заголовка User-Agent будет "TestUA". Если значение отличается, то заголовок User-Agent не будет передаваться на второе плечо. Для передачи заголовка с любым значением не следует использовать команду *value-pattern* в модификаторе, требуется указать в ней все возможные варианты, например, *value-pattern .**

Модификатор удаления заголовка (no-transit)

Схема:

В конфигурации есть два транка, настроена маршрутизация. Вызов, который приходит из TRUNK_IN, уходит в TRUNK_OUT. В TRUNK_OUT отправляется запрос INVITE, в теле которого есть заголовок Test_header. Требуется, чтобы в запросе INVITE, который отправляется в TRUNK_OUT, вырезался заголовок Test_header, если в его содержимом есть "example string".

> ⚠ По умолчанию все необязательные и пользовательские заголовки удаляются на входящем плече и не передаются на исходящее плечо. В данном примере демонстрируется модификация только для исходящего плеча (TRUNK_OUT), поэтому подразумевается что на входящем плече (TRUNK_IN) настроен модификатор transit для заголовка Test_header.

```
vesbc#
vesbc# configure
vesbc(config)# esbc

#Создание таблицы модификаторов MODTABLE_OUT:
vesbc(config-esbc)# mod-table sip MODTABLE_OUT
vesbc(esbc-mod-table)#

#Добавление в таблицу модификаторов правила на удаление заголовка:
vesbc(esbc-mod-table)# mod 0 no-transit
vesbc(esbc-mod-table-modification)#

#Выбор запроса, в котором будет удален заголовок (в данном случае INVITE):
vesbc(esbc-mod-table-modification)# sip method type Invite

#Указать название заголовка, который необходимо удалить (в данном случае Test_header):
vesbc(esbc-mod-table-modification)# sip header-pattern Test_header

#Указать содержимое заголовка, при совпадении с которым заголовок будет удален (в данном случае
 example string):
vesbc(esbc-mod-table-modification)# value-pattern "example string"

vesbc(esbc-mod-table-modification)# exit
vesbc(esbc-mod-table)# exit

#Привязать таблицу модификаторов к исходящему транку TRUNK_OUT:
vesbc(config-esbc)# trunk sip TRUNK_OUT
vesbc(config-esbc-trunk-sip)# mod-table sip out MODTABLE_OUT

#Применить и подтвердить изменения:
vesbc(config-esbc-trunk-sip)# do commit
Configuration has been successfully applied and saved to flash. Commit timer started, changes
will be reverted in 600 seconds.
vesbc(config-esbc-trunk-sip)# do confirm
Configuration has been confirmed. Commit timer canceled.
```

До внесения изменений в конфигурацию в TRUNK_OUT отправлялся следующий INVIITE:

```
INVITE sip:24000@192.168.114.129:5460 SIP/2.0
Via: SIP/2.0/UDP 192.168.114.130:5460;rport;branch=z9hG4bKPjjju.7u4O03Aty93vQq0Q1huigSIqGVIr
Max-Forwards: 70
From: "Simple UAC send bye" <sip:24001@192.168.114.130>;tag=CW.53L5FPJAUBsiRspMYqtjTtOTzZxHg
To: "24000" <sip:24000@192.168.114.129>
Contact: <sip:24001@192.168.114.130:5460;transport=udp>
Call-ID: V4OORObjNahUbinXtA648s9eI2kjE5cCI
CSeq: 18905 INVITE
Allow: PRACK, SUBSCRIBE, NOTIFY, REFER, INVITE, ACK, BYE, CANCEL, UPDATE
Supported: timer, 100rel, replaces
Session-Expires: 1800
Min-SE: 90
#Заголовок, который должен быть удален:
Test_header: example string
Content-Type: application/sdp
Content-Length:   157

v=0
o=tester 3927595234 3927595234 IN IP4 192.168.114.130
s=A conversation
c=IN IP4 192.168.114.130
t=0 0
m=audio 8066 RTP/AVP 8
a=rtpmap:8 PCMA/8000
```

После внесения изменений в конфигурацию в TRUNK_OUT отправляется следующий INVITE (заголовок Test_header отсутствует):

```
INVITE sip:24000@192.168.114.129:5460 SIP/2.0
Via: SIP/2.0/UDP 192.168.114.130:5460;rport;branch=z9hG4bKPjz8Y5BfoTrBQlqecLCu34TIyYn-6rX5dH
Max-Forwards: 70
From: "Simple UAC send bye" <sip:24001@192.168.114.130>;tag=qTwcY3ZHvA6SHvuRsoo7w40r9yXzjEEp
To: "24000" <sip:24000@192.168.114.129>
Contact: <sip:24001@192.168.114.130:5460;transport=udp>
Call-ID: yHvNLSIvp0DQYSRFPRpfgVUv9U0uKEHT
CSeq: 10147 INVITE
Allow: PRACK, SUBSCRIBE, NOTIFY, REFER, INVITE, ACK, BYE, CANCEL, UPDATE
Supported: timer, 100rel, replaces
Session-Expires: 1800
Min-SE: 90
Content-Type: application/sdp
Content-Length:   157

v=0
o=tester 3927597375 3927597375 IN IP4 192.168.114.130
s=A conversation
c=IN IP4 192.168.114.130
t=0 0
m=audio 8070 RTP/AVP 8
a=rtpmap:8 PCMA/8000
```

В случае если в заголовке Test_header будет содержимое, отличное от "example string", заголовок будет отправлен в TRUNK_OUT:

```
INVITE sip:24000@192.168.114.129:5460 SIP/2.0
Via: SIP/2.0/UDP 192.168.114.130:5460;rport;branch=z9hG4bKPj8e1WEAvAy16Bk8Vrj-VZiFK-bNOjnjY9
Max-Forwards: 70
From: "Simple UAC send bye" <sip:24001@192.168.114.130>;tag=R83mrTm4KQsFL1Bk87hTOB8e182yCSJ.
To: "24000" <sip:24000@192.168.114.129>
Contact: <sip:24001@192.168.114.130:5460;transport=udp>
Call-ID: eQueXFpyDZESB.hXK.uCGn7XL7TBUdmQ
CSeq: 8831 INVITE
Allow: PRACK, SUBSCRIBE, NOTIFY, REFER, INVITE, ACK, BYE, CANCEL, UPDATE
Supported: timer, 100rel, replaces
Session-Expires: 1800
Min-SE: 90

#Заголовок Test_header с содержимым, отличным от "example string", не удаляется:
Test_header: new string
Content-Type: application/sdp
Content-Length:   157

v=0
o=tester 3927597832 3927597832 IN IP4 192.168.114.130
s=A conversation
c=IN IP4 192.168.114.130
t=0 0
m=audio 8074 RTP/AVP 8
a=rtpmap:8 PCMA/8000
```

Модификатор транзита и замены заголовка (replace)

Схема:

В конфигурации есть два транка, настроена маршрутизация. Вызов, который приходит из TRUNK_IN, уходит в TRUNK_OUT. Из TRUNK_IN приходит INVITE с заголовком Test_header: 123. Требуется, чтобы в TRUNK_OUT отправился INVITE с заголовком Test_header: 123456.

```
vesbc#
vesbc# configure
vesbc(config)# esbc

#Создание таблицы модификаторов MODTABLE_IN:
vesbc(config-esbc)# mod-table sip MODTABLE_IN
vesbc(esbc-mod-table)#

#Добавление в таблицу модификаторов правила на замену заголовка:
vesbc(esbc-mod-table)# mod 1 replace

#Выбор запроса, в котором будут заменяться заголовки:
vesbc(esbc-mod-table-modification)# sip method-type Invite

#Указать название заголовка, содержимое которого необходимо заменить:
vesbc(esbc-mod-table-modification)# sip header-pattern Test_header

#Указать место в содержимом заголовка, которое необходимо заменить (конец строки исходного
содержимого заголовка):
vesbc(esbc-mod-table-modification)# value-pattern $

#Добавить правило для подмены содержимого заголовка (к концу строки исходного содержимого
заголовка добавляется 456):
vesbc(esbc-mod-table-modification)# replacement 456

vesbc(esbc-mod-table-modification)# exit
vesbc(esbc-mod-table)# exit

#Привязать таблицу модификаторов к входящему транку TRUNK_IN:
vesbc(config-esbc)# trunk sip TRUNK_IN
vesbc(config-esbc-trunk-sip)# mod-table sip in MODTABLE_IN

#Применить и подтвердить изменения:
vesbc(config-esbc-trunk-sip)# do commit
Configuration has been successfully applied and saved to flash. Commit timer started, changes
will be reverted in 600 seconds.
vesbc(config-esbc-trunk-sip)# do confirm
Configuration has been confirmed. Commit timer canceled.
```

После внесения изменений в конфигурацию с TRUNK_IN приходит следующий INVIITE:

```
INVITE sip:24000@192.168.114.130:5461 SIP/2.0
Via: SIP/2.0/UDP 192.168.114.129:5461;branch=z9hG4bK-375510-1-5
From: "Simple UAC send bye"  <sip:24001@192.168.114.130;cpc=priority>;tag=1375510
To: "24000" <sip:24000@192.168.114.130>
Call-ID: 1-375510@192.168.114.129
CSeq: 1 INVITE
Contact: <sip:24001@192.168.114.129:5461>
Max-Forwards: 70

#Заголовок, который необходимо протранзитить и заменить:
Test_header: 123
Content-Type: application/sdp
Content-Length:   149

v=0
o=tester 123456 654321 IN IP4 192.168.114.129
s=A conversation
c=IN IP4 192.168.114.129
t=0 0
m=audio 7624 RTP/AVP 8
a=rtpmap:8 PCMA/8000
```

На TRUNK_OUT отправляется уже модифицированный INVITE с измененным заголовком:

```
INVITE sip:24000@192.168.114.129:5460 SIP/2.0
Via: SIP/2.0/UDP 192.168.114.130:5460;rport;branch=z9hG4bKPjIbcILUaVB0cQTFaGLLb7ccpnbTQIRvV3
Max-Forwards: 70
From: "Simple UAC send bye" <sip:24001@192.168.114.130>;tag=toP8wIO79wo47ChSYy69MFOyd4vhGRNF
To: "24000" <sip:24000@192.168.114.129>
Contact: <sip:24001@192.168.114.130:5460;transport=udp>
Call-ID: dLsiFI4-aD2faceSTLZu.-kuHfN.pJtG
CSeq: 22556 INVITE
Allow: PRACK, SUBSCRIBE, NOTIFY, REFER, INVITE, ACK, BYE, CANCEL, UPDATE
Supported: timer, 100rel, replaces
Session-Expires: 1800
Min-SE: 90

#Измененный заголовок:
Test_header: 123456
Content-Type: application/sdp
Content-Length:   157

v=0
o=tester 3927607871 3927607871 IN IP4 192.168.114.130
s=A conversation
c=IN IP4 192.168.114.130
t=0 0
m=audio 8090 RTP/AVP 8
a=rtpmap:8 PCMA/8000
```

Пример использования локальных переменных pcre в модификации replace (схема та же):

```
vesbc#
vesbc# configure
vesbc(config)# esbc

#Создание таблицы модификаторов MODTABLE_IN:
vesbc(config-esbc)# mod-table sip MODTABLE_IN
vesbc(esbc-mod-table)#

#Добавление в таблицу модификаторов правила на замену заголовка:
vesbc(esbc-mod-table)# mod 1 replace

#Выбор запроса, в котором будут заменяться заголовки:
vesbc(esbc-mod-table-modification)# sip method-type Invite

#Указать название заголовка, содержимое которого необходимо заменить:
vesbc(esbc-mod-table-modification)# sip header-pattern Date

#Указать место в содержимом заголовка, которое необходимо заменить (шаблон – дата в формате
 "год-месяц-число"):
vesbc(esbc-mod-table-modification)# value-pattern "(\\d{4})-(\\d{2})-(\\d{2})"

#Добавить правило для подмены содержимого заголовка (меняем формат даты на "месяц/число/год"):
vesbc(esbc-mod-table-modification)# replacement "\\2/\\3/\\1"

vesbc(esbc-mod-table-modification)# exit
vesbc(esbc-mod-table)# exit

#Привязать таблицу модификаторов к входящему транку TRUNK_IN:
vesbc(config-esbc)# trunk sip TRUNK_IN
vesbc(config-esbc-trunk-sip)# mod-table sip in MODTABLE_IN

#Применить и подтвердить изменения:
vesbc(config-esbc-trunk-sip)# do commit
Configuration has been successfully applied and saved to flash. Commit timer started, changes
will be reverted in 600 seconds.
vesbc(config-esbc-trunk-sip)# do confirm
Configuration has been confirmed. Commit timer canceled.
```

После внесения изменений в конфигурацию с TRUNK_IN приходит следующий INVIITE:

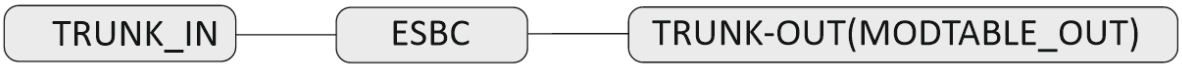
```
INVITE sip:135@10.25.72.151:5060 SIP/2.0
Via: SIP/2.0/UDP 10.25.72.35:5063;rport;branch=z9hG4bK-1104631-1-0
From: <sip:134@10.25.72.151:5060;user=phone>;tag=1
To: <sip:135@10.25.72.151:5060;user=phone>
Call-ID: 1-1104631@10.25.72.35
CSeq: 1 INVITE
Max-Forwards: 70
Supported: replaces, timer
Contact: <sip:134@10.25.72.35:5063>

#Заголовок, который необходимо протранзитить и изменить:
Date: 2024-09-10
Content-Type: application/sdp
Content-Length:   153
```

На TRUNK_OUT отправляется уже модифицированный INVITE с измененным заголовком:

```
Via: SIP/2.0/UDP 10.25.72.151:5060;rport;branch=z9hG4bKPjc5kLf-R0rh5Stla2eTvpoVAxOc0Jr.kX
Max-Forwards: 70
From: <sip:134@10.25.72.151>;tag=lMWgbj2x66hzNDHhP8ef8tWvB2HT2DwH
To: <sip:135@192.168.23.140>
Contact: <sip:134@10.25.72.151:5060;transport=udp>
Call-ID: c09c3761560702267daaee76eb769a9c
CSeq: 5021 INVITE
Allow: PRACK, SUBSCRIBE, NOTIFY, REFER, INVITE, ACK, BYE, CANCEL, UPDATE
Supported: 100rel, replaces

#Измененный заголовок:
Date: 09/10/2024
Content-Type: application/sdp
Content-Length:   163
```

Пример 2.

Схема:



В конфигурации есть два транка, настроена маршрутизация. Вызов, который приходит из TRUNK_IN, уходит в TRUNK_OUT. При отправке сообщения INVITE, полученного из TRUNK_IN в TRUNK_OUT, в host-part заголовков To и From будут использоваться IP-адрес, настроенный в качестве remote address в транке TRUNK_OUT, и IP-адрес sip-транспорта для TRUNK_OUT соответственно (при условии что в транке TRUNK_OUT не настроен домен).

Требуется при отправке INVITE заменять эти адреса на testdomain.loc.

Решение:

Конфигурация ESBC до использования модификаторов:

```
esbc
  media resource MEDIA_IN
    ip address 192.168.23.199
  exit
  media resource MEDIA_OUT
    ip address 192.168.80.129
  exit
  sip transport IN
    ip address 192.168.23.199
    port 5070
  exit
  sip transport OUT
    ip address 192.168.80.129
    port 5080
  exit
  route-table TO_TRUNK_OUT
    rule 0
      action direct-to-trunk TRUNK_OUT
    exit
  exit
  trunk sip TRUNK_IN
    sip transport IN
    route-table TO_TRUNK_OUT
    media resource 0 MEDIA_IN
    remote address 192.168.23.200
    remote port 5070
  exit
  trunk sip TRUNK_OUT
    sip transport OUT
    media resource 0 MEDIA_OUT
    remote address 192.168.80.26
    remote port 5080
  exit
exit
```

Т. к. в транке TRUNK_OUT не настроен домен, то в host-part заголовков To и From сообщения INVITE будут указаны IP-адреса в соответствии с конфигурацией:

```
INVITE sip:23002@192.168.80.26:5080 SIP/2.0
Via: SIP/2.0/UDP 192.168.80.129:5080;rport;branch=z9hG4bKPj11eb899a-a1c3-4659-b78d-4bba6bdc17ce
Max-Forwards: 70
From: "sipp" <sip:24001@192.168.80.129>;tag=c090d50d-4b15-4db1-94ac-3ea77fe3dd7d
To: "sut" <sip:23002@192.168.80.26>
Contact: <sip:24001@192.168.80.129:5080;transport=udp>
Call-ID: db38ba3ff093153f38b412372a1bed35
CSeq: 20022 INVITE
Allow: INVITE, ACK, BYE, CANCEL, PRACK, REGISTER, INFO, REFER, NOTIFY, OPTIONS, SUBSCRIBE,
MESSAGE, UPDATE
Supported: 100rel, replaces, ice
Content-Type: application/sdp
Content-Length:   241

[SDP]
```

Настраиваем модификатор **MOD_TABLE** для замены IP-адресов на testdomain.loc:

```
vesbc# configure
vesbc(config)# esbc
vesbc(config-esbc)# mod-table sip MOD_TABLE
#Создаем модификатор замены:
vesbc(esbc-mod-table)# mod 1 replace

#С помощью паттерна выбираем заголовки, в которых необходимо выполнить замену:
vesbc(esbc-mod-table-modification)# sip header-pattern '(From|To)'

#Указываем метод, в котором необходимо выполнить замену:
vesbc(esbc-mod-table-modification)# sip method pattern 'INVITE'

#Выбираем часть заголовка, которая начинается с символа @, содержит любое количество любых
символов и заканчивается символом >. Под это выражение попадает host-part заголовков:
vesbc(esbc-mod-table-modification)# value-pattern '@.*>'

#Указываем, что требуется заменить то, что мы получили в предыдущем правиле на @testdomain.loc>
:
vesbc(esbc-mod-table-modification)# replacement '@testdomain.loc>'
vesbc(esbc-mod-table-modification)# exit
vesbc(esbc-mod-table)# exit
vesbc(config-esbc)#
```

Используем это правило в TRUNK_OUT в кочестве OUT:

```
vesbc#
vesbc# configure
vesbc(config)# esbc
vesbc(config-esbc)# trunk sip TRUNK_OUT
vesbc(config-esbc-trunk-sip)# mod-table sip out MOD_TABLE
vesbc(config-esbc-trunk-sip)# do commit
vesbc(config-esbc-trunk-sip)# do confirm
```

Теперь в результате модификации в host-part заголовков To и From сообщения INVITE будет указан домен testdomain.loc:

```
INVITE sip:23002@192.168.80.26:5080 SIP/2.0
Via: SIP/2.0/UDP 192.168.80.129:5080;rport;branch=z9hG4bKPje431c80c-619a-43cc-a631-9ad3be4e6679
Max-Forwards: 70
From: "sipp" <sip:24001@testdomain.loc>;tag=0a5f2f31-e27e-4f7c-a3f8-70ca1d5a9f22
To: "sut" <sip:23002@testdomain.loc>
Contact: <sip:24001@192.168.80.129:5080;transport=udp>
Call-ID: 81a874656978d43e11d57e3662996fde
CSeq: 26399 INVITE
Allow: INVITE, ACK, BYE, CANCEL, PRACK, REGISTER, INFO, REFER, NOTIFY, OPTIONS, SUBSCRIBE,
MESSAGE, UPDATE
Supported: 100rel, replaces, ice
Content-Type: application/sdp
Content-Length:   241

[SDP]
```

Модификатор копирования (copy)

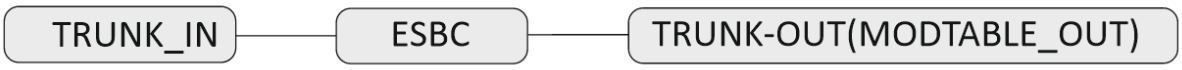
Работа с переменными модификатора

Значения переменных, полученных в модификаторе **copy**, можно использовать в модификаторах **replace** (поле replacement) и **add** (поле header value) в рамках одной таблицы модификации и только для текущего сообщения.

Например, при использовании модификатора **copy** в таблице на IN, для каждого входящего сообщения будет использоваться отдельный экземпляр таблицы, соответственно, в каждом случае значение переменных будет разным.

Подстроки ${name_perem} будут заменены на значение соответствующей переменной. Если переменная не задана — подстрока будет удалена. Длина переменной — до 128 символов.

Схема:



В конфигурации есть два транка, настроена маршрутизация. Вызов, который приходит из TRUNK_IN, уходит в TRUNK_OUT. В TRUNK_OUT отправляется запрос INVITE, в теле которого есть заголовок Diversion (предварительно следует настроить таблицу модификации на IN транка TRUNK_IN для транзита заголовка Diversion на второе плечо). Требуется, чтобы в запросе INVITE, который отправляется в TRUNK_OUT, вырезался заголовок Diversion, а его значение из user part было добавлено в display name заголовка From.

```
vesbc#
vesbc# configure
vesbc(config)# esbc

#Создание таблицы модификаторов MODTABLE_OUT:
vesbc(config-esbc)# mod-table sip MODTABLE_OUT
vesbc(esbc-mod-table)#

#Добавление в таблицу модификаторов правила copy для копирования значения user part в
переменную user_part:
vesbc(esbc-mod-table)# mod 0 copy
vesbc(esbc-mod-table-modification)#

#Выбор запроса, в котором будет использоваться модификатор copy (в данном случае INVITE):
vesbc(esbc-mod-table-modification)# sip method type Invite

#Указать название заголовка, из которого необходимо копировать значение (в данном случае
 Diversion):
vesbc(esbc-mod-table-modification)# sip header-pattern Diversion

#Указать содержимое заголовка, при совпадении с которым будет выполнено копирование в
переменную. В переменную будет скопирована та часть отбора, которая указана в скобках:
vesbc(esbc-mod-table-modification)# value-pattern '<sip:(.+)@'

#Указать переменную, в которую будет скопировано значение, указанное в скобках, в примере - (.
+):
vesbc(esbc-mod-table-modification)# variable-str 'user_part'
vesbc(esbc-mod-table-modification)# exit

#Добавление в таблицу модификаторов правила replace для замены заголовка From:
vesbc(esbc-mod-table)# mod 1 replace

#Указать название заголовка, в котором будет осуществляться замена:
vesbc(esbc-mod-table-modification)# sip header-pattern 'From'

#Выбор запроса, в котором будет использоваться модификатор replace (в данном случае INVITE):
vesbc(esbc-mod-table-modification)# sip method type Invite

#Указать часть содержимого заголовка, которую необходимо заменить:
vesbc(esbc-mod-table-modification)# value-pattern '.+ <sip:'

#Указать переменную user_part, которая содержит значение, полученное в модификации copy:
vesbc(esbc-mod-table-modification)# replacement '${user_part} <sip:$'
vesbc(esbc-mod-table-modification)# exit

#Добавление в таблицу модификаторов правила no-transit для удаления заголовка Diversion:
vesbc(esbc-mod-table)# mod 2 no-transit
vesbc(esbc-mod-table-modification)# sip header-pattern 'Diversion'
vesbc(esbc-mod-table-modification)# sip method type Invite
vesbc(esbc-mod-table-modification)# exit
vesbc(esbc-mod-table)# exit

#Привязать таблицу модификаторов к исходящему транку TRUNK_OUT:
vesbc(config-esbc)# trunk sip TRUNK_OUT
vesbc(config-esbc-trunk-sip)# mod-table sip out MODTABLE_OUT

#Применить и подтвердить изменения:
vesbc(config-esbc-trunk-sip)# do commit
```

```
Configuration has been successfully applied and saved to flash. Commit timer started, changes
will be reverted in 600 seconds.
vesbc(config-esbc-trunk-sip)# do confirm
Configuration has been confirmed. Commit timer canceled.
```

После внесения изменений в конфигурацию с TRUNK_IN приходит следующий INVIITE:

```
INVITE sip:24001@192.168.80.129:5080 SIP/2.0
Via: SIP/2.0/UDP 192.168.80.26:5070;rport;branch=z9hG4bK-473191-1-1
From: test <sip:24001@192.168.80.26:5070>;tag=1
To: sut <sip:23002@192.168.80.129:5070>
Call-ID: 1-473191@192.168.80.26
Cseq: 1 INVITE
Contact: <sip:24001@192.168.80.26:5070>
Max-Forwards: 70
Diversion: <sip:11111@test.loc>;reason=time-of-day
Subject: Performance Test
Allow: INVITE, ACK, BYE, CANCEL, UPDATE, SUBSCRIBE
Content-Type: application/sdp
Content-Length:   118

[SDP]...
```

На TRUNK_OUT отправляется уже модифицированный INVITE с измененным заголовком From и без заголовка Diversion:

```
INVITE sip:23002@192.168.80.26:5080 SIP/2.0
Via: SIP/2.0/UDP 192.168.80.129:5080;rport;branch=z9hG4bKPjbURYAQZxa2m1zsT6x.s6RQ28ONE4EifS
Max-Forwards: 70
From: "11111" <sip:24001@192.168.80.129>;tag=Jfl7n8XBMrh6vjCcB036Ogz6QX4BTDCo
To: "sut" <sip:23002@192.168.80.26>
Contact: <sip:24001@192.168.80.129:5080>
Call-ID: bbf5db1c228015eecddfe0d7079ce876
CSeq: 8798 INVITE
Allow: PRACK, SUBSCRIBE, NOTIFY, REFER, INVITE, ACK, BYE, CANCEL, UPDATE
Supported: 100rel, replaces
Content-Type: application/sdp
Content-Length:   119

[SDP]...
```

Использование системных переменных

В ESBC поддержано использование системных переменных в модификаторах **replace** (поле replacement) и **add** (поле header value).

Список системных переменных, которые можно использовать при модификации:

- *LOCAL_DOMAIN* — локальный домен;
- *LOCAL_ADDR* — локальный IP-адрес, сейчас то же самое, что LOCAL_HOST;
- *LOCAL_HOST* — локальный домен или IP-адрес;
- *LOCAL_PORT* — локальный порт;
- *REMOTE_DOMAIN* — домен удалённой стороны;
- *REMOTE_ADDR* — IP-адрес удалённой стороны;
- *REMOTE_HOST* — домен или IP-адрес удалённой стороны;
- *REMOTE_PORT* — порт удалённой стороны;

- *IFACE_TYPE* — тип интерфейса (TRUNK или USER);
- *IFACE_ID* — числовой идентификатор интерфейса;
- *IFACE_NAME* — имя интерфейса;
- *VERSION* — версия ESBC (x.y.z.patch);
- *TIMESTAMP* — текущее время в секундах (заполняется на момент применения модификации).

Синтаксис обращения к системным переменным:

```
${VAR_NAME}
```

Схема:

В конфигурации есть два транка, настроена маршрутизация. Вызов, который приходит из TRUNK_IN, уходит в TRUNK_OUT. Требуется, чтобы в запросе INVITE, который отправляется в TRUNK_OUT, добавлялся заголовок Call-Info с информацией об имени транка, на который отправляется запрос, и версией ESBC.

```
vesbc#
vesbc# configure
vesbc(config)# esbc

#Создание таблицы модификаторов MODTABLE_OUT:
vesbc(config-esbc)# mod-table sip MODTABLE_OUT
vesbc(esbc-mod-table)#

#Добавление в таблицу модификаторов правила copy для копирования значения user part в
переменную user_part:
vesbc(esbc-mod-table)# mod 0 add
vesbc(esbc-mod-table-modification)#

#Выбор запроса, в котором будет использоваться модификатор add (в данном случае INVITE):
vesbc(esbc-mod-table-modification)# sip method type Invite

#Указать название заголовка, который будет добавлен:
vesbc(esbc-mod-table-modification)# header name 'Call-Info'

#Указать содержимое заголовка с использованием системных переменных:
vesbc(esbc-mod-table-modification)# header value 'call to ${IFACE_NAME}; ESBC version: $
{VERSION}'
vesbc(esbc-mod-table-modification)# exit

#Привязать таблицу модификаторов к исходящему транку TRUNK_OUT:
vesbc(config-esbc)# trunk sip TRUNK_OUT
vesbc(config-esbc-trunk-sip)# mod-table sip out MODTABLE_OUT

#Применить и подтвердить изменения:
vesbc(config-esbc-trunk-sip)# do commit
Configuration has been successfully applied and saved to flash. Commit timer started, changes
will be reverted in 600 seconds.
vesbc(config-esbc-trunk-sip)# do confirm
Configuration has been confirmed. Commit timer canceled.
```

После внесения изменений в конфигурацию с TRUNK_IN приходит следующий INVIITE:

```
INVITE sip:23002@192.168.80.135:5060 SIP/2.0
Via: SIP/2.0/UDP 192.168.80.27:5061;rport;branch=z9hG4bK-2122485-1-1
From: "24001" <sip:24001@192.168.80.27:5061>;tag=1
To: "23002" <sip:23002@192.168.80.135:5060>
Call-ID: 1-2122485@192.168.80.27
Cseq: 1 INVITE
Contact: <sip:24001@192.168.80.27:5061>
Max-Forwards: 70
Subject: Performance Test
Allow: INVITE, ACK, BYE, CANCEL
Content-Type: application/sdp
Content-Length:   138

[SDP]...
```

На TRUNK_OUT отправляется уже модифицированный INVITE с заголовком Call-Info, который содержит имя вызываемой стороны и версию ESBC:

```
INVITE sip:23002@192.168.80.27:5063 SIP/2.0
Via: SIP/2.0/UDP 192.168.80.135:5060;rport;branch=z9hG4bKPj69d21930-f472-4e64-8555-6b68a532deae
Max-Forwards: 70
From: "24001" <sip:24001@192.168.80.135>;tag=f3db1c01-0c06-45cf-8b4d-a233070ae693
To: "23002" <sip:23002@192.168.80.27>
Contact: <sip:24001@192.168.80.135:5060;transport=udp>
Call-ID: 977eea09afecfc44932d4d9c1b2eeb15
CSeq: 6757 INVITE
Allow: INVITE, ACK, BYE, CANCEL
Supported: 100rel, replaces, ice, timer
Call-Info: call to TRUNK_OUT; ESBC version: 1.6.0.0085
Content-Type: application/sdp
Content-Length:   141

[SDP]...
```

9.9 Настройка SIP-профилей

В SIP-профиле настраивается конфигурация общих параметров SIP. Профиль используется в транках, транк-группах и абонентских интерфейсах.
В текущей версии ПО поддержаны следующие настройки:

- Контроль доступности направления;
- Список причин отбоя для перехода на следующее направление;
- Поведение при перенаправлении;
- Игнорирование OPTIONS.

Описание всех команд для настройки SIP-профилей приведено в разделе Настройки SIP-профиля.

9.9.1 Контроль доступности направления

Используется для периодической отправки keep-alive сообщений для контроля состояния встречной стороны.

В текущей версии ПО в качестве keep-alive сообщений используется метод OPTIONS.

По умолчанию keep-alive не используется. Для включения необходимо использовать команду *keepalive enable* в SIP-профиле.

Контроль осуществляется путем отправки сообщений OPTIONS с заданными интервалами *success-interval* (по умолчанию 60 сек.) и *failed-interval* (по умолчанию 20 сек.).

Алгоритм работы:

Сообщение OPTIONS отправляется только в случае, когда в транке отсутствует активность SIP после окончания периода success-interval. Т. е. в случае если через транк проходят вызовы с большей частотой, чем указано в настройке success-interval, то сообщения OPTIONS не будут отправляться на встречную сторону, т. к. очевидно, что направление доступно. Если после последнего отправленного или полученного сообщения SIP прошел период равный success-interval, то отправляется OPTIONS. При получении ответа на него (с любым статус-кодом) направление считается доступным. Сообщения OPTIONS будут отправляться с периодом success-interval до того момента, пока либо не появится активность SIP, либо не будут получены ответы на отправленные OPTIONS. Если не будет ответов на OPTIONS, транк считается недоступным, и сообщения OPTIONS будут отправляться с интервалом failed-interval до тех пор, пока транк снова не станет доступным.

Пример настройки:

```
vesbc#
vesbc# configure
vesbc(config)# esbc

#Создать SIP-профиль NEW_SIP_PROFILE:
vesbc(config-esbc)# sip profile NEW_SIP_PROFILE
vesbc(config-esbc-sip-profile)#

#Включить контроль доступности:
vesbc(config-esbc-sip-profile)# keepalive enable
vesbc(config-esbc-sip-profile)#

#Настроить интервалы контроля:
vesbc(config-esbc-sip-profile)# keepalive success-interval 120
vesbc(config-esbc-sip-profile)# keepalive failed-interval 30
vesbc(config-esbc-sip-profile)#

vesbc(config-esbc-sip-profile)# exit
vesbc(config-esbc)#

#Привязать SIP-профиль к транку TRUNK_OUT:
vesbc(config-esbc)# trunk sip NEW_TRUNK
vesbc(config-esbc-trunk-sip)# sip profile NEW_SIP_PROFILE
vesbc(config-esbc-trunk-sip)#

#Применить и подтвердить изменения:
vesbc(config-esbc-trunk-sip)# do commit
Configuration has been successfully applied and saved to flash. Commit timer started, changes
will be reverted in 600 seconds.
vesbc(config-esbc-trunk-sip)# do confirm
Configuration has been confirmed. Commit timer canceled.
```

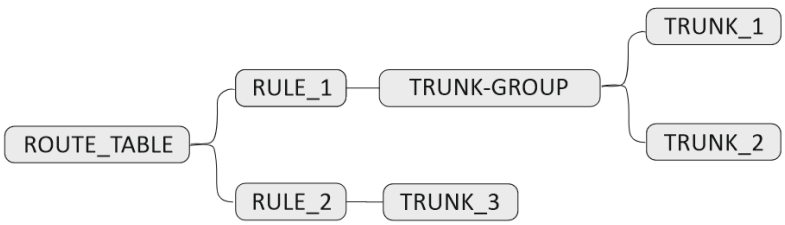
При использовании SIP-профиля с включенным контролем доступности, для абонентских интерфейсов отправка OPTIONS осуществляться не будет. Данная настройка используется только для контроля транков.

9.9.2 Список причин отбоя для перехода на следующее направление

Список причин отбоя для указания статус-кодов ответов SIP, по которым будет осуществляться перемаршрутизация вызовов и регистраций на альтернативное направление (следующий транк в транковой группе/следующее правило в таблице маршрутизации).

При создании маски для списка можно использовать регулярные выражения PCRE.

Пример использования:



В таблице маршрутизации два правила, первое — направляет вызов в TRUNK_GROUP, второе — направляет вызов в TRUNK_3.

```
vesbc#
vesbc# configure
vesbc(config)# esbc

#Создать список ответов:
vesbc(config-esbc)# cause-list sip LIST
vesbc(config-esbc-cause-list-sip)#

#Создать маску, по которой будут отбираться ответы для перемаршрутизации:
vesbc(config-esbc-cause-list-sip)# cause-mask 404
vesbc(config-esbc-cause-list-sip)# exit

#Создать SIP-профиль, привязать список к SIP-профилю:
vesbc(config-esbc)# sip profile SIP-PROFILE
vesbc(config-esbc-sip-profile)# cause-list LIST
vesbc(config-esbc-sip-profile)# exit

#Привязать к транковой группе TRUNK-GROUP SIP-профиль:
vesbc(config-esbc)# trunk-group TRUNK-GROUP
vesbc(config-esbc-trunk-group)# sip profile SIP-PROFILE
vesbc(config-esbc-trunk-group)#

#Применить и подтвердить изменения:
vesbc(config-esbc-trunk-group)# do commit
Configuration has been successfully applied and saved to flash. Commit timer started, changes
will be reverted in 600 seconds.
vesbc(config-esbc-trunk-group)# do confirm
Configuration has been confirmed. Commit timer canceled.
```

Входящий вызов начинает маршрутизироваться по таблице маршрутизации (ROUTE_TABLE). В результате вызов маршрутизируется по правилу RULE_1 на TRUNK_GROUP и оттуда в TRUNK_1.
TRUNK_1 недоступен, вызов отбивается по истечении Timer B, и происходит перемаршрутизация на TRUNK_2 (следующий транк в транковой группе). Из TRUNK_2 приходит ответ 404 Not Found, и т. к. код ответа совпадает с маской из списка, который используется в TRUNK-GROUP, то происходит маршрутизация на следующее направление. Поскольку в транковой группе больше нет транков, маршрутизация переходит к RULE_2, и вызов маршрутизируется в TRUNK_3.

Без использования списка причин отбоя, перемаршрутизация происходит только по недоступности транка.

Для абонентских интерфейсов, использование списка причин отбоя не влияет на маршрутизацию. Перемаршрутизация осуществляться не будет.

Перемаршрутизация абонентов

Вызов с зарегистрированного абонента будет направлен в тот транк, через который осуществлялась его регистрация. В случае неуспешного вызова, перемаршрутизация осуществляться не будет.
При вызове с незарегистрированного абонента сначала идёт проверка, разрешены ли с этого абонентского интерфейса вызовы без регистрации (allow_unreg_call), если проверка успешна, то вызов смаршрутизируется по привязанной таблице маршрутизации и в случае **недоступности транка/ совпадении ответа с маской из списка** произойдёт маршрутизация на следующее направление.

9.9.3 Поведение при перенаправлении

Настройка поведения при перенапрвлении позволяет использовать разные режимы обработки сообщений 3XX.

- forbidden — при получении 3xx ответа вызов завершается;
- transit — 3xx передаётся на другое плечо без изменений контакта;
- process — локальная обработка 3xx ответа.

Пример локальной обработки 3xx ответа

Схема:

Из транка A на ESBC прилетает инициирующий INVITE с номера number_a на номер number_b, этот INVITE пересылается на сторону B, откуда приходит ответ 302 с number_c@IP_c в заголовке Contact.

```
A(транк)---IP_a           |     IP_ESBC_1---ESBC---IP_ESBC_2       |        IP_b---B(транк/
абонентский интерфейс)
                          |                                        |
INVITE number_b@IP_ESBC_1-|-------------->INVITE number_b@IP_b------|---------------------->
From: number_a@IP_a       |               From: number_a@IP_ESBC_2  |
To: number_b@IP_ESBC_1    |               To: number_b@IP_b         |
                          |                                        |
                          |                    <-------------------|------302 Moved Temporarily
                          |                                        |      Contact:
number_c@IP_c
```

1) IP_c == IP_ESBC_2 и существует абонент с username == number_c:

Отправляем INVITE абоненту на тот транк, где он зарегистрирован и 181 в сторону A.

```
A(транк)---IP_a           |     IP_ESBC_1---ESBC---IP_ESBC_3       |        IP_d---D(trunk)
                          |                                        |
                          |           INVITE number_c@IP_d---------|---------------------->
                          |           From: number_a@IP_ESBC_3     |
                          |           To: number_b@IP_b            |
                          |           Diversion: number_b@IP_ESBC_3|
                          |                                        |
     <--------------------|-----------181 Call is Being Forwarded  |
```

2) IP_c == IP_ESBC_2 и абонент не найден:

Заменяем в Contact IP_ESBC_2 на IP_a и пересылаем на другое плечо.

```
A(транк)---IP_a           |     IP_ESBC_1---ESBC---IP_ESBC_2       |        IP_b---B(транк/
абонентский интерфейс)
                          |                                        |
    <---------------------|------302 Moved Temporarily<------------|--------302 Moved
Temporarily
                          |      Contact: number_c@IP_a            |        Contact:
number_c@IP_c
                          |                                        |
```

3) IP_c != IP_ESBC_2 и IP_b – доверенный транк:

Отправляем INVITE на указанный адрес и 181 в сторону A.

```
A(транк)---IP_a          |     IP_ESBC_1---ESBC---IP_ESBC_2       |                    IP_c
                         |                                        |
                         |           INVITE number_c@IP_c         |---------------------->
                         |           From: number_a@IP_ESBC_2     |
                         |           To: number_b@IP_b            |
                         |           Diversion: number_b@IP_ESBC_2|
                         |                                        |
     <--------------------|-----------181 Call is Being Forwarded  |
```

Для того чтобы транк считался доверенным, нужно включить опцию trusted-network в конфигурации транка.

В прочих случаях — вызов завершается.

9.9.4 Игнорирование OPTIONS

Данный режим используется для обработки входящих сообщений OPTIONS.

- ignore options enable — игнорирование запросов OPTIONS. На входящие запросы OPTIONS не будут отправляться ответы;
- no ignore options enable — отключение игнорирования запросов OPTIONS. На входящие запросы OPTIONS будут отправляться ответы 200 OK.

Игнорирование OPTIONS по умолчанию включено.

Если к user-interface привязан sip profile с включенным игнорированием OPTIONS, то при получении OPTIONS от зарегистрированных абонентов ESBC будет обрабатывать эти запросы и отвечать 200 OK (только если в запросе указан заголовок Contact).
Если OPTIONS приходят не с зарегистрированных абонентов, то такие запросы игнорируются.

9.10 Настройка медиапрофилей

Использование медиапрофилей позволяет гибко управлять типом медиаданных путем фильтрации медиасекций в SDP, транскодированием аудио и видео, шифрованием RTP-потока, контролем сессии по наличию RTP-потока.

Медиапрофили используются в абонентских интерфейсах, транках и транк-группах. Медиапрофиль, используемый для транка, входящего в транк-группу, переопределяет настройки медиапрофиля, используемого в транк-группе.

9.10.1 Управление типом медиаданных и кодеками

Обработка медиапотоков осуществляется в двух режимах: проксирование и транскодирование.

По умолчанию ESBC работает в режиме проксирования медиатрафика без использования транскодирования. Список паттернов кодеков, доступных для проксирования через ESBC, задается командой:

```
codec allow {all | <CODEC_PATTERN> [<PT>]}
```

<CODEC_PATTERN> — название кодека/часть названия кодека;

<PT> — payload type (необязательный параметр). При указании будет проводиться дополнительная проверка паттерна на полное совпадение кодека с указанным payload type.

Описание всех команд приведено в разделе Настройки медиапрофиля справочника команд CLI.

При создании медиапрофиля список паттернов для наиболее известных кодеков IANA, доступных для проксирования, добавляется автоматически и выглядит следующим образом:

```
media profile MEDIA_PROFILE
  codec allow AMR
  codec allow CLEARMODE
  codec allow CN
  codec allow G72
  codec allow G722/ 9
  codec allow G728 15
  codec allow G729/ 18
  codec allow GSM 3
  codec allow H26
  codec allow H261 31
  codec allow H263 34
  codec allow ILBC
  codec allow L16/44100 11
  codec allow L16/44100/2 10
  codec allow OPUS
  codec allow PCMA 8
  codec allow PCMU 0
  codec allow SPEEX
  codec allow T38 t38
  codec allow VP
  codec allow telephone-event
exit
```

Т. к. использование паттернов позволяет указывать не полное название кодека, а его часть, то запись вида "codec allow G72" означает, что кодеки G726-16, G726-24, G726-32, G726-40 будут доступны для проксирования.

Для кодеков со статическим payload type рекомендуется указывать номер payload type, иначе, если в SDP не будет указан атрибут rtpmap, вызов будет отбиваться кодом 488.

Для абонентских интерфейсов, транков и транковых групп, к которым не привязан ни один медиапрофиль, используется медиапрофиль по умолчанию, который не отображается в конфигурации. В данном медиапрофиле применяются паттерны кодеков, доступных для проксирования, указанные выше.

Для очистки списка используется команда *no codec allow all*. При использовании данной команды будут удалены паттерны кодеков, добавленные автоматически при создании профиля, и паттерны кодеков, добавленные/измененные пользователем.

Управление списком кодеков и типом медиаданных (audio, video, image) SDP осуществляется путем добавления/удаления/изменения паттернов codec allow. Максимальное количество паттернов в одном медиапрофиле — 64.

Для успешного согласования кодеков в режиме проксирования, необходимо, чтобы на входящем и исходящем направлении в медиапрофилях, привязанным к этим направлениям, содержались паттерны, позволяющие пропускать одни и те же кодеки. В случае когда согласование невозможно, на запросы INVITE ESBC будет отвечать сообщением 488.

Примеры использования медиапрофиля для управления кодеками и типами медиаданных в режиме проксирования

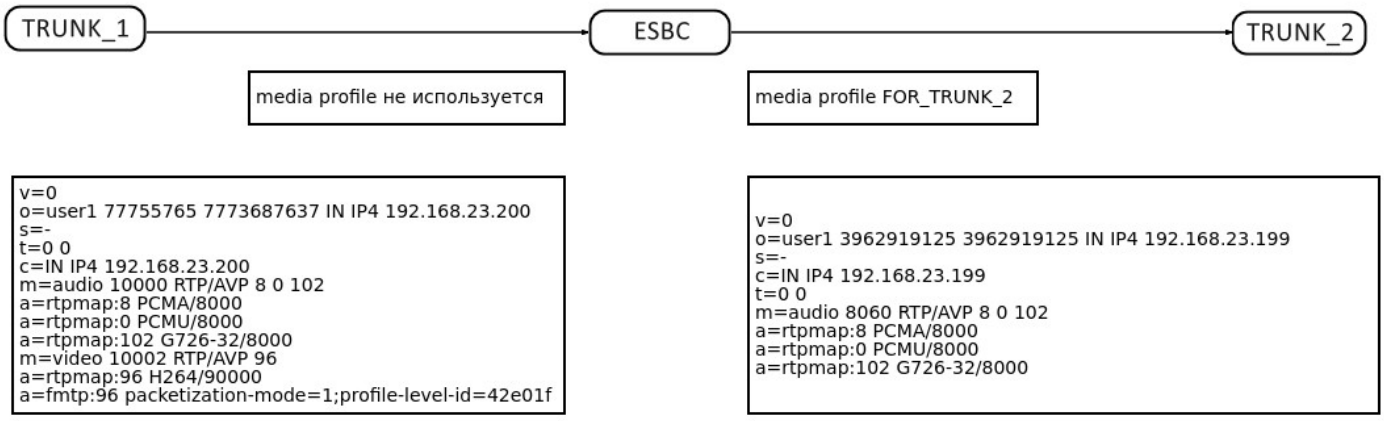
1. Запретить использование видео для транка TRUNK_2.


```
vesbc# configure
vesbc(config)# esbc

#Создать медиапрофиль для транка TRUNK_2:
vesbc(config-esbc)# media profile FOR_TRUNK_2

#Запретить использование всех видеокодеков:
vesbc(config-esbc-media-profile)# no codec allow H26
vesbc(config-esbc-media-profile)# no codec allow H261
vesbc(config-esbc-media-profile)# no codec allow H263
vesbc(config-esbc-media-profile)# no codec allow VP
vesbc(config-esbc-media-profile)# exit

#Привязать медиапрофиль к транку:
vesbc(config-esbc)# trunk sip TRUNK_2
vesbc(config-esbc-trunk-sip)# media profile FOR_TRUNK_2
vesbc(config-esbc-trunk-sip)# do commit
vesbc(config-esbc-trunk-sip)# do confirm
```

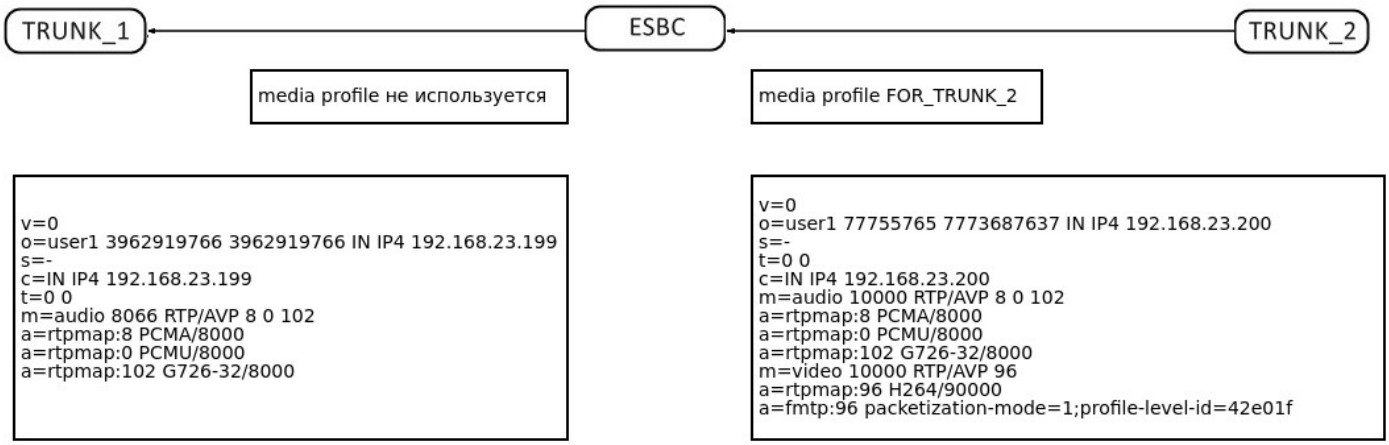
В результате конфигурация медиапрофиля будет выглядеть следующим образом

```
  media profile FOR_TRUNK_2
    codec allow AMR
    codec allow CLEARMODE
    codec allow CN
    codec allow G72
    codec allow G722/ 9
    codec allow G728 15
    codec allow G729/ 18
    codec allow GSM 3
    codec allow ILBC
    codec allow L16/44100 11
    codec allow L16/44100/2 10
    codec allow OPUS
    codec allow PCMA 8
    codec allow PCMU 0
    codec allow SPEEX
    codec allow T38 t38
    codec allow telephone-event
  exit
```

В данном примере для транка TRUNK_1 не требуется использование отдельного медиапрофиля, т. к. при вызовах, поступающих в TRUNK_1, и, маршрутизируемых в TRUNK_2, все видеокодеки из SDP будут удалены в соответствии с медиапрофилем, используемым для транка TRUNK_2.



Для вызовов, поступающих в TRUNK_2, все видеокодеки из SDP будут удалены вне зависимости от направления маршрутизации.

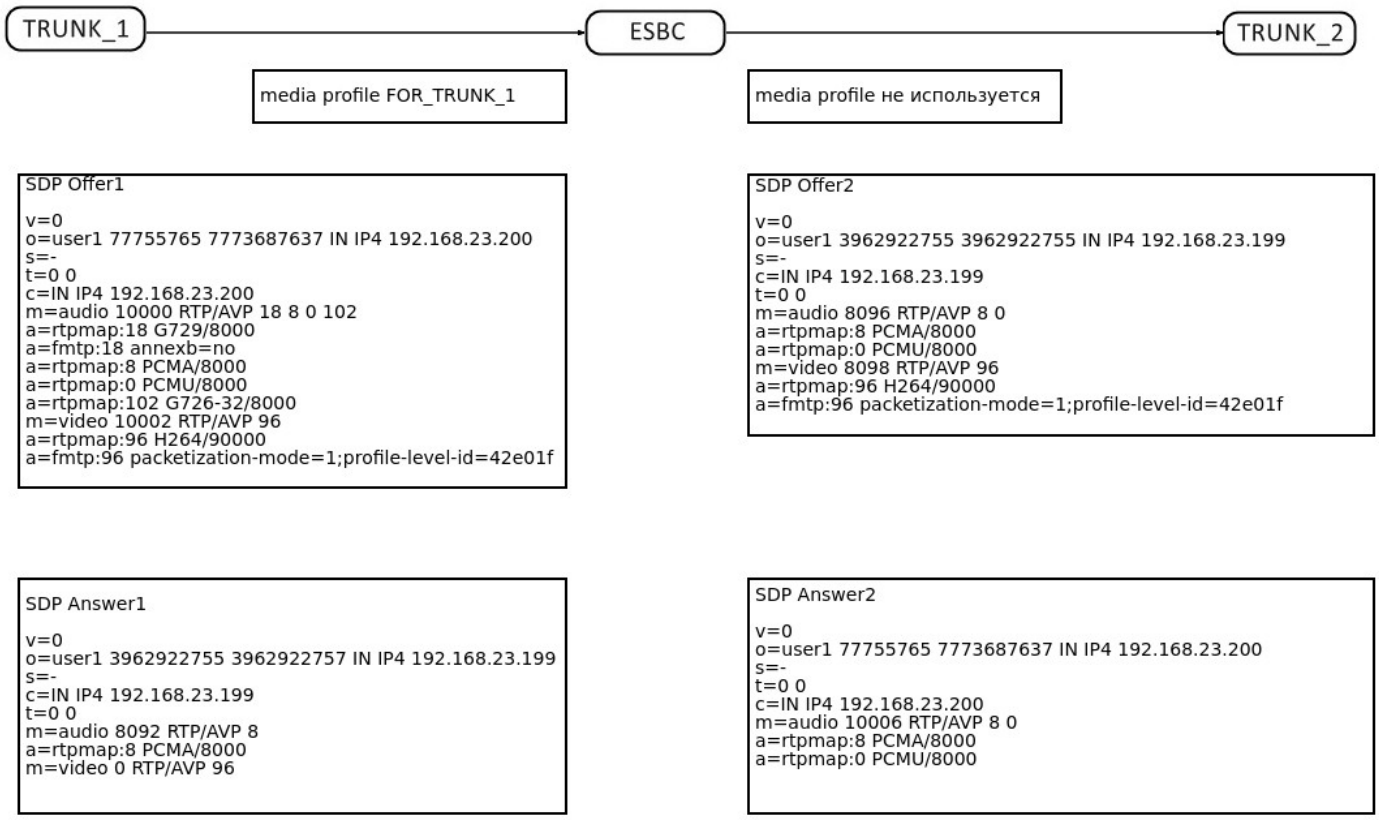


2. Запретить использование кодеков G729 и G726 для транка TRUNK_1.

```
vesbc# configure
vesbc(config)# esbc

#Создать медиапрофиль для транка TRUNK_1:
vesbc(config-esbc)# media profile FOR_TRUNK_1

#Запретить использование кодеков G729 и G726:
vesbc(config-esbc-media-profile)# no codec allow G729/
vesbc(config-esbc-media-profile)# no codec allow G72
vesbc(config-esbc-media-profile)# exit

#Привязать медиапрофиль к транку:
vesbc(config-esbc)# trunk sip TRUNK_1
vesbc(config-esbc-trunk-sip)# media profile FOR_TRUNK_1
vesbc(config-esbc-trunk-sip)# do commit
vesbc(config-esbc-trunk-sip)# do confirm
```

В результате конфигурация медиапрофиля будет выглядеть следующим образом:

```
media profile FOR_TRUNK_1
  codec allow AMR
  codec allow CLEARMODE
  codec allow CN
  codec allow G722/ 9
  codec allow G728 15
  codec allow GSM 3
  codec allow H26
  codec allow H261 31
  codec allow H263 34
  codec allow ILBC
  codec allow L16/44100 11
  codec allow L16/44100/2 10
  codec allow OPUS
  codec allow PCMA 8
  codec allow PCMU 0
  codec allow SPEEX
  codec allow T38 t38
  codec allow VP
  codec allow telephone-event
exit
```



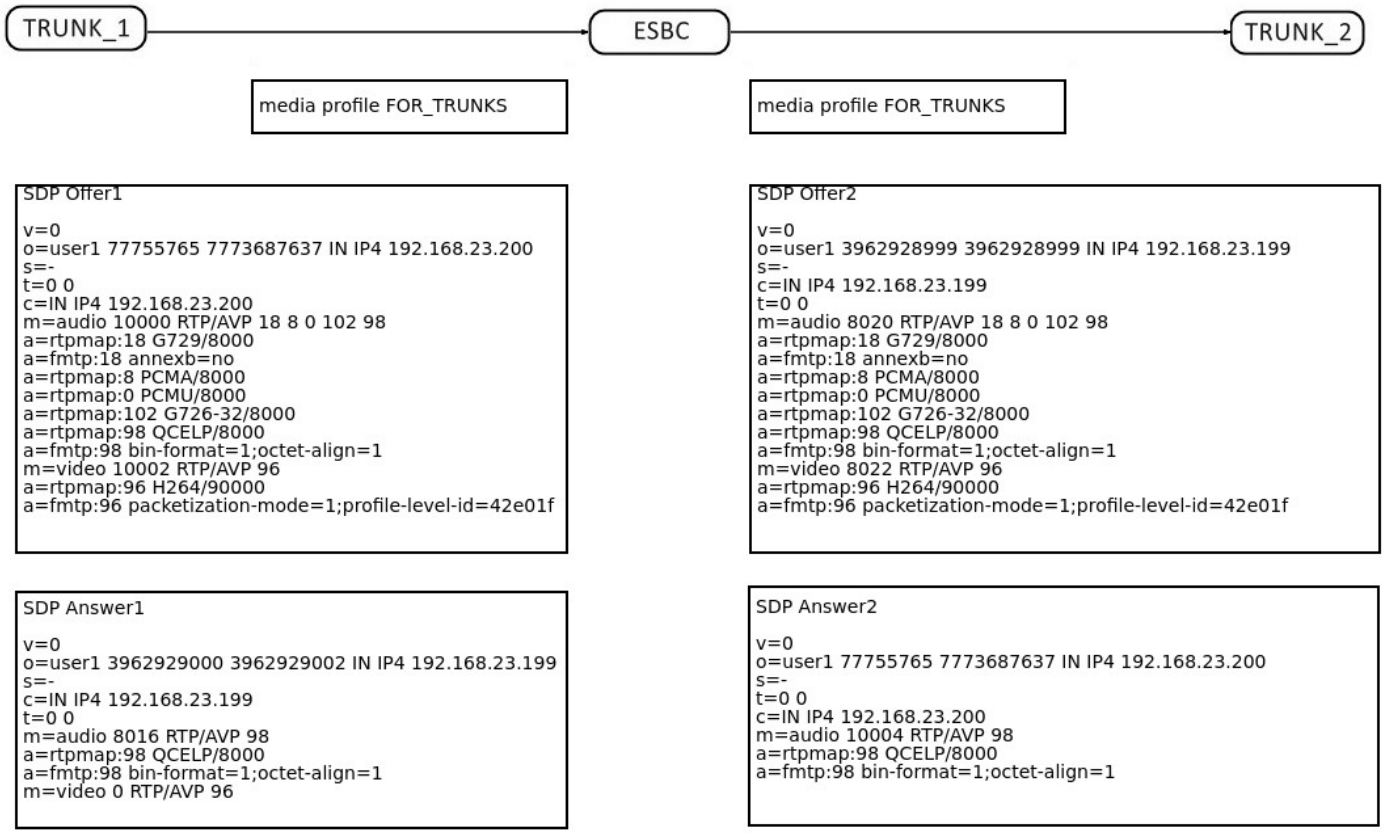
В данном примере в транк TRUNK_1 приходит INVITE с SDP Offer1, в котором наиболее приоритетным кодеком является G729, а также указан кодек G726, но т. к. настройками медиапрофиля FOR_TRUNK_1 данные кодеки запрещены, то в транк TRUNK_2 будет отправлен SDP Offer2 без данных кодеков. UA TRUNK_2 выбирает в качестве приоритетного кодек PCMA (SDP Answer2), и в результате ESBC отправляет в SDP Answer1 наиболее приоритетный кодек из SDP Offer1 (кроме G729) — PCMA.

3. Разрешить использование кодека QCELP для обоих транков (в дополнение к паттернам по умолчанию).

```
vesbc# configure
vesbc(config)# esbc

#Создать медиапрофиль для использования в обоих транках:
vesbc(config-esbc)# media profile FOR_TRUNKS

#Добавить паттерн для кодека QCELP:
vesbc(config-esbc-media-profile)#codec allow QCELP
vesbc(config-esbc-media-profile)# exit

#Привязать медиапрофиль к обоим транкам:
vesbc(config-esbc)# trunk sip TRUNK_1
vesbc(config-esbc-trunk-sip)# media profile FOR_TRUNKS
vesbc(config-esbc-trunk-sip)# exit
vesbc(config-esbc)# trunk sip TRUNK_2
vesbc(config-esbc-trunk-sip)# media profile FOR_TRUNKS
vesbc(config-esbc-trunk-sip)# exit
vesbc(config-esbc)# do commit
vesbc(config-esbc)# do confirm
```

В результате конфигурация медиапрофиля будет выглядеть следующим образом:

```
 media profile FOR_TRUNKS
codec allow AMR
codec allow CLEARMODE
codec allow CN
codec allow G72
codec allow G722/ 9
codec allow G728 15
codec allow G729/ 18
codec allow GSM 3
codec allow H26
codec allow H261 31
codec allow H263 34
codec allow ILBC
codec allow L16/44100 11
codec allow L16/44100/2 10
codec allow OPUS
codec allow PCMA 8
codec allow PCMU 0
codec allow QCELP
codec allow SPEEX
codec allow T38 t38
codec allow VP
codec allow telephone-event
exit
```



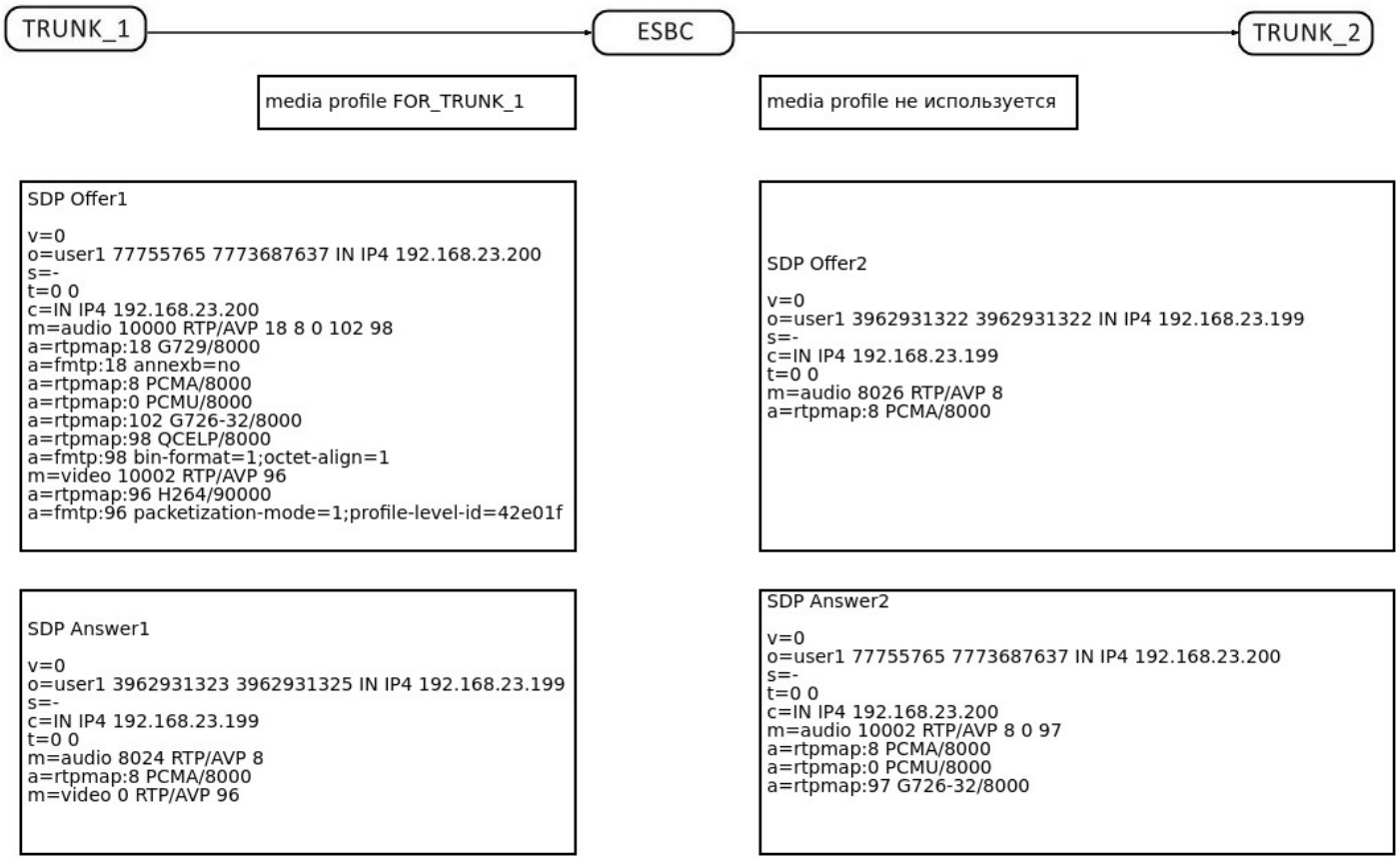
В данном примере в транк TRUNK_1 приходит INVITE с SDP Offer1, в котором содержится кодек QCELP, и т. к. настройками медиапрофиля FOR_TRUNKS этот кодек разрешен, то он будет передаваться SDP Offer2, отправляемый в транк TRUNK_2. UA TRUNK_2 выбирает кодек QCELP, и в результате он будет согласован в SDP Answer1.

4. Разрешить использование только кодека PCMA для TRUNK_1.

```
vesbc# configure
vesbc(config)# esbc

#Создать медиапрофиль для транка TRUNK_1:
vesbc(config-esbc)# media profile FOR_TRUNK_1

#Удалить все паттерны:
vesbc(config-esbc-media-profile)# no codec allow all

#Добавить паттерн только для кодека PCMA:
vesbc(config-esbc-media-profile)# codec allow PCMA 8
vesbc(config-esbc-media-profile)# exit

#Привязать медиапрофиль к транку:
vesbc(config-esbc)# trunk sip TRUNK_1
vesbc(config-esbc-trunk-sip)# media profile FOR_TRUNK_1
vesbc(config-esbc-trunk-sip)# do commit
vesbc(config-esbc-trunk-sip)# do confirm
```

В результате конфигурация медиапрофиля будет выглядеть следующим образом:

```
media profile FOR_TRUNK_1
  codec allow PCMA 8
exit
```



В данном примере в транк TRUNK_1 приходит INVITE с SDP Offer1 с набором кодеков. Т. к. настройками медиапрофиля FOR_TRUNK_1 запрещены все кодеки кроме PCMA, то в транк TRUNK_2 будет отправлен SDP Offer2, содержащий только кодек PCMA.

9.10.2 Транскодирование

Транскодирование — это возможность преобразования медиапотоков, основанных на разных кодеках.

Эта возможность позволяет:

- гибко справляться со сложными сетевыми соединениями и широким спектром медиакодеков;
- оптимизировать доступную полосу пропускания, принудительно используя различные кодеки сжатия;
- нормализовать трафик в сети предприятия, используя один кодек;
- заключать соглашения о взаимодействии между сетями VoIP для использования одобренных кодеков.

ESBC предоставляет возможности транскодирования на границе сети вместо использования ресурсов сети предприятия для тех же функций.

Список кодеков, поддерживаемых в режиме транскодирования:

Аудиокодеки	Видеокодеки
AMR	H263-1998
AMR-WB	H264
G722	VP8
G7221-24	VP9
G7221-32	
G7221C-24	
G7221C-32	
G7221C-48	
G726-16	
G726-24	
G726-32	
G726-40	
G729	
GSM	
ILBC	
L16-MONO	
OPUS	
PCMA	
PCMU	
SPEEX-NB	
SPEEX-UWB	
SPEEX-WB	

Поддержка кодеков для транскодирования осуществляется командами:

```
codec {audio | video | image} {all | <CODEC>}
no codec {audio | video | image} {all | <CODEC>}
```

<CODEC> — название кодека. Указывается из списка поддержанных для транскодирования кодеков.

all — включает транскодирование всех доступных кодеков заданного типа медиаданных.

> ⚠ Команда *codec image* в текущей версии ПО не поддерживается, данная команда аналогична команде *codec allow T38 t38*.

Описание всех команд приведено в разделе Настройки медиапрофиля справочника команд CLI.

Порядок обработки SDP для выбора режима работы:

1. Offer SDP фильтруется согласно разрешённым кодекам на плече A.
2. Offer SDP фильтруется согласно разрешённым кодекам на плече B.
3. Если в медиапрофиле на плече A включен транскодинг, и во входящем SDP присутствуют кодеки из списка разрешенных, то в конец Offer SDP добавляются недостающие кодеки, транскодинг которых включен в media profile на плече B.
4. Answer SDP фильтруется согласно разрешённым кодекам на плече B.
5. В конец Answer SDP добавляются недостающие кодеки, транскодинг которых включен в media profile на плече A.
6. Перед отправкой Answer SDP в плечо A производится согласование кодеков.

В результате, транскодирование включается, если самые приоритетные кодеки из Offer и Answer SDP на двух плечах не совпадают. В таком случае в Answer SDP будет выбран наиболее приоритетный кодек, который был получен в Offer SDP, и для которого включена поддержка транскодирования в медиапрофиле на плече A.

Иначе, при совпадении приоритетных кодеков в Offer SDP и Answer SDP, будет использоваться проксирование медиаданных.

С целью снижения нагрузки на ESBC, транскодирование включается только в случае, когда использовать проксирование медиатрафика невозможно.

Для включения транскодирования необходимо использовать медиапрофили с включенным транскодированием (codec audio/video) на обоих направлениях.

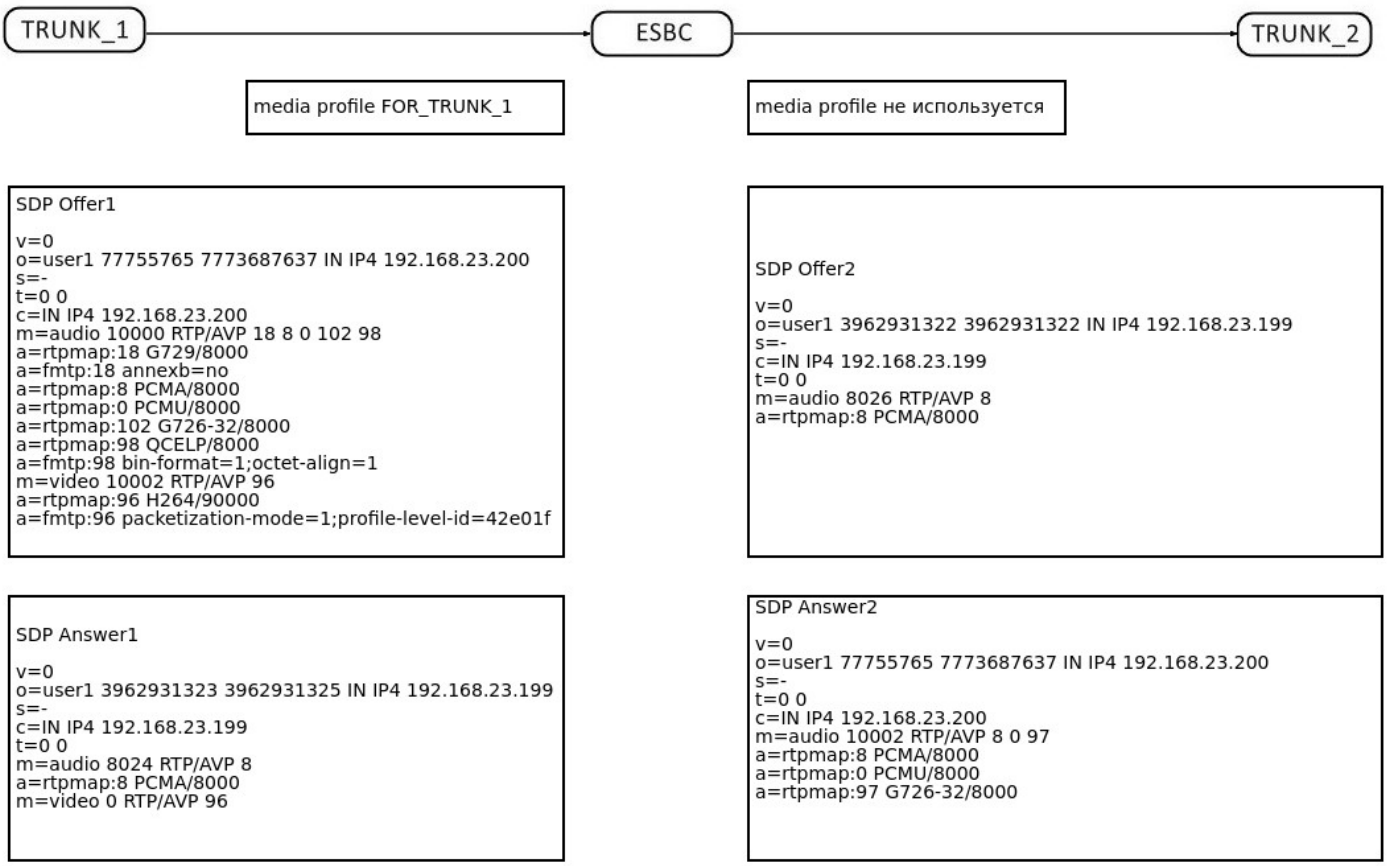
Если на одном из направлений не используется медиапрофиль (т. е. используется медиапрофиль по умолчанию), или в профиле не настроено ни одно правило codec audio/video, то транскодирование осуществляться не будет.

Пример:

В транке TRUNK_1 используется медиапрофиль FOR_TRUNK_1, в котором разрешены кодеки PCMA и PCMU для проксирования, и не указаны кодеки, разрешенные для транскодирования.

В транке TRUNK_2 используется медиапрофиль FOR_TRUNK_2, в котором также кодеки PCMA и PCMU разрешены для проксирования, и кодек G729 разрешен для транскодирования.

В SDP Offer1, полученном с транка TRUNK_1, указаны кодеки PCMA и PCMU, и т. к. в медиапрофиле FOR_TRUNK_1 отсутствуют кодеки, разрешенные для транскодирования, в SDP Offer2, который будет отправлен в TRUNK_2, кодек G729 не будет добавлен. Соответственно при вызовах из TRUNK_1 в TRUNK_2 (и в обратном направлении) возможно только проксирование медиаданных.



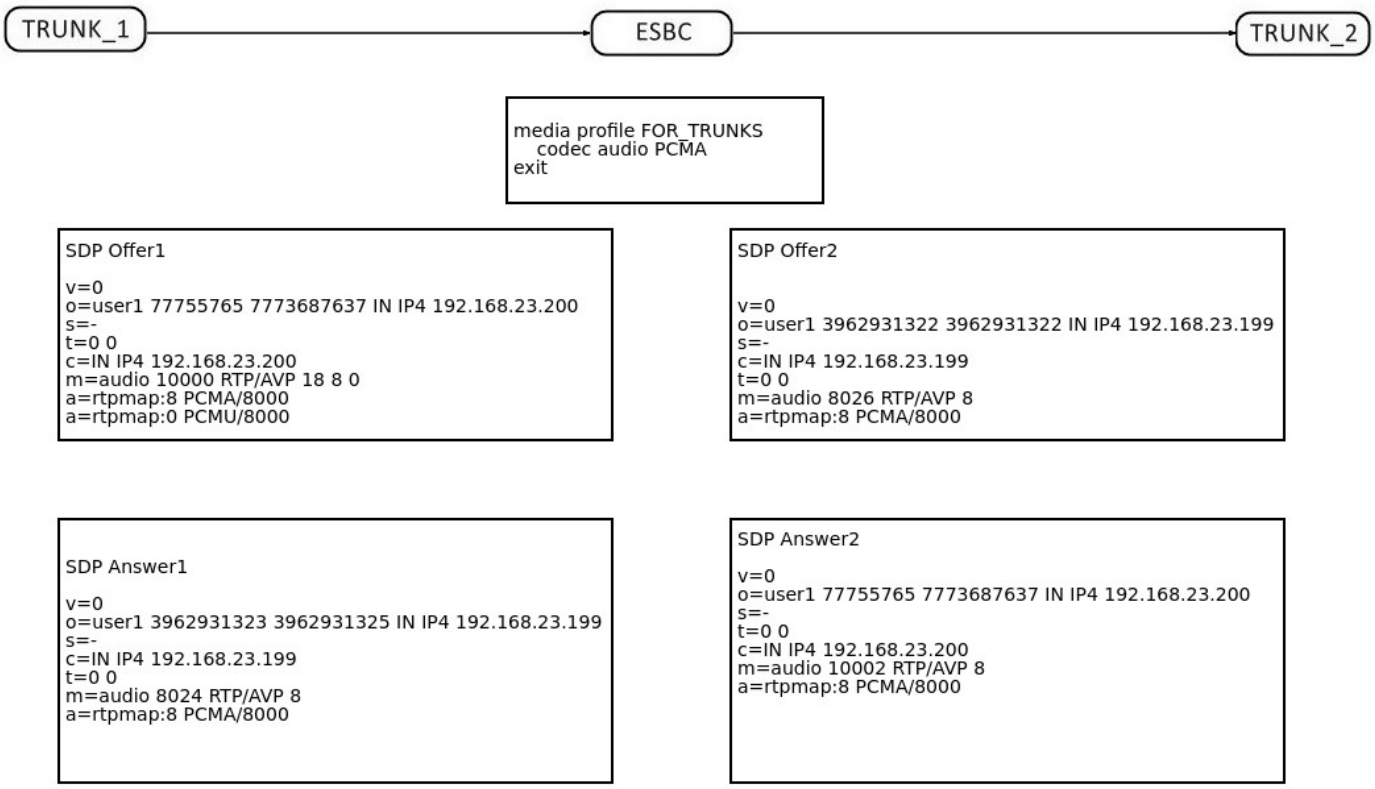
Если в медиапрофиле не содержится ни одного паттерна для проксирования кодеков, а указаны только кодеки, доступные для транскодирования, то при наличии одинаковых кодеков в медиапрофилях, используемых на входящем и исходящем направлениях, медиаданные будут передаваться в режиме проксирования.

Т. о. включение поддержки транскодирования для кодеков командами codec {audio | video | image} {all | <CODEC>} не означает, что передаваемые через ESBC медиаданные всегда будут транскодироваться.

Пример:

Для транков TRUNK_1 и TRUNK_2 используется один и тот же медиапрофиль FOR_TRUNKS, в котором указан только кодек PCMA, разрешенный для транскодирования, и отсутствуют паттерны кодеков для проксирования.

В SDP Offer1, полученном с транка TRUNK_1, указаны кодеки PCMA и PCMU, и т. к. в медиапрофиле FOR_TRUNKS не указан кодек PCMU (ни для проксирования, ни для транскодирования),то в SDP Offer2, который будет отправлен в TRUNK_2, кодек PCMU не будет добавлен. При получении SDP Answer2 происходит согласование кодека PCMA, и в TRUNK_1 будет отправлен SDP Answer1 с кодеком PCMA. При этом медиаданные будут передаваться в режиме проксирования, т. к. наиболее приоритетные кодеки в SDP Offer и SDP Answer совпадают.



Примеры использования медиапрофилей для управления кодеками в режиме транскодирования

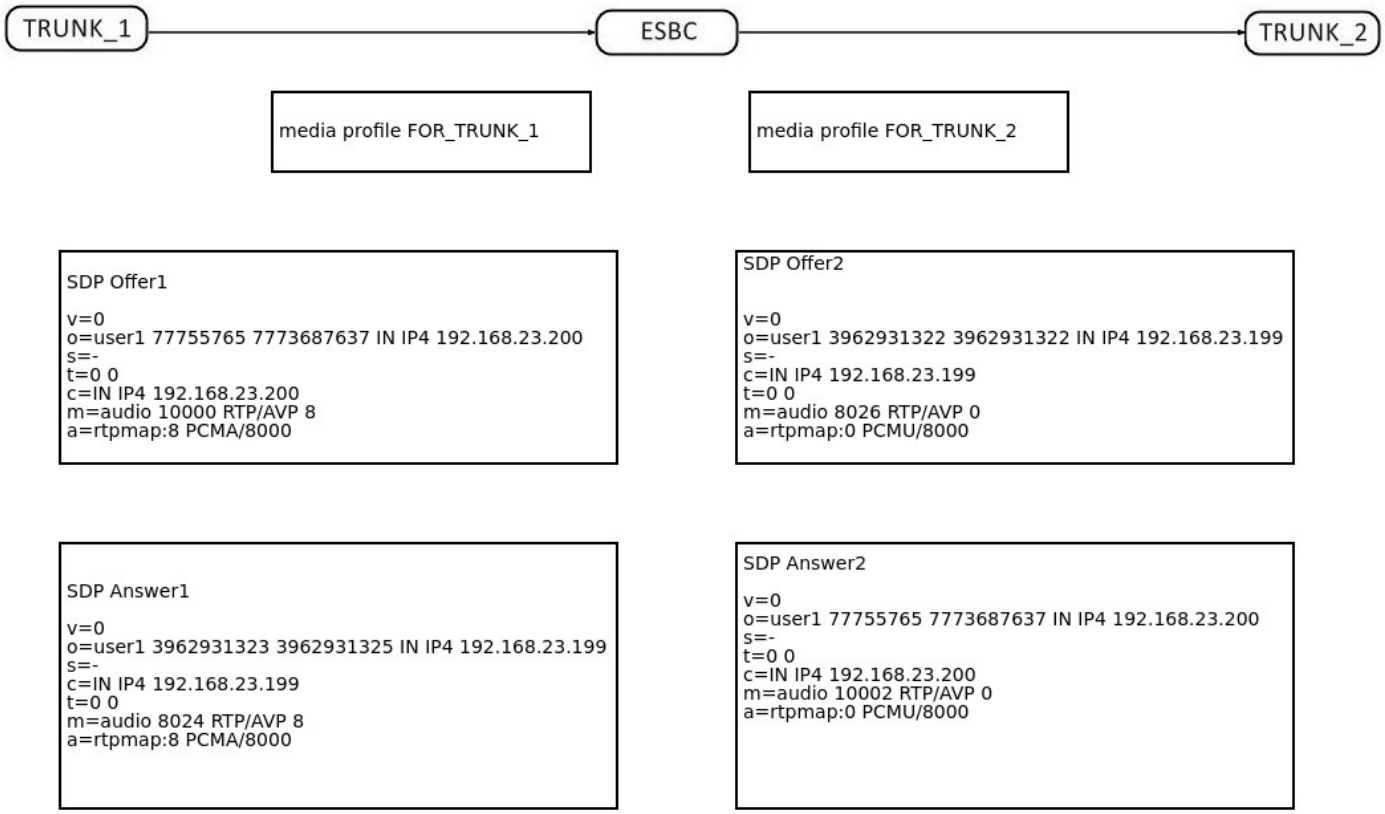
1. Настроить только режим транскодирования кодеков PCMA <---> PCMU между направлениями.

```
vesbc# configure
vesbc(config)# esbc

#Создать медиапрофиль для транка TRUNK_1:
vesbc(config-esbc)# media profile FOR_TRUNK_1

#Запретить использование всех кодеков в режиме проксирования:
vesbc(config-esbc-media-profile)# no codec allow all

#Включить поддержку кодека PCMA в режиме транскодирования:
vesbc(config-esbc-media-profile)# codec audio PCMA
vesbc(config-esbc-media-profile)# exit

#Привязать медиапрофиль к транку TRUNK_1:
vesbc(config-esbc)# trunk sip TRUNK_1
vesbc(config-esbc-trunk-sip)# media profile FOR_TRUNK_1
vesbc(config-esbc-trunk-sip)# exit

#Создать медиапрофиль для транка TRUNK_2:
vesbc(config-esbc)# media profile FOR_TRUNK_2

#Запретить использование всех кодеков в режиме проксирования:
vesbc(config-esbc-media-profile)# no codec allow all

#Включить поддержку кодека PCMU в режиме транскодирования:
vesbc(config-esbc-media-profile)# codec audio PCMU
vesbc(config-esbc-media-profile)# exit

#Привязать медиапрофиль к транку TRUNK_2:
vesbc(config-esbc)# trunk sip TRUNK_2
vesbc(config-esbc-trunk-sip)# media profile FOR_TRUNK_2
vesbc(config-esbc-trunk-sip)# do commit
vesbc(config-esbc-trunk-sip)# do confirm
```

В результате конфигурация медиапрофилей будет выглядеть следующим образом:

```
  media profile FOR_TRUNK_1
    codec audio PCMA
  exit
  media profile FOR_TRUNK_2
    codec audio PCMU
  exit
```

В SDP Offer1, полученном с транка TRUNK_1, указан кодек PCMA, и т. к. в медиапрофиле FOR_TRUNK_2 указан только кодек PCMU для транскодирования, в SDP Offer2, который будет отправлен в TRUNK_2, кодек PCMA будет заменен на PCMU. Соответственно при вызовах из TRUNK_1 в TRUNK_2 (и в обратном направлении) возможно только транскодирование медиаданных.



Если в В SDP Offer1, полученном с транка TRUNK_1, будут указаны любые кодеки кроме PCMA, то вызов не будут установлен, ESBC отправит на INVITE ответ 488.

2. Использование медиапрофилей для проксирования и транскодирования аудиоданных.

Для транков TRUNK_1 и TRUNK_2 используются медиапрофили, позволяющие проксировать все кодеки и транскодировать аудио G722 <---> G729 и GSM <---> G729.

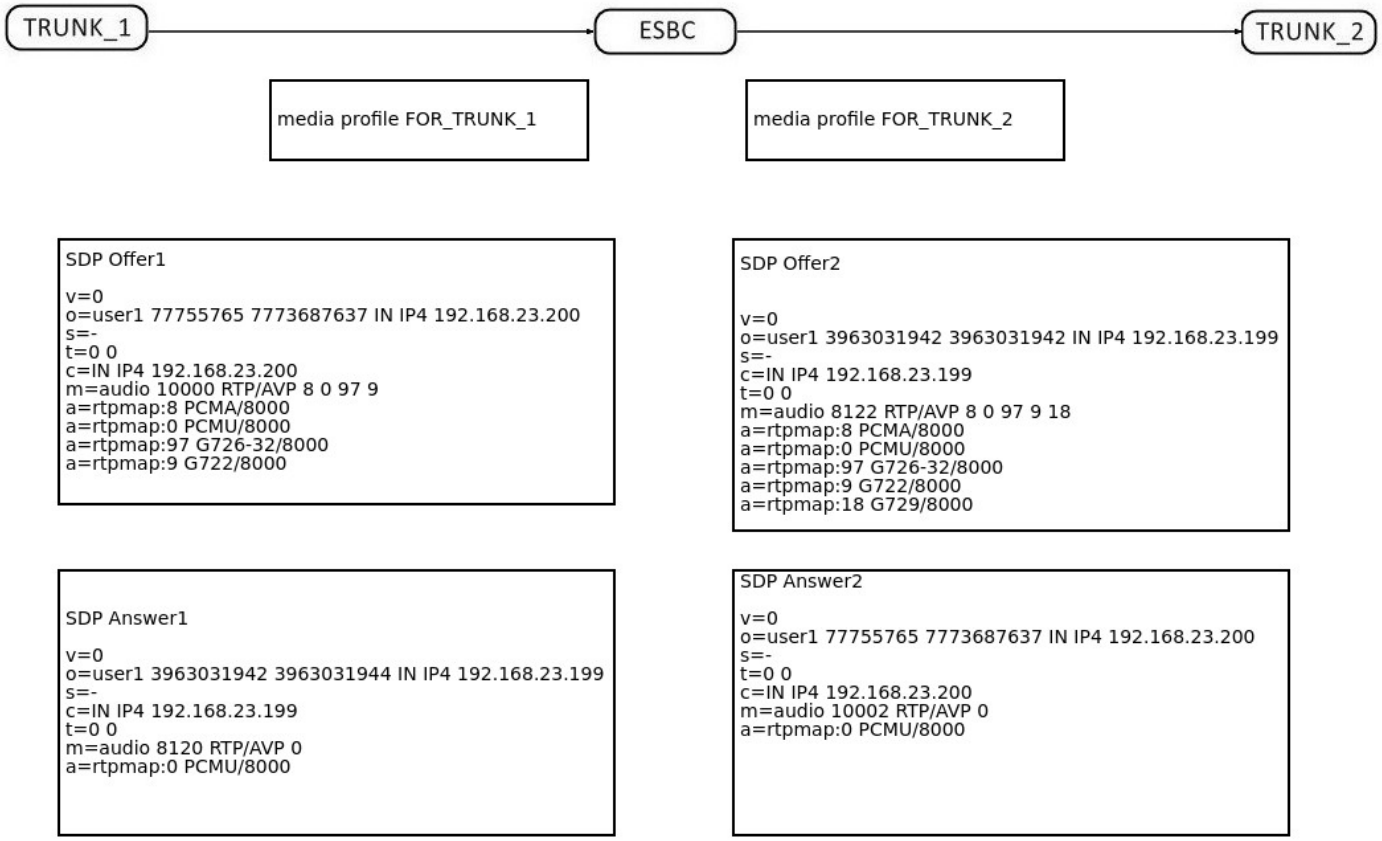
Настройка медиапрофилей:

```
vesbc# configure
vesbc(config)# esbc

#Создать медиапрофиль для транка TRUNK_1:
vesbc(config-esbc)# media profile FOR_TRUNK_1

#Включить поддержку кодеков G722 и GSM в режиме транскодирования:
vesbc(config-esbc-media-profile)# codec audio G722
vesbc(config-esbc-media-profile)# codec audio GSM
vesbc(config-esbc-media-profile)# exit

#Привязать медиапрофиль к транку TRUNK_1:
vesbc(config-esbc)# trunk sip TRUNK_1
vesbc(config-esbc-trunk-sip)# media profile FOR_TRUNK_1
vesbc(config-esbc-trunk-sip)# exit

#Создать медиапрофиль для транка TRUNK_2:
vesbc(config-esbc)# media profile FOR_TRUNK_2

#Включить поддержку кодека G729 в режиме транскодирования:
vesbc(config-esbc-media-profile)# codec audio G729
vesbc(config-esbc-media-profile)# exit

#Привязать медиапрофиль к транку TRUNK_2:
vesbc(config-esbc)# trunk sip TRUNK_2
vesbc(config-esbc-trunk-sip)# media profile FOR_TRUNK_2
vesbc(config-esbc-trunk-sip)# do commit
vesbc(config-esbc-trunk-sip)# do confirm
```

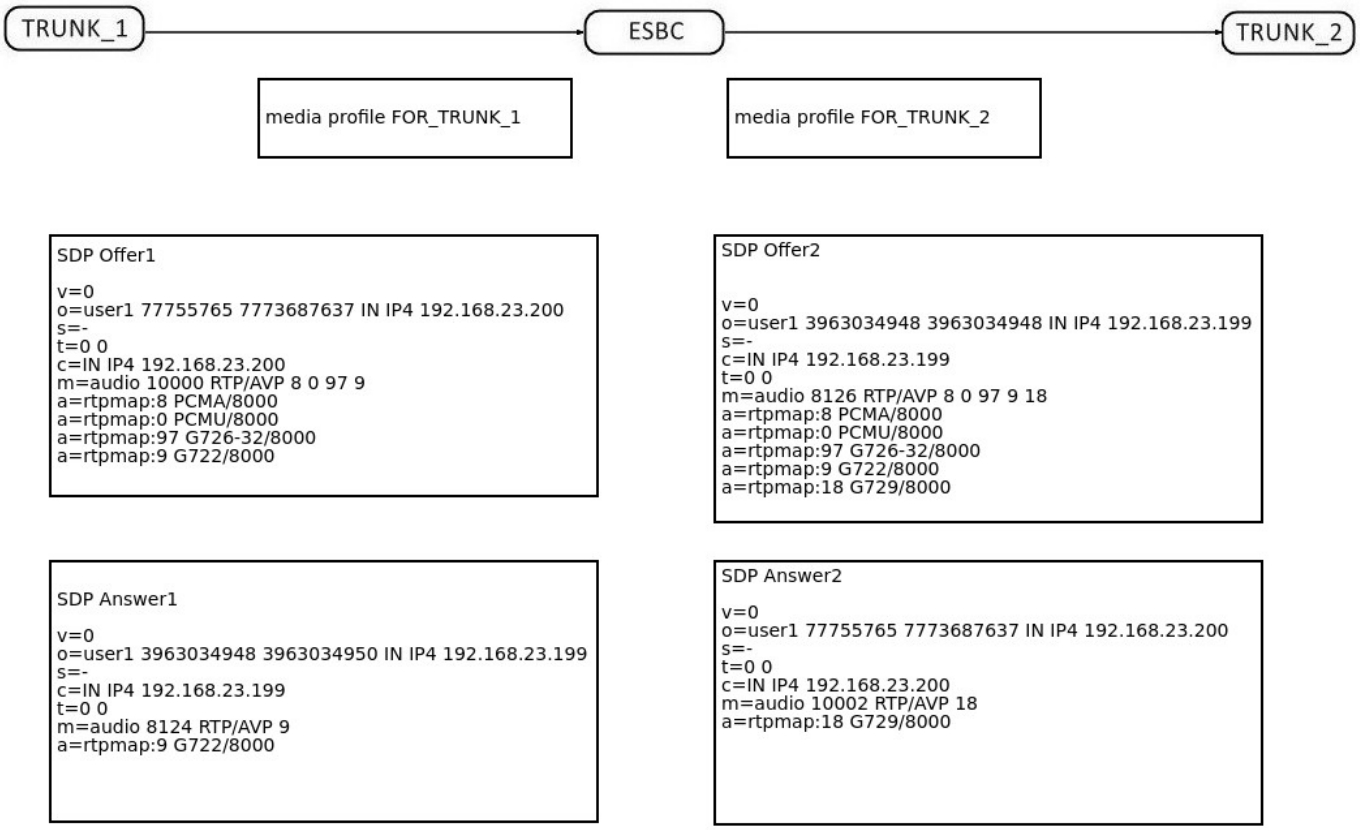
В результате конфигурация медиапрофилей будет выглядеть следующим образом:

```
media profile FOR_TRUNK_1
  codec audio GSM
  codec audio G722
  codec allow AMR
  codec allow CLEARMODE
  codec allow CN
  codec allow G72
  codec allow G722/ 9
  codec allow G728 15
  codec allow G729/ 18
  codec allow GSM 3
  codec allow H26
  codec allow H261 31
  codec allow H263 34
  codec allow ILBC
  codec allow L16/44100 11
  codec allow L16/44100/2 10
  codec allow OPUS
  codec allow PCMA 8
  codec allow PCMU 0
  codec allow SPEEX
  codec allow T38 t38
  codec allow VP
  codec allow telephone-event
exit

media profile FOR_TRUNK_2
  codec audio G729
  codec allow AMR
  codec allow CLEARMODE
  codec allow CN
  codec allow G72
  codec allow G722/ 9
  codec allow G728 15
  codec allow G729/ 18
  codec allow GSM 3
  codec allow H26
  codec allow H261 31
  codec allow H263 34
  codec allow ILBC
  codec allow L16/44100 11
  codec allow L16/44100/2 10
  codec allow OPUS
  codec allow PCMA 8
  codec allow PCMU 0
  codec allow SPEEX
  codec allow T38 t38
  codec allow VP
  codec allow telephone-event
exit
```

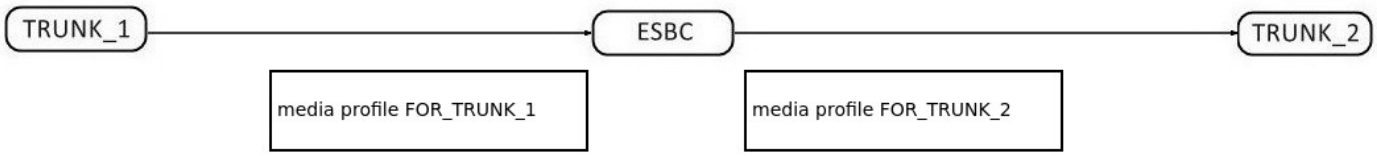
2.1 В SDP Offer1, полученном с транка TRUNK_1, указаны кодеки PCMA, PCMU, G726 и G722. Т. к. в медиапрофиле FOR_TRUNK_1 есть кодек G722, разрешенный для транскодирования, то в SDP Offer2, который будет отправлен в TRUNK_2, будет добавлен кодек G729. Остальные кодеки будут передаваться из SDP Offer1 в SDP Offer2, т. к. на обоих медиапрофилях настроены паттерны, разрешающие проксирование этих кодеков. В SDP Answer2, полученном из TRUNK_2, указан кодек PCMU. Этот кодек будет согласован ESBC в SDP Answer1. Т. к. этот кодек был в SDP Offer1, то будет выбран режим проксирования медиаданных.



2.2 В SDP Offer1, полученном с транка TRUNK_1, указаны кодеки PCMA, PCMU, G726 и G722. Т. к. в медиапрофиле FOR_TRUNK_1 есть кодек G722, разрешенный для транскодирования, то в SDP Offer2, который будет отправлен в TRUNK_2, будет добавлен кодек G729. Остальные кодеки будут передаваться из SDP Offer1 в SDP Offer2, т. к. на обоих медиапрофилях настроены паттерны, разрешающие проксирование этих кодеков. В SDP Answer2, полученном из TRUNK_2, указан кодек G729. Т. к. этого кодека не было в SDP Offer1, то будет согласован единственный возможный кодек для TRUNK_1 — G722. Т. к. кодеки на плечах TRUNK_1 и TRUNK_2 отличаются, будет включено транскодирование медиаданных G722 <---> G729.



2.3 В SDP Offer1, полученном с транка TRUNK_1, указаны кодеки PCMA, PCMU, GSM, G726 и G722. Т. к. в медиапрофиле FOR_TRUNK_1 есть кодеки G722 и GSM, разрешенные для транскодирования, то в SDP Offer2, который будет отправлен в TRUNK_2, будет добавлен кодек G729. Остальные кодеки будут передаваться из SDP Offer1 в SDP Offer2, т. к. на обоих медиапрофилях настроены паттерны, разрешающие проксирование этих кодеков. В SDP Answer2, полученном из TRUNK_2, указан кодек G729. Т. к. этого кодека не было в SDP Offer1, то будет согласован наиболее приоритетный кодек для TRUNK_1 — GSM. Т. к. кодеки на плечах TRUNK_1 и TRUNK_2 отличаются, будет включено транскодирование медиаданных GSM <---> G729.



SDP Offer1

```
v=0
o=user1 77755765 7773687637 IN IP4 192.168.23.200
s=-
t=0 0
c=IN IP4 192.168.23.200
m=audio 10000 RTP/AVP 8 0 3 97 9
a=rtpmap:8 PCMA/8000
a=rtpmap:0 PCMU/8000
a=rtpmap:3 GSM/8000
a=rtpmap:97 G726-32/8000
a=rtpmap:9 G722/8000
```

SDP Offer2

```
v=0
o=user1 3963035354 3963035354 IN IP4 192.168.23.199
s=-
c=IN IP4 192.168.23.199
t=0 0
m=audio 8130 RTP/AVP 8 0 3 97 9 18
a=rtpmap:8 PCMA/8000
a=rtpmap:0 PCMU/8000
a=rtpmap:3 GSM/8000
a=rtpmap:97 G726-32/8000
a=rtpmap:9 G722/8000
a=rtpmap:18 G729/8000
```

SDP Answer1

```
v=0
o=user1 3963035354 3963035356 IN IP4 192.168.23.199
s=-
c=IN IP4 192.168.23.199
t=0 0
m=audio 8128 RTP/AVP 3
a=rtpmap:3 GSM/8000
```

SDP Answer2

```
v=0
o=user1 77755765 7773687637 IN IP4 192.168.23.200
s=-
t=0 0
c=IN IP4 192.168.23.200
m=audio 10004 RTP/AVP 18
a=rtpmap:18 G729/8000
```

9.10.3 Таймаут ожидания RTP-пакетов

Это функция контроля состояния разговора по наличию RTP-трафика от встречного устройства. Контроль осуществляется следующим образом: если в течение заданного времени от встречного устройства не поступает ни одного RTP-пакета, то вызов завершается.
По умолчанию контроль выключен.

```
vesbc#
vesbc# configure
vesbc(config)# esbc

#Создание медиапрофиля NEW_MEDIA_PROFILE:
vesbc(config-esbc)# media profile NEW_MEDIA_PROFILE
vesbc(config-esbc-media-profile)#

#Включение таймера в медиапрофиле:
vesbc(config-esbc-media-profile)# rtp timeout 100
vesbc(config-esbc-media-profile)#

vesbc(config-esbc-media profile)# exit
vesbc(config-esbc)#

#Привязать медиапрофиль к транку NEW_TRUNK:
vesbc(config-esbc)# trunk sip NEW_TRUNK
vesbc(config-esbc-trunk-sip)# media profile NEW_MEDIA_PROFILE
vesbc(config-esbc-trunk-sip)#

#Применить и подтвердить изменения:
vesbc(config-esbc-trunk-sip)# do commit
Configuration has been successfully applied and saved to flash. Commit timer started, changes
will be reverted in 600 seconds.
vesbc(config-esbc-trunk-sip)# do confirm
Configuration has been confirmed. Commit timer canceled.
```

Если после внесения изменений во время вызова с транка NEW_TRUNK в течение 100 секунд не будут приходить RTP-пакеты, то вызов будет принудительно завершён.

9.10.4 SRTP

SRTP (Secure Real-time Transport Protocol) — это расширенная версия протокола RTP с набором защитных механизмов. Протокол был опубликован организацией IETF в стандарте RFC 3711. SRTP обеспечивает конфиденциальность за счет шифрования RTP-нагрузки. Для шифрования медиапотока SRTP стандартизирует использование только единственного шифра — AES, который может использоваться в двух режимах:

- Сегментированный целочисленный счётчик — типичный режим, который осуществляет произвольный доступ к любым блокам, что является существенным для трафика RTP, передающегося в публичных сетях с непредсказуемым уровнем надежности и возможной потерей пакетов. Но стандарт для шифрования данных RTP — только обычное целочисленное значение счётчика. AES, работающий в этом режиме, является алгоритмом шифрования по умолчанию, с длиной шифровального ключа в 128 бит и ключом сессии длиной в 112 бит.
- f8-режим — вариант режима способа обратной связи, расширенного, чтобы быть доступным с изменённой функцией инициализации. Значения по умолчанию для шифровального ключа и ключа сессии — то же, что и в AES в режиме, описанном выше.

SRTP использует функцию формирования ключа для создания ключей на основе мастер-ключа. Протокол управления ключами создает все ключи в сессии с помощью мастер-ключа. За счет того, что у каждой сессии свой уникальный ключ, все сессии защищены. Поэтому, если одна сессия была скомпрометирована, то остальные по-прежнему под защитой.

В конфигурации доступны 2 метода обмена ключами:

- DTLS-SRTP (RFC 5763)
- SDES (RFC 4568)

и 3 режима использования SRTP:

- disable — SRTP запрещён;
- optional — SRTP не обязателен, но ключи будут подставлены в offer SDP второго плеча без изменения профиля транспорта в медиасекции SDP;
- mandatory — SRTP обязателен, профиль транспорта в медиасекции SDP будет изменён на соответствующий профиль SRTP.

Если выбран режим mandatory и включены оба метода, то на втором плече будет выбран DTLS-SRTP, как более приоритетный.

⚠ По умолчанию поддержка SRTP выключена.

По умолчанию при использовании DTLS-SRTP используются сертификаты, сгенерированные автоматически. Для использования сертификатов, загруженных пользователем, необходимо в медиапрофиле указать криптопрофиль с необходимыми сертификатами командой *crypto profile*. Подробное описание криптопрофилей приведено в разделе Настройка криптопрофилей.

Пример использования SRTP

Схема:

В конфигурации есть два транка, настроена маршрутизация. Вызов, который приходит из TRUNK_IN, уходит в TRUNK_OUT. На TRUNK_OUT включаем обязательное использование SRTP с методом обмена ключами — SDES.

```
vesbc#
vesbc# configure
vesbc(config)# esbc

#Создание медиапрофиля NEW_MEDIA_PROFILE:
vesbc(config-esbc)# media profile NEW_MEDIA_PROFILE
vesbc(config-esbc-media profile)#

#Настройка SRTP (включили обязательный режим использования, метод обмена ключами – SDES):
vesbc(config-esbc-media-profile)# srtp keying
  dtls-srtp  Enable DTLS-SRTP keying method
  sdes       Enable SDES keying method

vesbc(config-esbc-media-profile)# srtp keying sdes
vesbc(config-esbc-media-profile)# srtp mode
  disable    SRTP is disabled
  mandatory  SRTP is mandatory
  optional   SRTP is optional

vesbc(config-esbc-media-profile)# srtp mode mandatory
vesbc(config-esbc-media-profile)#


vesbc(config-esbc-media-profile)# exit
vesbc(config-esbc)#

#Привязать медиапрофиль к транку TRUNK_OUT:
vesbc(config-esbc)# trunk sip TRUNK_OUT
vesbc(config-esbc-trunk-sip)# media profile NEW_MEDIA_PROFILE
vesbc(config-esbc-trunk-sip)#

#Применить и подтвердить изменения:
vesbc(config-esbc-trunk-sip)# do commit
Configuration has been successfully applied and saved to flash. Commit timer started, changes
will be reverted in 600 seconds.
vesbc(config-esbc-trunk-sip)# do confirm
Configuration has been confirmed. Commit timer canceled.
```

С TRUNK_IN приходит INVITE с SDP offer:

```
Session Description Protocol
    Session Description Protocol Version (v): 0
    Owner/Creator, Session Id (o): 100 61 74 IN IP4 10.25.72.54
    Session Name (s): Talk
    Connection Information (c): IN IP4 10.25.72.54
    Time Description, active time (t): 0 0
    Session Attribute (a): rtcp-xr:rcvr-rtt=all:10000 stat-summary=loss,dup,jitt,TTL voip-
metrics
    Session Attribute (a): record:off
    Media Description, name and address (m): audio 7078 RTP/AVP 96 97 98 0 8 18 101 99 100
    Media Attribute (a): rtpmap:96 opus/48000/2
    Media Attribute (a): fmtp:96 useinbandfec=1
    Media Attribute (a): rtpmap:97 speex/16000
    Media Attribute (a): fmtp:97 vbr=on
    Media Attribute (a): rtpmap:98 speex/8000
    Media Attribute (a): fmtp:98 vbr=on
    Media Attribute (a): fmtp:18 annexb=yes
    Media Attribute (a): rtpmap:101 telephone-event/48000
    Media Attribute (a): rtpmap:99 telephone-event/16000
    Media Attribute (a): rtpmap:100 telephone-event/8000
    Media Attribute (a): rtcp-fb:* trr-int 5000
    Media Attribute (a): rtcp-fb:* ccm tmmbr
    [Generated Call-ID: l0XaoKkqav]
```

На второе плечо (TRUNK_OUT) пересылаем SDP offer с ключами:

```
Session Description Protocol
    Session Description Protocol Version (v): 0
    Owner/Creator, Session Id (o): 100 3932018917 3932018917 IN IP4 192.168.23.199
    Session Name (s): Talk
    Connection Information (c): IN IP4 192.168.23.199
    Time Description, active time (t): 0 0
    Session Attribute (a): rtcp-xr:rcvr-rtt=all:10000 stat-summary=loss,dup,jitt,TTL voip-
metrics
    Session Attribute (a): record:off
    Media Description, name and address (m): audio 8064 RTP/SAVP 96 97 98 0 8 18 101 99 100
    Media Attribute (a): rtpmap:96 opus/48000/2
    Media Attribute (a): fmtp:96 useinbandfec=1
    Media Attribute (a): rtpmap:97 speex/16000
    Media Attribute (a): fmtp:97 vbr=on
    Media Attribute (a): rtpmap:98 speex/8000
    Media Attribute (a): fmtp:98 vbr=on
    Media Attribute (a): fmtp:18 annexb=yes
    Media Attribute (a): rtpmap:101 telephone-event/48000
    Media Attribute (a): rtpmap:99 telephone-event/16000
    Media Attribute (a): rtpmap:100 telephone-event/8000
    Media Attribute (a): rtcp-fb:* trr-int 5000
    Media Attribute (a): rtcp-fb:* ccm tmmbr
    Media Attribute (a): crypto:1 AES_256_CM_HMAC_SHA1_80
inline:FGdOolKfBlrQzUIedHcIqs9uauWEnUbqxXpop9PaI1dPIHVnO/vdb7JJHRLBLw==
    Media Attribute (a): crypto:2 AES_256_CM_HMAC_SHA1_32
inline:Galc9UfOqBFNmr3ICc3Fiuc3HgEXlj+p1dRw85LavzjWR1sGZUr1nsLQjfaTQA==
    Media Attribute (a): crypto:3 AES_CM_128_HMAC_SHA1_80 inline:jEjWFKpqdf6d94g/
ddSjj1i08dEWQA1tTI75Hqx3
    Media Attribute (a): crypto:4 AES_CM_128_HMAC_SHA1_32 inline:uFYI2UDA/
+woJJY4fWljfoxRROffXNtEO81bBnHJ
    [Generated Call-ID: 503d40e930910767a2dd95f88b483189]
```

9.10.5 Контроль источника RTP

Контроль источника RTP позволяет принимать медиатрафик только с IP-адреса и порта, указанного в SDP встречной стороны, повышая безопасность взаимодействия при использовании публичной сети передачи данных.

Включение/выключение режима осуществляется командами *rtp source-verification* и *no rtp source-verification* соответственно. При отключенной проверке IP-адрес и порт источника RTP не проверяется.

По умолчанию опция включена.

⚠ При использовании опции "Контроль источника RTP" совместно с включенной опцией "nat comedia" в транке или абонентском интерфейсе, RTP-трафик будет передаваться на IP-адрес и порт из входящего потока.

9.11 Настройка профилей безопасности

Профили безопасности используются для управления механизмом защиты от SIP-атак. Использование профилей безопасности позволяет гибко управлять уровнями защиты для каждого направления.

Необходимый уровень защиты обеспечивается следующими настройками профиля:

- фильтрация SIP-флуда;
- блокировка по AOR/User-Agent;
- объединение ошибок по IP-адресу.

Профиль безопасности может использоваться в транках, транк-группах и абонентских интерфейсах.

Для обеспечения единой политики безопасности может быть использован один профиль для всех транков и один профиль для всех абонентских интерфейсов. Это может быть как один и тот же профиль безопасности, так и разные. Эти профили указываются в общих настройках ESBC.

> Если для транка и транковой группы, в которую входит этот транк, используются разные профили безопасности, то будет применяться профиль, указанный в настройках транка. Если для транка/абонентского интерфейса и в общих настройках (для всех транков/абонентских интерфейсов) используются разные профили безопасности, то будет применяться профиль, указанный в настройках транка/абонентского интерфейса.

Описание всех команд для настройки профилей безопасности приведено в разделе Настройки профиля безопасности.

9.11.1 Общий принцип работы модуля fail2ban

Модуль занимается анализом возникающих ошибок для дальнейшей блокировки источников "подозрительного SIP-трафика". При возникновении ошибки в модуль отправляется информация о типе ошибки и об источнике. При накоплении достаточного количества ошибок источник блокируется.

Виды ошибок:

- ошибка регистрации;
- ошибка вызова;
- ошибка подписки;
- флуд SIP-пакетов;
- некорректный SIP-пакет;
- срабатывание флуд-фильтра;
- получение пакета вне транка/абонентского интерфейса.

Лимит количества ошибок зависит от нескольких факторов:

- вес ошибки;
- интервал времени между ошибками;
- количество ошибок одного вида;
- количество AOR, которые использовались в SIP-сообщениях с одного IP-адреса;
- количество IP-адресов, которые присылали SIP-сообщения с одним AOR.

При добавлении адреса в чёрный список указывается причина блокировки. Чёрный список можно просмотреть в CLI командой *show esbc black-list* или в WEB на странице Мониторинг → Списки доступа → Чёрный список.

Причины блокировки:

- ACCOUNT HACKING — превышен лимит по количеству ошибок с одинаковым AOR/User-Agent;
- PACKET FLOODING — превышен лимит по количеству ошибок с одного IP-адреса;
- BURST ERRORS — превышен глобальный лимит по количеству ошибок в секунду;

- GLOBAL RPS LIMIT — превышен глобальный лимит по количеству заблокированных запросов в секунду;
- IP RPS LIMIT — превышен лимит по количеству заблокированных запросов в секунду с одного IP-адреса;
- MONITORED ADDRESSES LIMIT — превышено максимальное количество IP-адресов с ошибками;
- DISTRIBUTED SPAM — превышено максимальное количество IP-адресов с одинаковым заблокированным атрибутом (AOR, User-Agent);
- BLOCKED ATTRIBUTES LIMIT — превышено максимальное количество заблокированных атрибутов (AOR, User-Agent);
- IP BLOCKED ATTRIBUTES LIMIT — превышено максимальное количество заблокированных атрибутов (AOR, User-Agent) с одного IP-адреса.

9.11.2 Фильтрация SIP-флуда

ESBC поддерживает создание флуд-фильтров для механизма конфигурируемой защиты от SIP-flood, а также для фильтрации клиентских приложений. Фильтр применяется ко всему SIP-сообщению (включая тело — SDP, XML и т. д.).

В настройках фильтра можно указать до 64 масок/паттернов, по которым происходит поиск. В случае нахождения сообщение определяется как флуд и отбрасывается.

К профилю безопасности можно привязать до 8 флуд-фильтров.

При создании паттерна можно использовать регулярные выражения PCRE.

Пример настройки флуд-фильтров

```
#Создание абонентских интерфейсов:
vesbc# configure
vesbc(config)# esbc
vesbc(config-esbc)# user-interface sip UI_1
vesbc(config-esbc-user-interface-sip)# sip transport TRANSPORT_UI_1
vesbc(config-esbc-user-interface-sip)# media resource 0 MEDIA_UI_1
vesbc(config-esbc-user-interface-sip)# route-table TO_SSW
vesbc(config-esbc-user-interface-sip)# exit
vesbc(config-esbc)# user-interface sip UI_2
vesbc(config-esbc-user-interface-sip)# sip transport TRANSPORT_UI_2
vesbc(config-esbc-user-interface-sip)# media resource 0 MEDIA_UI_2
vesbc(config-esbc-user-interface-sip)# route-table TO_SSW
vesbc(config-esbc-user-interface-sip)# exit
vesbc(config-esbc)# user-interface sip UI_3
vesbc(config-esbc-user-interface-sip)# sip transport TRANSPORT_UI_3
vesbc(config-esbc-user-interface-sip)# media resource 0 MEDIA_UI_3
vesbc(config-esbc-user-interface-sip)# route-table TO_SSW
vesbc(config-esbc-user-interface-sip)# exit

#Создание флуд-фильтра:
vesbc# configure
vesbc(config)# esbc
vesbc(config-esbc)# flood filter FLOOD_FILTER
vesbc(config-esbc-flood-filter)# pattern 0 7543546
vesbc(config-esbc-flood-filter)# pattern 1 flood
vesbc(config-esbc-flood-filter)# exit

#Привязка флуд-фильтра к профилю безопасности:
vesbc(config-esbc)# security profile SECURITY_PROFILE
vesbc(config-esbc-security-profile)# flood filter 0 FLOOD_FILTER
vesbc(config-esbc-security-profile)# exit

#Привязка профиля безопасности с флуд-фильтром ко всем абонентским интерфейсам:
vesbc(config-esbc)# general
vesbc(config-esbc-general)# security profile user-interface SECURITY_PROFILE

#Применение и подтверждение изменений:
vesbc(config-esbc-general)# do commit
Configuration has been successfully applied and saved to flash. Commit timer started, changes
will be reverted in 600 seconds.
vesbc(config-esbc-general)# do confirm
Configuration has been confirmed. Commit timer canceled.
```

После применения изменений все входящие инициирующие запросы с абонентских интерфейсов UI_1, UI_2, UI_3 будут проверяться на наличие подстрок "7543546" и "flood". Если хотя бы одна подстрока будет найдена в сообщении, то оно отбросится, а в модуль fail2ban отправится уведомление о срабатывании флуд-фильтра, при накоплении достаточного количества ошибок источник блокируется.

Для того чтобы фильтр применялся ко всем сообщениям, а не только к инициирующим запросам, необходимо включить опцию apply-to-created.

Фильтрация клиентских приложений

При помощи флуд-фильтров можно реализовать фильтрацию клиентских приложений, реализуется это добавлением паттерна следующего вида:

```
User-Agent:.*потенциально вредоносное клиентское приложение.*
```

В конфигурации флуд-фильтра есть команда, которая автоматически добавляет в конфигурацию фильтра 18 паттернов с часто используемыми вредоносными клиентскими приложениями.

Соответствие запрещенных User-Agent и создаваемым дефолтным паттерном представлено в таблице ниже.

Запрещенный User-Agent	**Создаваемый дефолтный паттерн**
scan	User-Agent:.*scan.*
crack	User-Agent:.*crack.*
flood	User-Agent:.*flood.*
kill	User-Agent:.*kill.*
sipcli	User-Agent:.*sipcli.*
sipv sipvicious	User-Agent:.*sipv.*
sipsak	User-Agent:.*sipsak.*
sundayddr	User-Agent:.*sundayaddr.*
iWar	User-Agent:.*iWar.*
SIVuS	User-Agent:.*SIVuS.*
Gulp	User-Agent:.*Gulp.*
smap	User-Agent:.*smap.*
friendly-request	User-Agent:.*friendly-request.*
VaxIPUserAgent VaxSIPUserAgent	User-Agent:.*VaxS{0,1}IPUserAgent.*
siparmyknife	User-Agent:.*siparmyknife.*
Test Agent	User-Agent:.*Test Agent.*
SIPBomber	User-Agent:.*SIPBomber.*

Запрещенный User-Agent	Создаваемый дефолтный паттерн
Siprogue	User-Agent:.*Siprogue.*

Если в настройках фильтра недостаточно незаполненных паттернов для создания всех default patterns, то они не создадутся.

9.11.3 Блокировка по AOR/User-Agent

По умолчанию при превышении лимита по ошибкам блокируется только адрес источника вредоносного трафика, в конфигурации профиля безопасности можно включить блокировку по AOR из заголовка From и блокировку по значению заголовка User-Agent.

При использовании одного и того же атрибута (AOR, User-Agent, IP-адрес) источником вредоносного трафика раньше всего сработает блокировка по AOR, потом по IP-адресу, потом по User-Agent. Из-за этого при ошибках с одинаковым AOR и IP-адресом может оказаться заблокированным только AOR или при ошибках с одинаковым IP и User-Agent может оказаться заблокированным только IP-адрес.

Пример включения блокировки по AOR:

```
vesbc#
vesbc# configure
vesbc(config)# esbc
vesbc(config-esbc)# security profile SECURITY_PROFILE

#Сбор ошибок по AOR:
vesbc(config-esbc-security-profile)# check aor
vesbc(config-esbc-security-profile)# exit

#Привязка профиля безопасности к абонентскому интерфейсу:
vesbc(config-esbc)# user-interface sip ABONENTS
vesbc(config-esbc-user-interface-sip)# security profile SECURITY_PROFILE

#Применение и подтверждение изменений:
vesbc(config-esbc-user-interface-sip)# do commit
Configuration has been successfully applied and saved to flash. Commit timer started, changes
will be reverted in 600 seconds.
vesbc(config-esbc-user-interface-sip)# do confirm
Configuration has been confirmed. Commit timer canceled.
```

Пример блокировки:

С адреса 192.168.80.134 через абонентский интерфейс приходят сообщения INVITE с AOR 123@anonymous.invalid, вызовы без регистрации на интерфейсе запрещены, поэтому эти запросы отбиваются 403 Forbidden.

Если в профиле безопасности, привязанному к абонентскому интерфейсу, отключена блокировка по AOR, то через определенное количество запросов заблокируется только IP-адрес, и запросы с него больше обрабатываться не будут.

```
IP black-list:
-------------------------------------------------------------------------------
IP address       Ban reason       AOR    AOR        Blocking    Time of blocking
                                         error      timeout
                                         count      in minutes
--------------- ----------------  -----  ---------  ----------  -------------------
192.168.80.134  PACKET FLOODING          0          1440        2025-07-30 11:38:44
```

Если к абонентскому интерфейсу привязан профиль безопасности с включенной блокировкой по AOR, то через какое-то время в чёрный список добавится AOR.
Все запросы с любого адреса, в котором будет заблокированный AOR во From, обрабатываться не будут.

```
AOR black-list:
-------------------------------------------------------------------------------
AOR                      Ban reason        AOR        Forgive   Time of blocking
                                           error      time in
                                           count      minutes
----------------------   ----------------  ---------  --------  -------------------
123@anonymous.invalid    ACCOUNT HACKING   81         60        2025-07-30 11:49:41
```

Блокировка по AOR срабатывает раньше, чем блокировка по адресу, поэтому адрес не успеет попасть в черный список.

Пример включения блокировки по User-Agent:

```
vesbc#
vesbc# configure
vesbc(config)# esbc
vesbc(config-esbc)# security profile SECURITY_PROFILE

#Сбор ошибок по User-Agent:
vesbc(config-esbc-security-profile)# check user-agent
vesbc(config-esbc-security-profile)# exit

#Привязка профиля безопасности к абонентскому интерфейсу:
vesbc(config-esbc)# user-interface sip ABONENTS
vesbc(config-esbc-user-interface-sip)# security profile SECURITY_PROFILE

#Применение и подтверждение изменений:
vesbc(config-esbc-user-interface-sip)# do commit
Configuration has been successfully applied and saved to flash. Commit timer started, changes
will be reverted in 600 seconds.
vesbc(config-esbc-user-interface-sip)# do confirm
Configuration has been confirmed. Commit timer canceled.
```

Пример блокировки:

С нескольких адресов 192.168.80.13x через абонентский интерфейс приходят сообщения INVITE с User-Agent: sipflood, вызовы без регистрации на интерфейсе запрещены, поэтому эти запросы отбиваются 403 Forbidden.

Если в профиле безопасности, привязанному к абонентскому интерфейсу, отключена блокировка по User-Agent, то через определенное количество запросов заблокируются только IP-адреса, и запросы с них больше обрабатываться не будут:

```
IP black-list:
-------------------------------------------------------------------------------
IP address       Ban reason        AOR    AOR      Blocking     Time of blocking
                                          error    timeout
                                          count    in minutes
---------------  ----------------  ----   -----    ----------   -------------------

192.168.80.132   PACKET FLOODING           0        1440         2025-07-31 04:51:53

192.168.80.133   PACKET FLOODING           0        1440         2025-07-31 04:52:22

192.168.80.134   PACKET FLOODING           0        1440         2025-07-31 04:52:46
192.168.80.136   PACKET FLOODING           0        1440         2025-07-31 04:53:11

192.168.80.137   PACKET FLOODING           0        1440         2025-07-31 04:54:30
```

Если к абонентскому интерфейсу привязан профиль безопасности со включенной блокировкой по User-Agent, то через какое-то время в чёрный список помимо IP-адресов добавится ещё и User-Agent.
Все запросы с любого адреса, в котором будет заблокированный User-Agent, обрабатываться не будут:

```
IP black-list:
-------------------------------------------------------------------------------
IP address       Ban reason          AOR    AOR     Blocking    Time of blocking
                                            error   timeout
                                            count   in minutes
---------------   ----------------  ----   ------  ----------  -------------------
192.168.80.132   PACKET FLOODING           0       1440        2025-07-31 04:54:23
192.168.80.133   PACKET FLOODING           0       1440        2025-07-31 04:54:52
192.168.80.134   PACKET FLOODING           0       1440        2025-07-31 04:55:16
192.168.80.136   PACKET FLOODING           0       1440        2025-07-31 04:55:41
192.168.80.137   PACKET FLOODING           0       1440        2025-07-31 04:56:00

User-agent black-list:
-------------------------------------------------------------------------------
UA                    Ban reason        UA error    Forgive      Time of blocking
                                        count       time in
                                                    minutes
-----------------   ----------------   ---------   ----------   -------------------
sipflood            ACCOUNT HACKING    138         60           2025-07-31 04:56:07
```

9.11.4 Объединение ошибок по IP-адресу

Данная опция позволяет объединять ошибки по IP-адресу.

Поведение по умолчанию – опция выключена, ошибки для каждого AOR/User-Agent считаются отдельно, блокируются при большом количестве ошибок с отдельного AOR/User-Agent.

При включенной опции, если ошибки имеют разный AOR/User-Agent, но одинаковый IP-адрес, то при блокировке адреса заблокируются все связанные AOR/User-Agent.

Настройка обеспечивает лучшую защиту от распределенных атак, но если много разных AOR/UA используют одинаковый IP-адрес, то могут быть ложные срабатывания.

Объединение ошибок работает, только если в профиле безопасности включена блокировка по AOR или User-Agent.

Пример включения объединения ошибок по IP-адресу:

```
vesbc#
vesbc# configure
vesbc(config)# esbc
vesbc(config-esbc)# security profile SECURITY_PROFILE

#Сбор ошибок по AOR:
vesbc(config-esbc-security-profile)# check user-agent
#Объединение ошибок по адресу
vesbc(config-esbc-security-profile)# errors aggregation
vesbc(config-esbc-security-profile)# exit

#Привязка профиля безопасности к абонентскому интерфейсу:
vesbc(config-esbc)# user-interface sip ABONENTS
vesbc(config-esbc-user-interface-sip)# security profile SECURITY_PROFILE

#Применение и подтверждение изменений:
vesbc(config-esbc-user-interface-sip)# do commit
Configuration has been successfully applied and saved to flash. Commit timer started, changes
will be reverted in 600 seconds.
vesbc(config-esbc-user-interface-sip)# do confirm
Configuration has been confirmed. Commit timer canceled.
```

Пример блокировки:

С адреса 192.168.80.133 через абонентский интерфейс приходят сообщения INVITE с разными AOR, вызовы без регистрации на интерфейсе запрещены, поэтому эти запросы отбиваются 403 Forbidden.

Если в профиле безопасности, привязанному к абонентскому интерфейсу, отключено объединение ошибок по адресу, то через определенное количество запросов заблокируется только IP-адрес и запросы с него больше обрабатываться не будут:

```
IP black-list:
-------------------------------------------------------------------------------
IP address       Ban reason       AOR    AOR      Blocking     Time of blocking
                                         error    timeout
                                         count    in minutes
---------------  ---------------  -----  ------   ----------   -------------------

192.168.80.133   PACKET FLOODING         0        1440         2025-07-31 06:38:44
```

Если к абонентскому интерфейсу привязан профиль безопасности, в котором включено объединение ошибок по адресу, то через какое-то время в чёрный список помимо IP-адреса добавятся все связанные с этим адресом AOR.

Все запросы с любого адреса, в котором будут заблокированные AOR во From, обрабатываться не будут.

```
IP black-list:
-------------------------------------------------------------------------------------
IP address     Ban reason     AOR                    AOR   Blocking  Time of blocking
                                                     error timeout
                                                     count in minutes
-------------- ------------- ---------------------- ----- ---------- -------------------
192.168.80.133 IP BLOCKED    24018@anonymous.invalid 1     1439       2025-07-31 07:11:32
               ATTRIBUTES LIMIT

AOR black-list:
-------------------------------------------------------------------------------
AOR                       Ban reason         AOR      Forgive   Time of blocking
                                             error    time in
                                             count    minutes
------------------------  -----------------  ------   -------   -------------------
24001@anonymous.invalid   PACKET FLOODING    1        19        2025-07-31 07:11:29
24002@anonymous.invalid   PACKET FLOODING    1        19        2025-07-31 07:11:29
24003@anonymous.invalid   PACKET FLOODING    1        19        2025-07-31 07:11:29
24004@anonymous.invalid   PACKET FLOODING    1        19        2025-07-31 07:11:30
24005@anonymous.invalid   PACKET FLOODING    1        19        2025-07-31 07:11:30
24006@anonymous.invalid   PACKET FLOODING    1        19        2025-07-31 07:11:30
24007@anonymous.invalid   PACKET FLOODING    1        19        2025-07-31 07:11:30
24008@anonymous.invalid   PACKET FLOODING    1        19        2025-07-31 07:11:30
24009@anonymous.invalid   PACKET FLOODING    1        19        2025-07-31 07:11:31
24010@anonymous.invalid   PACKET FLOODING    1        19        2025-07-31 07:11:31
24011@anonymous.invalid   PACKET FLOODING    1        19        2025-07-31 07:11:31
24012@anonymous.invalid   PACKET FLOODING    1        19        2025-07-31 07:11:31
24013@anonymous.invalid   PACKET FLOODING    1        19        2025-07-31 07:11:31
24014@anonymous.invalid   PACKET FLOODING    1        19        2025-07-31 07:11:32
24015@anonymous.invalid   PACKET FLOODING    1        19        2025-07-31 07:11:32
24016@anonymous.invalid   PACKET FLOODING    1        19        2025-07-31 07:11:32
24017@anonymous.invalid   PACKET FLOODING    1        19        2025-07-31 07:11:32
24018@anonymous.invalid   PACKET FLOODING    1        58        2025-07-31 07:11:32
```

9.12 Настройка криптопрофилей

При использовании протоколов TLS или WebSocket Secure (WSS) в качестве SIP-транспорта, а также протокола DTLS-SRTP для шифрования RTP-трафика, возможно использование сертификатов, автоматически сгенерированных самим ESBC, сгенерированных по требованию пользователя на ESBC или загруженных пользователем. По умолчанию используются сертификаты, автоматически сгенерированные самим ESBC, дополнительных настроек для их использования не требуется.

Для генерации сертификатов и ключей средствами ESBC используется команда *crypto generate*. Подробное описание команд для генерации сертификатов и ключей на ESBC приведено в разделе Генерация и просмотр ключей и сертификатов.

Для загрузки сертификатов и ключей на устройство через CLI используется команда *copy*, пример:

```
vesbc# copy tftp://10.0.0.1:/ca.crt crypto:cert/ca.crt
```

⚠ Путь для сохранения сертификатов ca и cert — crypto:cert/
Путь для сохранения private-key — crypto:private-key/

Для управления пользовательскими сертификатами и версией TLS используется crypto profile.

Описание всех команд для настройки криптопрофилей приведено в разделе Настройки криптопрофиля.

Пример настройки crypto profile:

```
vesbc#
vesbc# configure
vesbc(config)# esbc
vesbc(config-esbc)# crypto profile CRYPTO-PROFILE

#Установка сертификата удостоверяющего центра (CA certificate):
vesbc(config-esbc-crypto-profile)# ca default_ca.pem

#Установка клиентского сертификата (X.509):
vesbc(config-esbc-crypto-profile)# cert default_cert.pem

#Установка private-key:
vesbc(config-esbc-crypto-profile)# private-key default_cert_key.pem

#Установка пароля private-key (необязательно):
vesbc(config-esbc-crypto-profile)# password private-key PASSWORD

#Установка минимальной и максимальной версии TLS (необязательно):
vesbc(config-esbc-crypto-profile)# tls min 1.1
vesbc(config-esbc-crypto-profile)# tls max 1.2
vesbc(config-esbc-crypto-profile)# exit
```

⚠ Если не устанавливать значения версии TLC, то при установлении соединения будет использоваться любая версия 1.0–1.3.
Настройки *tls min* и *tls max* используются только при применении crypto profile для SIP-транспорта и не используются для шифрования DTLS-SRTP при применении crypto-profile в медиапрофиле.

Для того чтобы использовать crypto profile для SIP-транспорта, необходимо его указать в настройках нужного транспорта:

```
vesbc(config-esbc)# sip transport SIP-TRANSPORT
vesbc(config-esbc-sip-transport)# crypto profile CRYPTO-PROFILE
vesbc(config-esbc-sip-transport)# exit
```

Настройки crypto profile будут использоваться, только если выбран режим работы SIP-транспорта tls или wss, для остальных режимов настройки игнорируются.

Для того чтобы использовать crypto profile для шифрования DTLS-SRTP, необходимо его указать в настройках медиа профиля:

```
vesbc(config-esbc)# media profile MEDIA-PROFILE
vesbc(config-esbc-media-profile)# crypto profile CRYPTO-PROFILE
vesbc(config-esbc-media-profile)# exit
```

Настройки crypto profile будут использоваться, только если выбран режим шифрования srtp keying dtls-srtp, для остальных режимов настройки игнорируются.

Пример использования crypto profile:

Задача:

Использовать сертификат, загруженный пользователем на ESBC, для абонентских подключений по tls версии 1.3 и шифрования медиа DTLS-SRTP.

Решение:

1. Выполнить базовую настройку ESBC для обеспечения маршрутизации абонентских подключений в сторону ECSS-10:

```
vesbc(config)# esbc
vesbc(config-esbc)# media resource USERS
vesbc(config-esbc-media-resource)# ip address 20.0.0.1
vesbc(config-esbc-media-resource)# exit
vesbc(config-esbc)# media resource ECSS
vesbc(config-esbc-media-resource)# ip address 192.168.1.1
vesbc(config-esbc-media-resource)# exit
vesbc(config-esbc)# sip transport USERS
vesbc(config-esbc-sip-transport)# ip address 20.0.0.1
vesbc(config-esbc-sip-transport)# port 5061
vesbc(config-esbc-sip-transport)# mode tls
vesbc(config-esbc-sip-transport)# exit
vesbc(config-esbc)# sip transport ECSS
vesbc(config-esbc-sip-transport)# ip address 192.168.1.1
vesbc(config-esbc-sip-transport)# port 5070
vesbc(config-esbc-sip-transport)# exit
vesbc(config-esbc)# route-table TO_ECSS
vesbc(config-esbc-route-table)# rule 0
vesbc(config-esbc-route-table-rule)# action direct-to-trunk ECSS
vesbc(config-esbc-route-table-rule)# exit
vesbc(config-esbc-route-table)# exit
vesbc(config-esbc)# trunk sip ECSS
vesbc(config-esbc-trunk-sip)# sip transport ECSS
vesbc(config-esbc-trunk-sip)# media resource 0 ECSS
vesbc(config-esbc-trunk-sip)# remote address 192.168.1.2
vesbc(config-esbc-trunk-sip)# remote port 5070
vesbc(config-esbc-trunk-sip)# exit
vesbc(config-esbc)# user-interface sip USERS
vesbc(config-esbc-user-interface-sip)# sip transport USERS
vesbc(config-esbc-user-interface-sip)# route-table TO_ECSS
vesbc(config-esbc-user-interface-sip)# media resource 0 USERS
vesbc(config-esbc-user-interface-sip)# exit
```

2. Загрузить файлы сертификата, CA и private-key на ESBC через CLI (в примере указан протокол tftp):

```
#Загрузка CA сертификата:
vesbc# copy tftp://10.0.0.1:/ca.crt crypto:cert/ca.crt

#Загрузка клиентского сертификата:
vesbc# copy tftp://10.0.0.1:/cert.crt crypto:cert/cert.crt

#Загрузка private-key:
vesbc# copy tftp://10.0.0.1:/key.pem crypto:private-key/key.pem
```

3. Создать crypto profile, указать в нем файлы сертификатов, private-key и версию TLS:

```
vesbc(config-esbc)# crypto profile CRYPTO_PROFILE
vesbc(config-esbc-crypto-profile)# ca ca.crt
vesbc(config-esbc-crypto-profile)# cert cert.crt
vesbc(config-esbc-crypto-profile)# private-key key.pem
vesbc(config-esbc-crypto-profile)# tls min 1.3
vesbc(config-esbc-crypto-profile)# tls max 1.3
vesbc(config-esbc-crypto-profile)# exit
```

4. Создать медиапрофиль для использования DTLS-SRTP и привязать к нему crypto profile:

```
vesbc(config-esbc)# media profile MP_USERS
vesbc(config-esbc-media-profile)# srtp mode mandatory
vesbc(config-esbc-media-profile)# srtp keying dtls-srtp
vesbc(config-esbc-media-profile)# crypto profile CRYPTO_PROFILE
vesbc(config-esbc-media-profile)# exit
```

5. Привязать media profile MP_USERS к user-interface sip USERS:

```
vesbc(config-esbc)# user-interface sip USERS
vesbc(config-esbc-user-interface-sip)# media profile MP_USERS
vesbc(config-esbc-user-interface-sip)# exit
```

6. Привязать crypto profile CRYPTO_PROFILE к sip transport USERS:

```
vesbc(config-esbc)# sip transport USERS
vesbc(config-esbc-sip-transport)# crypto profile CRYPTO_PROFILE
vesbc(config-esbc-sip-transport)# exit
```

7. Применить настройки:

```
vesbc(config-esbc)# do commit
Configuration has been successfully applied and saved to flash. Commit timer started, changes
will be reverted in 600 seconds.
vesbc(config-esbc)# do confirm
Configuration has been confirmed. Commit timer canceled.
```

9.13 Настройка NAT

С целью преодоления соединений через устройства NAT, в ESBC реализована поддержка nat comedia-mode для абонентов и транков.

Настройка и принцип работы nat comedia-mode для транков (trunk)

Включение режима nat comedia-mode осуществляется в настройках транка:

```
vesbc# configure
vesbc(config)# esbc
vesbc(config-esbc)# trunk sip
vesbc(config-esbc-trunk-sip)# nat comedia-mode
    Select NAT comedia mode for trunk:
        off
        on
        flexible

vesbc(config-esbc-trunk-sip)# nat comedia-mode on
```

Возможна работа в двух режимах:

- *flexible* — проверяет источник во входящем RTP-потоке и транслирует исходящий поток на IP-адрес и UDP-порт, с которого принимается медиапоток, в случае прерывания входящего RTP-потока более чем на 1 секунду, исходящий поток продолжает транслироваться;
- *on* — проверяет источник во входящем RTP-потоке и транслирует исходящий поток на IP-адрес и UDP-порт, с которого принимается медиапоток, в случае прерывания входящего RTP-потока более чем на 1 секунду, исходящий поток перестает транслироваться.

Настройка и принцип работы nat comedia-mode для абонентов (user-interface)

Включение режима nat comedia-mode осуществляется в настройках абонентского интерфейса:

```
vesbc#
vesbc# configure
vesbc(config)# esbc
vesbc(config-esbc)# user-interface sip USERS
vesbc(config-esbc-user-interface-sip)# nat comedia-mode
    Select NAT comedia mode for user-interface:
        off
        on
        flexible
vesbc(config-esbc-user-interface-sip)# nat comedia-mode on
```

Возможна работа в двух режимах:

- *flexible* — проверяет источник во входящем RTP-потоке и транслирует исходящий поток на IP-адрес и UDP-порт, с которого принимается медиапоток, в случае прерывания входящего RTP-потока более чем на 1 секунду, исходящий поток продолжает транслироваться;
- *on* — проверяет источник во входящем RTP-потоке и транслирует исходящий поток на IP-адрес и UDP-порт, с которого принимается медиапоток, в случае прерывания входящего RTP-потока более чем на 1 секунду, исходящий поток перестает транслироваться.

Также данная настройка позволяет передавать сообщения протокола SIP симметрично (на порт, с которого был принят запрос) в случае, если клиент в инициирующем запросе не использовал параметр RPORT.

Команда *nat keep-alive-interval* в настройках абонентского интерфейса используется для настройки интервала для поддержки соединения за NAT. При включении опции, абоненту, с заданным интервалом будут отправляться пакеты с содержанием "0d0a" для предотвращения разрушения сессии на NAT.

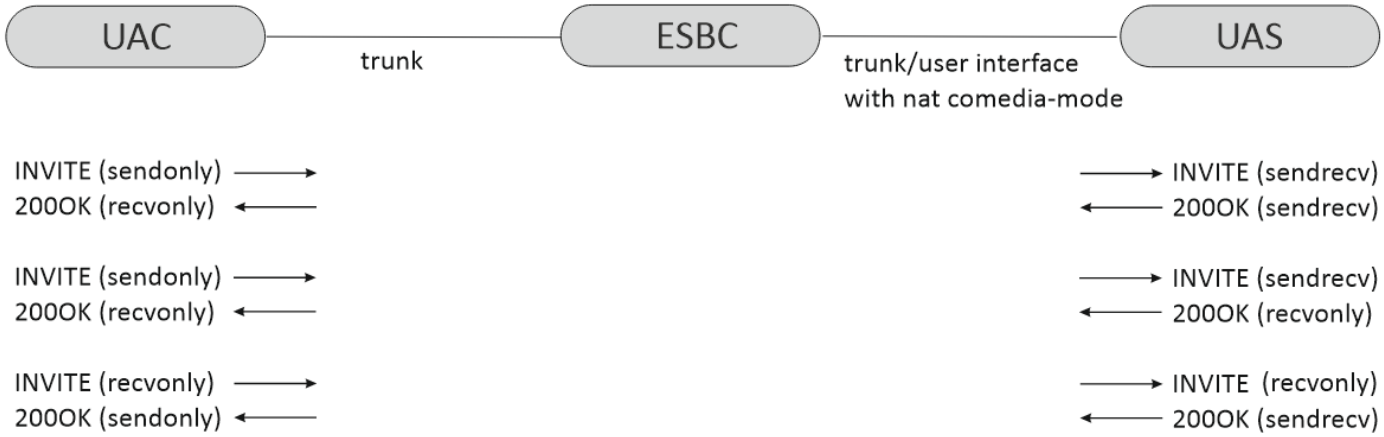
Подмена атрибутов direction в SDP

При включении опции *nat comedia-mode* атрибут direction sendonly в SDP при отправке offer/answer sdp заменяются на sendrecv.

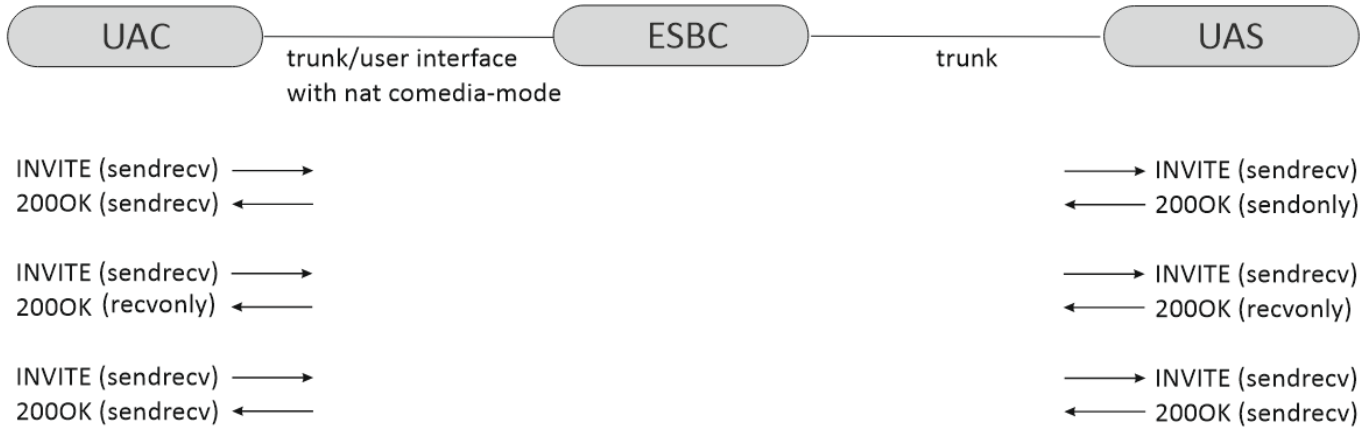
Данный механизм используется для предотвращения ситуации в которой абонент за NAT не начнет первым отправку RTP-пакетов в сторону ESBC и соответственно ESBC не начнет отправку встречного потока RTP к абоненту.

Примеры:

1. Замена атрибутов direction в offer sdp:



2. Замена атрибутов direction в answer sdp:



9.14 Настройка Public IP

Public IP (рус. «публичный IP-адрес») — это внешний IP-адрес, который используется при отправке запросов пользователю или удаленному адресу из внешней сети.

Настройка используется в случае, когда ESBC не имеет публичного IP-адреса и выход в публичную сеть осуществляется через NAT. В таком случае в качестве Public IP указывается адрес WAN-интерфейса NAT для подстановки в сигнальные сообщения протокола SIP.

Public IP можно можно настроить для абонентского интерфейса, транка и транковой группы.

Если Public IP настроен в транке и в транковой группе, в которую входит этот транк, то будет использоваться Public IP из настроек транка.

В качестве публичного адреса можно использовать как IPv4, так и IPv6 адрес.

При наличии Public IP, адреса в SDP, заголовках Via и Contact будут заменены на значение public-ip из конфигурации объекта. Media будет работать в режиме NAT-comedia.

Для корректной работы опции Public IP необходимо организовать проброс портов для сигнализации SIP и медиапортов RTP на вышестоящем устройстве NAT "один к одному".

Пример настройки Public IP для транка

```
#Настройка SIP-транспорта для транка:
vesbc# configure
vesbc(config)# esbc
vesbc(config-esbc)# sip transport TRUNK_TRANSPORT
vesbc(config-esbc-sip-transport)# ip address 192.168.1.1
vesbc(config-esbc-sip-transport)# exit

#Настройка медиаресурсов для транка:
vesbc(config-esbc)# media resource TRUNK_MEDIA
vesbc(config-esbc-media-resource)# ip address 192.168.1.1
vesbc(config-esbc-media-resource)# exit

#Настройка параметров транка:
vesbc(config-esbc)# trunk sip TRUNK_PUBLIC_IP
vesbc(config-esbc-trunk-sip)# sip transport TRUNK_TRANSPORT
vesbc(config-esbc-trunk-sip)# media resource 0 TRUNK_MEDIA
vesbc(config-esbc-trunk-sip)# remote address 192.168.1.3
vesbc(config-esbc-trunk-sip)# remote port 5060
vesbc(config-esbc-trunk-sip)# public-ip 10.25.0.1

#Применение и подтверждение изменений:
vesbc(config-esbc-trunk-sip)# do commit
Configuration has been successfully applied and saved to flash. Commit timer started, changes
will be reverted in 600 seconds.
vesbc(config-esbc-trunk-sip)# do confirm
Configuration has been confirmed. Commit timer canceled.
```

Пример использования Public IP

ESBC получает сообщение, которое должно быть смаршрутизировано в транк TRUNK_PUBLIC_IP:

```
INVITE sip:23002@192.168.1.1:5060 SIP/2.0
Via: SIP/2.0/UDP 192.168.1.4:5061;rport;branch=z9hG4bK-1914230-1-1
From: "24001" <sip:24001@192.168.1.4:5061>;tag=1
To: "23002" <sip:23002@192.168.1.1:5060>
Call-ID: 1-1914230@192.168.1.4
Cseq: 1 INVITE
Contact: <sip:24001@192.168.1.4:5061>
Max-Forwards: 70
Allow: INVITE, ACK, BYE, CANCEL
Content-Type: application/sdp
Content-Length: 138

Session Description Protocol Version (v): 0
Owner/Creator, Session Id (o): user1 77755765 7773687637 IN IP4 192.168.1.4
Session Name (s): -
Time Description, active time (t): 0 0
Connection Information (c): IN IP4 192.168.1.4
Media Description, name and address (m): audio 10000 RTP/AVP 8
Media Attribute (a): rtpmap:8 PCMA/8000
```

ESBC пересылает INVITE в транк TRUNK_PUBLIC_IP.
В SDP, Via и Contact вместо адреса привязанного SIP-транспорта (192.168.1.1) используется Public IP транка (10.25.0.1):

```
INVITE sip:23002@192.168.1.3:5060 SIP/2.0
Via: SIP/2.0/UDP 10.25.0.1:5060;rport;branch=z9hG4bKPj6e357f04-e13e-4ead-8386-2246d12450b4
Max-Forwards: 70
From: "24001" <sip:24001@192.168.1.1>;tag=76776c9a-022b-4ccf-9458-e83e2701f6c8
To: "23002" <sip:23002@192.168.1.3>
Contact: <sip:24001@10.25.0.1:5060;transport=udp>
Call-ID: 5fc229f6657d7706f2b6c81a44a5b10e
CSeq: 28491 INVITE
Allow: INVITE, ACK, BYE, CANCEL
Supported: 100rel, replaces, ice
Content-Type: application/sdp
Content-Length:   135

Session Description Protocol Version (v): 0
Owner/Creator, Session Id (o): user1 77755765 7773687637 IN IP4 10.25.0.1
Session Name (s): -
Time Description, active time (t): 0 0
Connection Information (c): IN IP4 10.25.0.1
Media Description, name and address (m): audio 10000 RTP/AVP 8
Media Attribute (a): rtpmap:8 PCMA/8000
```

9.15 Настройка QoS

QoS (Quality of Service, рус. «качество обслуживания») — технология предоставления различным классам данных различных приоритетов в обслуживании.

Приоритет определяется значением DSCP (0-63) в поле IP-заголовка DS.

Установить необходимое значение DS можно отдельно для:

- аудио-пакетов;
- видео-пакетов;
- пакетов сигнализации-SIP.

Параметры QoS настраиваются в конфигурации абонентского интерфейса, транка и транковой группы.

> Если QoS настроен для транка и для транковой группы, в которую входит этот транк, то будет использоваться QoS из настроек транка.

Пример настройки QoS для аудиотрафика в конфигурации абонентского интерфейса:

```
vesbc# configure
vesbc(config)# esbc
vesbc(config-esbc)# user-interface sip ABONENT_QOS_50
vesbc(config-esbc-user-interface-sip)# sip transport ABONENT_TRANSPORT
vesbc(config-esbc-user-interface-sip)# media resource 0 ABONENT_MEDIA
vesbc(config-esbc-user-interface-sip)# route-table TO_SSW
vesbc(config-esbc-user-interface-sip)# dscp audio 50

#Применение и подтверждение изменений:
vesbc(config-esbc-user-interface-sip)# do commit
Configuration has been successfully applied and saved to flash. Commit timer started, changes
will be reverted in 600 seconds.
vesbc(config-esbc-user-interface-sip)# do confirm
Configuration has been confirmed. Commit timer canceled.
```

В исходящем аудиотрафике абонентам, зарегистрированным через интерфейс ABONENT_QOS_50, поле DS в IP-пакете будет выглядеть следующим образом:

```
Differentiated Services Field: 0xc8 (DSCP: Unknown, ECN: Not-ECT)
    1100 10.. = Differentiated Services Codepoint: Unknown (50)
    .... ..00 = Explicit Congestion Notification: Not ECN-Capable Transport (0)
```

9.16 Изменение количества модулей

При обработке сигнального SIP-трафика и медиапотоков RTP, ресурсы CPU используются разными модулями ESBC. Соответственно для оптимизации нагрузки на CPU предусмотрена возможность управлять количеством модулей.

При высокой нагрузке сигнальным SIP-трафиком наибольшую нагрузку на ядро CPU производит модуль sip worker, а при большом количестве одновременных вызовов (особенно в режиме транскодирования медии) — media worker.

Поэтому для установления баланса производительности, для многоядерных систем следует использовать оптимальное количество каждого из модулей, т. к. каждый дополнительный экземпляр модуля будет использовать ресурс дополнительного ядра CPU системы.

По умолчанию в системе используется по одному экземпляру каждого модуля.

Список модулей, количество которых можно изменить:

- core
- sip worker
- sip balancer
- media worker
- media balancer

Максимальное количество модулей определяется динамически в зависимости от количества ядер CPU.

После изменения количества модулей для стабильной работы необходим перезапуск ПО ESBC.

Заданное в конфигурации количество модулей не изменяется при увеличении/уменьшении количества ядер CPU системы.

Описание всех команд для настройки количества модулей приведено в разделе Общие настройки ESBC.

Пример:

```
vesbc#
vesbc# config
vesbc(config)# esbc

#Переход в общие настройки:
vesbc(config-esbc)# general
vesbc(config-esbc-general)#

#Увеличение количества медиа-воркеров до 2:
vesbc(config-esbc-general)# count media worker 2
vesbc(config-esbc-general)#

#Применение и подтверждение изменений:
vesbc(config-esbc-general)# do commit
2024-09-09T05:26:55+00:00 %SYS-W-EVENT: WARNING!!! After changing ESBC modules count, the
system may work unstable. Please restart software.
2024-09-09T05:26:57+00:00 snmpd restarted
Configuration has been successfully applied and saved to flash. Commit timer started, changes
will be reverted in 600 seconds.
2024-09-09T05:26:58+00:00 %CLI-I-CRIT: user admin from console  input: do commit
vesbc(config-esbc-general)# do confirm
Configuration has been confirmed. Commit timer canceled.
2024-09-09T05:27:01+00:00 %CLI-I-CRIT: user admin from console  input: do confirm
vesbc(config-esbc-general)#

#Перезапуск ПО ESBC для корректного перераспределения модулей:
vesbc(config-esbc-general)# do reload esbc force
Do you really want to reload esbc now? (y/N): y
```

⚠ Для вывода предупреждения о необходимости перезапуска нужно, чтобы уровень syslog severity был не ниже warning.

9.17 Ограничение входящего трафика

На ESBC имеется возможность контролировать интенсивность входящего трафика. В конфигурации доступна настройка максимального количества:

- вызовов в секунду (max cps);
- одновременных вызовов (max calls);
- запросов в секунду (max rps).

Ограничения можно настроить для всей системы и отдельно для транка, транковой группы, абонентского интерфейса.

Пример глобального ограничения:

```
vesbc#
vesbc# config
vesbc(config)# esbc

#Переход в общие настройки:
vesbc(config-esbc)# general
vesbc(config-esbc-general)#

#Ограничение максимального RPS:
vesbc(config-esbc-general)# max rps
  COUNT  Possible max rps: 1-4294967295

vesbc(config-esbc-general)# max rps 40

#Применение и подтверждение изменений:
vesbc(config-esbc-general)# do commit
2025-04-22T08:44:46+00:00 snmpd restarted
Configuration has been successfully applied and saved to flash. Commit timer started, changes
will be reverted in 600 seconds.
2025-04-22T08:44:46+00:00 %CLI-I-CRIT: user admin from console  input: do commit
vesbc(config-esbc-general)# do confirm
Configuration has been confirmed. Commit timer canceled.
vesbc(config-esbc-general)#
```

После применения изменений ESBC не будет обрабатывать более 40 входящих SIP-запросов в секунду.

Пример ограничения на транке:

```
vesbc#
vesbc# config
vesbc(config)# esbc

#Переход в настройки транка:
vesbc(config-esbc)# trunk sip TRUNK
vesbc(config-esbc-trunk-sip)#

#Ограничение максимального CPS:
vesbc(config-esbc-trunk-sip)# max cps 10
vesbc(config-esbc-trunk-sip)#

#Применение и подтверждение изменений:
vesbc(config-esbc-trunk-sip)# do commit
2025-04-22T08:44:46+00:00 snmpd restarted
Configuration has been successfully applied and saved to flash. Commit timer started, changes
will be reverted in 600 seconds.
2025-04-22T08:44:46+00:00 %CLI-I-CRIT: user admin from console  input: do commit
vesbc(config-esbc-trunk-sip)# do confirm
Configuration has been confirmed. Commit timer canceled.
vesbc(config-esbc-trunk-sip)#
```

После применения изменений ESBC не будет обрабатывать более 10 входящих вызовов на SIP-транк TRUNK в секунду.

Пример ограничения на абонентском интерфейсе:

```
vesbc#
vesbc# config
vesbc(config)# esbc

#Переход в настройки абонентского интерфейса:
vesbc(config-esbc)# user-interface sip USER_IFACE
vesbc(config-esbc-user-interface)#

#Ограничение максимального количества одновременных вызовов:
vesbc(config-esbc-user-interface-sip)# max calls 500
vesbc(config-esbc-user-interface-sip)#

#Применение и подтверждение изменений:
vesbc(config-esbc-user-interface)# do commit
2025-04-22T08:44:46+00:00 snmpd restarted
Configuration has been successfully applied and saved to flash. Commit timer started, changes
will be reverted in 600 seconds.
2025-04-22T08:44:46+00:00 %CLI-I-CRIT: user admin from console  input: do commit
vesbc(config-esbc-user-interface)# do confirm
Configuration has been confirmed. Commit timer canceled.
vesbc(config-esbc-user-interface)#
```

После применения изменений ESBC не будет обрабатывать более 500 входящих вызовов на абонентский интерфейс USER_IFACE.

Ограничение трафика на транковой группе

Ограничение на транковой группе применяется для всех транков, входящих в состав этой группы, и имеет приоритет над ограничением, установленным в настройках транка.
При этом суммарное количество входящего трафика на транках, входящих в состав группы, также не может превышать ограничение на группе.

Пример:

```
vesbc#
vesbc# config
vesbc(config)# esbc

#Переход в настройки транка:
vesbc(config-esbc)# trunk sip TRUNK_1
vesbc(config-esbc-trunk-sip)#

#Ограничение максимального CPS на транке:
vesbc(config-esbc-trunk-sip)# max cps 50

#Переход в настройки транковой группы и добавление транков:
vesbc(config-esbc-trunk-sip)# exit
vesbc(config-esbc)# trunk-group GROUP
vesbc(config-esbc-trunk-group)# trunk 0 TRUNK_0
vesbc(config-esbc-trunk-group)# trunk 1 TRUNK_1
vesbc(config-esbc-trunk-group)# trunk 2 TRUNK_2

#Ограничение максимального CPS на группе:
vesbc(config-esbc-trunk-group)# max cps 30

#Применение и подтверждение изменений:
vesbc(config-esbc-trunk-group)# do commit
2025-04-22T08:44:46+00:00 snmpd restarted
Configuration has been successfully applied and saved to flash. Commit timer started, changes
will be reverted in 600 seconds.
2025-04-22T08:44:46+00:00 %CLI-I-CRIT: user admin from console  input: do commit
vesbc(config-esbc-trunk-group)# do confirm
Configuration has been confirmed. Commit timer canceled.
vesbc(config-esbc-trunk-group)#
```

После применения изменений на транках TRUNK_0, TRUNK_1, TRUNK_2 не может быть суммарно более 30 входящих вызовов в секунду.

Лицензионное ограничение обработки вызовов

Максимальное количество одновременных вызовов и максимальное количество вызовов в секунду ограничиваются лицензиями ESBC-LIMIT-MAX-CALLS и ESBC-LIMIT-MAX-CPS соответственно.
При этом в конфигурации можно задать ограничение, которое превышает лицензионное значение, но ESBC не будет обрабатывать больше, чем позволяет лицензия, пример:

```
#Просмотр активных лицензий:
vesbc# show licence
Feature                          Source     State          Value          Valid from  Expiries
-------------------------------  --------   -----------   -------------  ---------   --------
ESBC-LIMIT-MAX-CALLS                ELM      Active         5000             --         --
ESBC-LIMIT-MAX-CPS                  ELM      Active         100              --         --
ESBC-VIRTUAL-LIMIT-DEFAULT          ELM      Active         true             --         --
ESBC-VIRTUAL-LIMIT-NET              ELM      Active         10000000000      --         --
vesbc#
vesbc# config
vesbc(config)# esbc

#Переход в общие настройки:
vesbc(config-esbc)# general
vesbc(config-esbc-general)#

#Ограничение максимального CPS:
vesbc(config-esbc-general)# max cps
  COUNT  Possible max cps: 1-1000   #конфигурационное ограничение

vesbc(config-esbc-general)# max cps 1000
2025-04-22T09:10:17+00:00 %SYS-W-EVENT: WARNING!!! Configured max cps 1000 exceed licence limit
that is equal to 100   #предупреждение о том, что введённое значение превышает лицензионное

#Применение и подтверждение изменений:
vesbc(config-esbc-general)# do commit
2025-04-22T08:44:46+00:00 snmpd restarted
Configuration has been successfully applied and saved to flash. Commit timer started, changes
will be reverted in 600 seconds.
2025-04-22T08:44:46+00:00 %CLI-I-CRIT: user admin from console  input: do commit
vesbc(config-esbc-general)# do confirm
Configuration has been confirmed. Commit timer canceled.
vesbc(config-esbc-general)#
```

После применения изменений в конфигурации будет отображаться max cps 1000, но обрабатываться будет не более 100 вызовов в секунду.

9.18 Мониторинг

В ESBC доступен мониторинг следующих параметров:

- активные вызовы;
- чёрный список;
- белый список;
- состояние транков;
- список зарегистрированных абонентов;
- статистика SIP.

Активные вызовы

Поддержана возможность просматривать активные вызовы командой show esbc active calls в CLI.

В выводе информации об активных сессиях присутствует:

- Total call sessions — общее количество активных сессий;
- Session id — id активной сессии;
- Duration (sec) — длительность активной сессии в секундах;
- CGPN unmodified — номер вызывающей стороны до модификаций;
- CDPN unmodified — номер вызываемой стороны до модификаций;
- Source — источник вызова;
- Destination — место назначения вызова;
- CGPN modified — номер вызывающей стороны после модификаций (если модификаций нет или они не применялись, то номер останется без изменений);
- CDPN modified — номер вызываемой стороны после модификаций (если модификаций нет или они не применялись, то номер останется без изменений).

Присутствует возможность удаления активных сессий командой clear esbc active calls в CLI.

Поддерживается вывод до 50000 активных звонков.

Черный список

Поддержана возможность просматривать черный список командой show esbc black-list (IP-адреса, AOR, User-Agent) в CLI и на странице Мониторинг → Списки доступа → Чёрный список (IP-адреса) в WEB.

В выводе черного списка может присутствовать до 3 таблиц (по блокируемым объектам):

1. IP black-list:
   - IP address — заблокируемый IP-адрес;
   - Ban reason — причина блокировки;
   - AOR;
   - AOR error count — количество ошибок AOR;
   - Blocking timeout in minutes — оставшееся время блокировки в минутах;
   - Time of blocking — время блокировки.
2. AOR:
   - AOR;
   - Ban reason — причина блокировки;
   - AOR error count — количество ошибок AOR;
   - Forgive time in minutes — оставшееся время блокировки в минутах;
   - Time of blocking — время блокировки.
3. User-agent black-list:
   - UA;
   - Ban reason — причина блокировки;
   - UA error count — количество ошибок UA;

- Forgive time in minutes — оставшееся время блокировки в минутах;
- Time of blocking — время блокировки.

Причины блокировок описаны в разделе Общий принцип работы модуля fail2ban.

Присутствует возможность очистки черного списка командой clear esbc black-list в CLI или кнопкой «Удалить» в WEB.

Белый список

Поддержана возможность просматривать белый список адресов командой show esbc white-list в CLI и на странице Мониторинг → Списки доступа → Белый список в WEB.

Реализовано добавление в белый список динамических адресов и доменов.

В белом списке также присутствуют два параметра:

- Is dynamic — объекты, которые не присутствуют в конфигурации ESBC, но были подтверждены иным способом (например, адрес абонента при регистрации заносится в белый список);
- Is configured — объекты, которые присутствуют в конфигурации ESBC.

Состояние транков

Поддержана возможность просматривать состояние транков командой show esbc trunks в CLI или на странице Мониторинг → Телефония → Транки в WEB.

В таблице выводится:

- Trunk — имя транка;
- Trunk type — тип транка;
- Status — статус транка (принимает значения Uncontrolled, Available или Not available, в зависимости от настройки SIP профиля и реального состояния транка);
- Last change time — время изменения статуса транка.

Список зарегистрированных абонентов

Поддержана возможность просматривать просмотра зарегистрированных абонентов командой show esbc users в CLI или на странице Мониторинг → Телефония → Абоненты в WEB.

Выводится общее количество AOR и Contact, а также базовый вывод информации о абонентах.

Количество AOR и Contact может не совпадать, если абоненты имеют несколько Contact.

Также есть возможность просмотра подробной информации по конкретному абоненту, используя дополнительный параметр после основной команды (sip <AOR> detailed).

Выводится подробная информация по определенному AOR:

- User AOR;
- User type — тип абонента;
- IN User contact — входящий заголовок Contact;
- IP address of user — IP-адрес абонента;
- User interface name — user-interface, через который зарегистрировался абонент;
- Expires — время перерегистрации;
- Registration expires in — время до перерегистрации;
- Trunk name — trunk, на который отправлен запрос Register от абонента;
- IP address of registrar — IP-адрес сервера регистрации;
- OUT Trunk contact — исходящий заголовок Contact.

Присутствует возможность удаления активных регистраций командой clear esbc registration в CLI.

Статистика SIP

Есть возможность просматривать статистику для всей системы, всех транков, всех абонентских интерфейсов или по конкретному объекту командой show esbc statistics (вызовы, регистрации, подписки, RPS) в CLI или на странице Мониторинг → Телефония → Статистика вызовов в WEB.

Ведение статистики по умолчанию включено.

При вызове команды для просмотра статистики отображаются таблицы с метриками, описание каждой метрики можно найти в разделе show esbc statistics Справочника команд CLI.

В CLI отображаются счётчики за последнюю секунду. В WEB есть возможность просмотреть историю за последнюю минуту, час, 3 дня.

Для отключения ведения статистики необходимо в меню general отключить ее командой statistics disable.

Пример:

Из TRUNK_IN в TRUNK_OUT через ESBC поступает 2 вызова каждую секунду длительностью 25 секунд.

```
#Просмотр полной статистики при активных вызовах:
vesbc# show esbc statistics all
ESBC global call counters:
------------------------------------------------------------------------
Counter Name                Incoming               Outgoing
-------------------------   --------------------   --------------------
CALLS PER SECOND            2                      2
CALL LEGS                   50                     50
REQUESTS IN CALL            6                      7
RESPONSES IN CALL           8                      8
ANSWERED CALLS              2                      2
CALLS TO WRONG NUMBER       0                      0
BUSY CALLS                  0                      0
NO ANSWERED CALLS           0                      0
FORBIDDEN CALLS             0                      0
UNAUTHORIZED CALLS          0                      0
3XX CODES                   0                      0
4XX CODES                   0                      0
5XX CODES                   0                      0
6XX CODES                   0                      0

ESBC global register counters:
------------------------------------------------------------------------
Counter Name                Incoming               Outgoing
-------------------------   --------------------   --------------------
REGISTERS PER SECOND        0                      0
REGISTER TRANSACTIONS       0                      0
RESPONSES                   0                      0
SUCCESS REGISTERS           0                      0
REQUEST TIMEOUT             0                      0
FORBIDDEN REGISTERS         0                      0
UNAUTHORIZED REGISTERS      0                      0
INTERVAL TOO BRIEF          0                      0
3XX CODES                   0                      0
4XX CODES                   0                      0
5XX CODES                   0                      0
6XX CODES                   0                      0

ESBC global subscribe counters:
------------------------------------------------------------------------
Counter Name                Incoming               Outgoing
-------------------------   --------------------   --------------------
SUBSCRIBES PER SECOND       0                      0
ACTIVE SUBSCRIBES           0                      0
REQUESTS IN SUBSCRIBE       0                      0
RESPONSES IN SUBSCRIBE      0                      0
SUCCESS SUBSCRIBES          0                      0
REQUEST TIMEOUT             0                      0
FORBIDDEN SUBSCRIBES        0                      0
UNAUTHORIZED SUBSCRIBES     0                      0
INTERVAL TOO BRIEF          0                      0
3XX CODES                   0                      0
4XX CODES                   0                      0
5XX CODES                   0                      0
6XX CODES                   0                      0

ESBC global rps counters:
------------------------------------------------------------------------
```

```
Counter Name                Incoming                Outgoing
-------------------------   --------------------    --------------------
REQUESTS PER SECOND         6                       6
INVITE PER SECOND           2                       2
ACK PER SECOND              2                       2
BYE PER SECOND              2                       3
CANCEL PER SECOND           0                       0
REFER PER SECOND            0                       0
PRACK PER SECOND            0                       0
SUBSCRIBE PER SECOND        0                       0
NOTIFY PER SECOND           0                       0
UPDATE PER SECOND           0                       0
OPTIONS PER SECOND          0                       0
INFO PER SECOND             0                       0
REGISTER PER SECOND         0                       0
MESSAGE PER SECOND          0                       0

#Просмотр статистики вызовов после завершения вызовов:
vesbc# show esbc statistics call
ESBC global call counters:
-------------------------------------------------------------------------
Counter Name                Incoming                Outgoing
-------------------------   --------------------    --------------------
CALLS PER SECOND            0                       0
CALL LEGS                   0                       0
REQUESTS IN CALL            0                       0
RESPONSES IN CALL           0                       0
ANSWERED CALLS              0                       0
CALLS TO WRONG NUMBER       0                       0
BUSY CALLS                  0                       0
NO ANSWERED CALLS           0                       0
FORBIDDEN CALLS             0                       0
UNAUTHORIZED CALLS          0                       0
3XX CODES                   0                       0
4XX CODES                   0                       0
5XX CODES                   0                       0
6XX CODES                   0                       0
```

9.19 Аварии

Включение генерации аварий происходит включением SNMP-трапов командой snmp-server enable traps esbc в CLI.

Данная команда без указания параметров включает весь набор SNMP-трапов:

- cdr-send-error — ошибки отправки CDR;
- cdr-write-error — ошибки записи CDR;
- general-max-calls-limit — превышение общего лимита вызовов;
- general-max-cps-limit — превышение общего лимита cps;
- general-max-rps-limit — превышение общего лимита rps;
- media-resources — превышение медиаресурсов;
- module-connection — падение модулей ESBC;
- trunk-group-max-calls-limit — превышение лимита вызовов на транк-группе;
- trunk-group-max-cps-limit — превышение лимита cps на транк-группе;
- trunk-group-max-rps-limit — превышение общего лимита rps на транк-группе;
- trunk-max-calls-limit — превышение лимита вызовов на транке;
- trunk-max-cps-limit — превышение лимита cps на транке;
- trunk-max-rps-limit — превышение общего лимита rps на транке;

- trunk-unavailable — недоступность транка;
- user-interface-max-calls-limit — превышение лимита вызовов на user-interface;
- user-interface-max-cps-limit — превышение лимита cps на user-interface;
- user-interface-max-rps-limit — превышение общего лимита rps на user-interface;
- voip-block-aor — блокировки по AOR;
- voip-block-ip — блокировки по IP-адресу;
- voip-block-user-agent — блокировки по User-Agent.

Более подробное описание конфигурирования SNMP-трапов можно прочитать в разделе Управление SNMP.

Список аварийных событий выводится командой show alarms brief в CLI. Данная команда выводит историю аварий, включая уже нормализированные аварии.

Для отображения только активных аварий используется команда show alarms brief active.

Текст аварий и причины их нормализации представлены в таблице ниже.

Авария	Текст аварии	Причины нормализации
cdr-send-error	CDR alarm: failed to send to <main | reserve> ftp server	успешная отправка CDR на FTP-сервер
cdr-write-error	CDR alarm: failed to write	успешная запись CDR
general-max-calls-limit	Host <host_name> max calls limit reached	через 10 секунд после последней аварии ИЛИ при снижении счетчика на 80% от лимита
general-max-cps-limit	Host <host_name> max cps limit reached	через 10 секунд после последней аварии
general-max-rps-limit	Host <host_name> max rps limit reached	через 10 секунд после последней аварии
media-resources	Session<session_id>: <Trunk/User interface><trunk_name/ui_name> media resources out	через 15 секунд после последней аварии ИЛИ уничтожение превышающий сессии
module-connection	Module <module_type> host <host_id> is down	при успешном добавлении модуля в диспетчер
trunk-group-max-calls-limit	Trunk-Group <trunk_group_name> max calls limit reached	через 10 секунд после последней аварии ИЛИ при снижении счетчика на 80% от лимита
trunk-group-max-cps-limit	Trunk-Group <trunk_group_name> max cps limit reached	через 10 секунд после последней аварии
trunk-group-max-rps-limit	Trunk-Group <trunk_group_name> max rps limit reached	через 10 секунд после последней аварии

Авария	Текст аварии	Причины нормализации
trunk-max-calls-limit	Trunk <trunk_name> max calls limit reached	через 10 секунд после последней аварии ИЛИ при снижении счетчика на 80% от лимита
trunk-max-cps-limit	Trunk <trunk_name> max cps limit reached	через 10 секунд после последней аварии
trunk-max-rps-limit	Trunk <trunk_name> max rps limit reached	через 10 секунд после последней аварии
trunk-unavailable	Trunk <trunk_name> is unavailable	при обновлении статуса транка на "Available"
user-interface-max-calls-limit	User interface <ui_name> max calls limit reached	через 10 секунд после последней аварии ИЛИ при снижении счетчика на 80% от лимита
user-interface-max-cps-limit	User interface <ui_name> max rps limit reached	через 10 секунд после последней аварии
user-interface-max-rps-limit	User interface <ui_name> max cps limit reached	через 10 секунд после последней аварии
voip-block-aor	Отсутствует	не нормализуются
voip-block-ip	Отсутствует	не нормализуются
voip-block-user-agent	Отсутствует	не нормализуются

Для удаления аварий используется команда clear alarms в CLI.

9.20 Настройка CDR

CDR (Call Detail Record, рус. «запись сведений о звонках») — это запись, содержащая подробную информацию о совершённых вызовах.

В файл CDR записываются следующие данные:

- Заголовок файла (опционален) (<hostname> CDR. File started at 'YYYYMMDDhhmmss');
- Отличительный признак (опционален);
- Время поступления вызова;
- Время ответа на вызов;
- Входящий номер вызывающего абонента;
- Исходящий номер вызывающего абонента;
- Входящий номер вызываемого абонента;
- Исходящий номер вызываемого абонента;
- Имя trunk/user-interface вызывающего абонента;
- Имя trunk/user-interface вызываемого абонента;
- Длительность вызова;
- Причина разъединения (согласно ITU-T Q.850);
- Индикатор успешного вызова (1 — успешный, 0 — неуспешный);
- Сторона-инициатор разъединения (1 — вызывающая сторона, 2 — вызываемая сторона, 3 — ESBC);
- Call-ID входящего вызова;
- Call-ID исходящего вызова;
- Номер вызываемого абонента при переадресации;
- IP-адрес шлюза вызывающего абонента;
- IP-адрес шлюза вызываемого абонента;
- Список IP-адресов из заголовка Record-Route при установлении соединения в направлении от вызывающего абонента;
- Список IP-адресов из заголовка Via при установлении соединения в направлении от вызывающего абонента;
- IP-адрес из заголовка Contact вызывающего абонента;
- IP-адрес из заголовка Contact вызываемого абонента.

Значения параметров в файле CDR записываются в указанном выше порядке и разделяются символом ";".

Хранение записей CDR осуществляется в локальном хранилище ESBC или на внешнем USB-накопителе.

Отправка на внешний сервер осуществляется по протоколу FTP. Поддерживается отправка на два FTP-сервера.

Дополнительно поддерживается сохранение и отправка SDR в SYSLOG. Для отправки в SYSLOG требуется дополнительная конфигурация syslog.

Описание всех команд для настройки CDR приведено в разделе Настройки CDR.

Пример настройки записи CDR с опциональными полями, локальным хранением и отправкой на сервер FTP с резервированием в случае неудачной отправки приведен ниже.

```
vesbc#
vesbc# configure
vesbc(config-esbc)# cdr
vesbc(config-esbc-cdr)# enable

#Добавление заголовка в CDR-запись:
vesbc(config-esbc-cdr)# add-header

#Запись неудачных вызовов:
vesbc(config-esbc-cdr)# collect unsuccess

#Запись пустых CDR:
vesbc(config-esbc-cdr)# collect empty-files

#Режим создания записей:
vesbc(config-esbc-cdr)# create-mode periodically
vesbc(config-esbc-cdr)# per days 1
vesbc(config-esbc-cdr)# period hours 12
vesbc(config-esbc-cdr)# per minutes 30

#Включение отправки логов:
vesbc(config-esbc-cdr)# syslog enable

#Добавление отличительного признака:
vesbc(config-esbc-cdr)# signature otlichitelnyi_priznak

#Настройка локального хранения:
vesbc(config-esbc-cdr)# local
vesbc(config-esbc-cdr-local)# create-directories by-date
vesbc(config-esbc-cdr-local)# keep days 30
vesbc(config-esbc-cdr-local)# keep hours 12
vesbc(config-esbc-cdr-local)# keep minutes 30
vesbc(config-esbc-cdr-local)# path flash:cdr/cdr_record
vesbc(config-esbc-cdr-local)# save
vesbc(config-esbc-cdr-local)# exit

#Настройка основного FTP-сервера:
vesbc(config-esbc-cdr)# ftp
vesbc(config-esbc-cdr-ftp)# login main_ftp_server
vesbc(config-esbc-cdr-ftp)# password password_m_ftp
vesbc(config-esbc-cdr-ftp)# path /main_ftp/cdr_record
vesbc(config-esbc-cdr-ftp)# remote address 192.168.23.100
vesbc(config-esbc-cdr-ftp)# save
vesbc(config-esbc-cdr-ftp)# exit

#Настройка резервного FTP-сервера:
vesbc(config-esbc-cdr)# reserved-ftp
vesbc(config-esbc-cdr-res-ftp)# as-reserved
vesbc(config-esbc-cdr-res-ftp)# login reserve_ftp_server
vesbc(config-esbc-cdr-res-ftp)# password password_r_ftp
vesbc(config-esbc-cdr-res-ftp)# path /reserve_ftp/cdr_record
vesbc(config-esbc-cdr-res-ftp)# remote address 192.168.23.200
vesbc(config-esbc-cdr-res-ftp)# save

#Применение и подтверждение изменений:
vesbc(config-esbc-cdr-res-ftp)# do commit
Configuration has been successfully applied and saved to flash. Commit timer started, changes
will be reverted in 600 seconds.
```

```
vesbc(config-esbc-cdr-res-ftp)# do confirm
Configuration has been confirmed. Commit timer canceled.
```

Если отправка записи CDR на основной FTP-сервер (192.168.23.100) по какой-либо причине не произойдет, то она попытается отправиться на резервный FTP-сервер (192.168.23.200), в случае неудачи и на резервном, запись сохранится только локально.

Пример записи файла CDR

На ESBC настроена следующая конфигурация CDR:

```
cdr
  enable
  add-header
  signature test_signature
  period minutes 1
  local
    save
    path flash:cdr/
    create-directories by-date
    keep days 1
  exit
exit
```

После совершения успешного вызова длительностью 3 секунды, с номера 24001 на номер 23001 будет сформирован файл CDR (через 1 минуту) вида:

```
1 vesbc CDR. File started at '20250806124432'
2 test_signature
3 2025-08-06 12:44:16;2025-08-06 12:44:16;24001;24001;23002;23002;UAC;UAS;000003;016;1;1;1-1120206@192.168.23.200;c06ae58b882019de18c8a99e4b530794;;192.168.23.200;192.168.23.200;;192.168.23.200;192.168.23.200;192.168.23.200;
4
```

9.21 Работа с логами

Логирование ESBC осуществляется с помощью syslog. Более подробно настройки syslog описаны в разделе Управление SYSLOG справочника команд CLI.

По умолчанию логирование модулей ESBC выключено.

> Включение логирования всех модулей при большой вызывной нагрузке может повлиять на производительность системы. Наибольшее влияние на производительность оказывает вывод логов в консоль (syslog console).

Модули, входящие в состав ESBC

Название	Описание	Назначение
esbc_core	модуль основной логики	обработка вызовов, отвечает за маршрутизацию вызовов, обеспечивает взаимодействие остальных модулей
esbc_sip_balancer	модуль управления подсистемой SIP	получение сообщений SIP (на открытый сокет) и передача их в модуль **esbc_sip_worker**
esbc_sip_worker	модуль расширения подсистемы SIP	адаптер протокола SIP, обрабатывает сообщения и передает данные модулю **esbc_core**

esbc_media_balancer	модуль управления подсистемой media	управление ресурсами в подсистеме media, выделяет RTP-порты и передает их в модуль **esbc_media_worker**
esbc_media_worker	модуль расширения подсистемы media	обработка медиапотоков (RTP)
esbc_config_manager	адаптер базы данных конфигурации	хранение конфигурации системы
esbc_access_mediator	модуль внешнего доступа	обработка внешних взаимодействий с системой CLI
esbc_ipc	брокер сообщений	обеспечение связи всех модулей в системе
esbc_dispatcher	модуль контроля состояния модулей	контроль модулей, индикация об изменении состояний модулей
esbc_sm	модуль управления абонентскими записями	добавление/удаление записей о регистрации абонентов, добавление/удаление/изменение контактов регистрации, хранение и восстановление записей из базы, предоставление информации о записях и контактах абонентов другим модулям системы
esbc_voip_guard	модуль fail2ban	отслеживает попытки обращения к сервису телефонии, при обнаружении постоянно повторяющихся неудачных попыток обращения с одного и того же IP-адреса или хоста, модуль блокирует попытки с этого IP-адреса/хоста
esbc_sysio	модуль взаимодействия с ОС	служит прослойкой между ESBC и ОС, на которой он разворачивается, предоставляет единый интерфейс взаимодействия с системой и реализует мониторинг различных системных событий
esbc_mon	модуль мониторинга	обеспечение функции мониторинга и сбора статистики
esbc_aaa	модуль aaa	аутентификация, хранение информации о вызовах

Включение логирования модулей ESBC производится в разделе debug:

```
vesbc#

#Переход в раздел debug:
vesbc# debug
vesbc(debug)#

#Включение логирования модуля esbc_dispatcher:
vesbc(debug)# debug esbc disp

#Включение логирования модуля esbc_config_manager:
vesbc(debug)# debug esbc cfgmgr

#Включение логирования модуля esbc_access_mediator:
vesbc(debug)# debug esbc accmed

#Включение логирования модуля esbc_mon:
vesbc(debug)# debug esbc mon

#Включение логирования модуля esbc_aaa:
vesbc(debug)# debug esbc aaa

#Включение логирования модуля esbc_core:
vesbc(debug)# debug esbc core

#Включение логирования модуля esbc_sip_balancer:
vesbc(debug)# debug esbc sipbl

#Включение логирования модуля esbc_sip_worker:
vesbc(debug)# debug esbc sipwrk

#Включение логирования модуля esbc_media_balancer:
vesbc(debug)# debug esbc mediabl

#Включение логирования модуля esbc_media_worker:
vesbc(debug)# debug esbc mediawrk

#Включение логирования модуля esbc_sysio:
vesbc(debug)# debug esbc sysio

#Включение логирования модуля esbc_sm:
vesbc(debug)# debug esbc submngr

#Включение логирования модуля esbc_voip_guard:
vesbc(debug)# debug esbc voip-guard
```

Для отключения логирования модулей ESBC используется команда, аналогичная включению, с приставкой **no**:

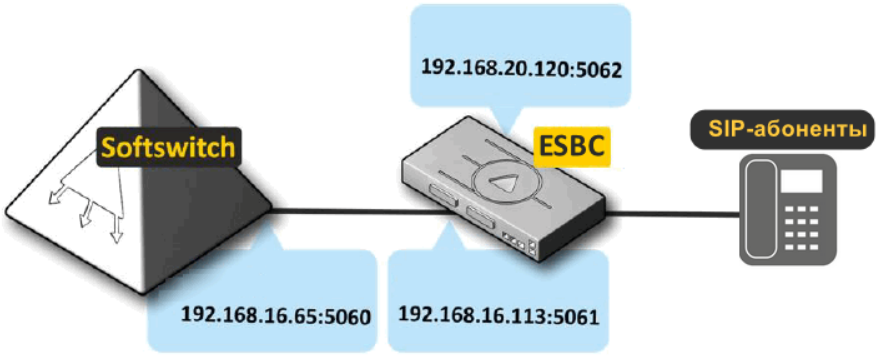
```
#Выключение логирования модуля esbc_voip_guard:
vesbc(debug)# no debug esbc voip-guard
```

Для установки параметров логирования по умолчанию используется команда *no debug all*. Данная команда отключает логирование всех модулей ESBC.

9.22 Примеры настройки ESBC

9.22.1 Настройка для SIP-абонентов

Схема применения:



Описание:

SIP-абоненты (IP-телефон/VoIP шлюз/Мобильный SIP-клиент и т. д.) отправляют сообщение на IP-адрес 192.168.20.120 порт 5062, ESBC пересылает данный трафик с IP-адреса 192.168.16.113 порт 5061 на адрес Softswitch (IP АТС/SIP-proxy и т. д) 192.168.16.65 порт 5060.

Для реализации данной схемы общий алгоритм настройки следующий:

1. Настроить сетевые интерфейсы на ESBC в сторону SSW и в сторону SIP-абонентов.
2. Создать SIP-транспорт в сторону SSW и SIP-абонентов.
3. Создать медиаресурсы для обоих направлений, назначить им диапазон портов для передачи голоса.
4. Создать абонентский интерфейс и SIP-транк.
5. Создать правила, по которым будут маршрутизироваться вызовы от абонентов до SSW.

Порядок конфигурирования ESBC:

1. Настроить IP-адрес на интерфейсе в сторону SSW:

```
vesbc# configure
vesbc(config)# interface gigabitethernet 1/0/1
vesbc(config-if-gi)# description "SSW"
vesbc(config-if-gi)# ip address 192.168.16.113/24
vesbc(config-if-gi)# ip firewall disable
```

2. Настроить IP-адрес на внешнем интерфейсе в сторону абонентов:

```
vesbc# configure
vesbc(config)# interface gigabitethernet 1/0/2
vesbc(config-if-gi)# description "ABONENTS"
vesbc(config-if-gi)# ip address 192.168.20.120/24
```

3. Создать SIP-транспорт в сторону SSW:

```
vesbc#
vesbc# configure
vesbc(config)# esbc
vesbc(config-esbc)# sip transport TRANSPORT_SSW
vesbc(config-esbc-sip-transport)# ip address 192.168.16.113
vesbc(config-esbc-sip-transport)# port 5061
```

4. Создать SIP-транспорт в сторону абонентов:

```
vesbc#
vesbc# configure
vesbc(config)# esbc
vesbc(config-esbc)# sip transport TRANSPORT_ABONENTS
vesbc(config-esbc-sip-transport)# ip address 192.168.20.120
vesbc(config-esbc-sip-transport)# port 5062
```

5. Создать медиаресурсы для согласования и передачи голоса на плече SSW --- ESBC:

```
vesbc#
vesbc# configure
vesbc(config)# esbc
vesbc(config-esbc)# media resource MEDIA_SSW
vesbc(config-esbc-media-resource)# ip address 192.168.16.113

#Указать диапазон портов, который будет выделяться на ESBC для передачи голоса. Данная команда
необязательная. Если ее не указывать, будет использоваться диапазон портов 8000-65535.
vesbc(config-esbc-media-resource)# port-range 1024-65535
```

6. Создать медиаресурсы для согласования и передачи голоса на плече ESBC --- Абонентский шлюз/SIP-абоненты:

```
vesbc#
vesbc# configure
vesbc(config)# esbc
vesbc(config-esbc)# media resource MEDIA_ABONENTS
vesbc(config-esbc-media-resource)# ip address 192.168.20.120
```

7. Создать SIP-транк в сторону SSW:

```
vesbc#
vesbc# configure
vesbc(config)# esbc
vesbc(config-esbc)# trunk sip TRUNK_SSW
vesbc(config-esbc-trunk-sip)# sip transport TRANSPORT_SSW
vesbc(config-esbc-trunk-sip)# remote address 192.168.16.65
vesbc(config-esbc-trunk-sip)# remote port 5060
vesbc(config-esbc-trunk-sip)# media resource 0 MEDIA_SSW
```

8. Создать абонентский интерфейс в сторону абонентов:

```
vesbc#
vesbc# configure
vesbc(config)# esbc
vesbc(config-esbc)# user-interface sip ABONENTS
vesbc(config-esbc-user-interface-sip)# sip transport TRANSPORT_ABONENTS
vesbc(config-esbc-user-interface-sip)# media resource 0 MEDIA_ABONENTS

#Если абоненты находятся за NAT выполнить команду:
vesbc(config-esbc-user-interface-sip)# nat comedia-mode on
```

9. Создать таблицу маршрутизации и добавить туда правила, по которым вызовы, приходящие с абонентов будут маршрутизироваться на SSW:

```
vesbc#
vesbc# configure
vesbc(config)# esbc
vesbc(config-esbc)# route-table TO_SSW
vesbc(config-esbc-route-table)# rule 0
vesbc(config-esbc-route-table-rule)# action direct-to-trunk TRUNK_SSW
```

10. Привязать созданную таблицу маршрутизации к абонентскому интерфейсу:

```
vesbc#
vesbc# configure
vesbc(config)# esbc
vesbc(config-esbc)# user-interface sip ABONENTS
vesbc(config-esbc-user-interface-sip)# route-table TO_SSW
```

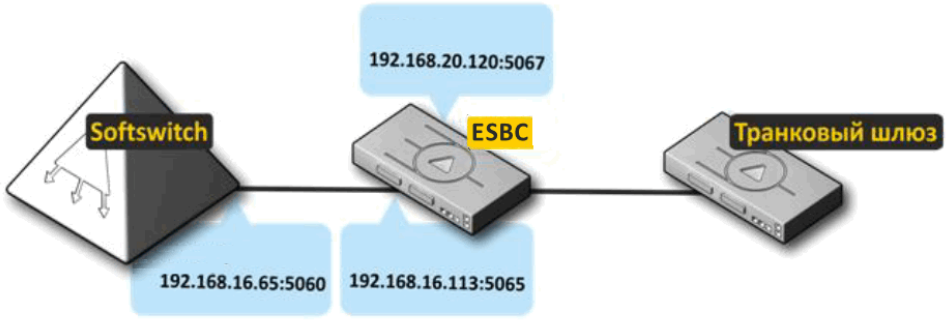
11. Применить конфигурацию и подтвердить изменения:

```
vesbc# commit
vesbc# confirm
```

⚠ В приведённой схеме описаны базовые настройки.

9.22.2 Настройка для SIP-транков

Схема применения:



Описание:

Транковый шлюз (IP АТС/ SIP-proxy/Удаленный SSW и др.) отправляет сообщения с IP-адреса 192.168.20.99 порта 5060 на IP-адрес 192.168.20.120 порт 5067, ESBC пересылает данный трафик с IP-адреса 192.168.16.113 порта 5065 на адрес Softswitch 192.168.16.65 порт 5060. И в обратную сторону SSW отправляет сообщения с IP-адреса 192.168.16.65 порта 5060 на IP-адрес 192.168.16.113 порт 5065, ESBC пересылает данный трафик с IP-адреса 192.168.20.120 порта 5067 на адрес транкового шлюза 192.168.20.99 порт 5060.

Для реализации данной схемы общий алгоритм настройки следующий:

1. Настроить сетевые интерфейсы на ESBC в сторону SSW и в сторону транкового шлюза.
2. Создать SIP-транспорт в сторону SSW и транкового шлюза.
3. Создать медиаресурсы для обоих направлений, назначить им диапазон портов для передачи голоса.
4. Создать 2 SIP-транка в сторону SSW и в сторону транкового шлюза.
5. Создать правила, по которым будут маршрутизироваться вызовы от транкового шлюза до SSW и наоборот от SSW до транкового шлюза.

Порядок конфигурирования ESBC:

1. Настроить IP-адрес на интерфейсе в сторону SSW:

```
vesbc# configure
vesbc(config)# interface gigabitethernet 1/0/1
vesbc(config-if-gi)# description "SSW"
vesbc(config-if-gi)# ip address 192.168.16.113/24
```

2. Настроить IP-адрес на интерфейсе в сторону транкового шлюза:

```
vesbc# configure
vesbc(config)# interface gigabitethernet 1/0/2
vesbc(config-if-gi)# description "TRUNK_GATEWAY"
vesbc(config-if-gi)# ip address 192.168.20.120/24
```

3. Создать SIP-транспорт в сторону SSW:

```
vesbc#
vesbc# configure
vesbc(config)# esbc
vesbc(config-esbc)# sip transport TRANSPORT_SSW
vesbc(config-esbc-sip-transport)# ip address 192.168.16.113
vesbc(config-esbc-sip-transport)# port 5065
```

4. Создать SIP-транспорт в сторону транкового шлюза:

```
vesbc#
vesbc# configure
vesbc(config)# esbc
vesbc(config-esbc)# sip transport TRANSPORT_TRUNK_GATEWAY
vesbc(config-esbc-sip-transport)# ip address 192.168.20.120
vesbc(config-esbc-sip-transport)# port 5067
```

5. Создать медиаресурсы для согласования и передачи голоса на плече SSW --- ESBC:

```
vesbc#
vesbc# configure
vesbc(config)# esbc
vesbc(config-esbc)# media resource MEDIA_SSW
vesbc(config-esbc-media-resource)# ip address 192.168.16.113

# Указать диапазон портов, который будет выделяться на ESBC для передачи голоса. Данная команда
необязательная. Если ее не указывать, будет использоваться диапазон портов 8000-65535.
vesbc(config-esbc-media-resource)# port-range 1024-65535
```

6. Создать медиаресурсы для согласования и передачи голоса на плече ESBC --- Транковый шлюз:

```
vesbc#
vesbc# configure
vesbc(config)# esbc
vesbc(config-esbc)# mediaresource MEDIA_TRUNK_GATEWAY
vesbc(config-esbc-media-resource)# ip address 192.168.20.120
```

7. Создать SIP-транк в сторону SSW:

```
vesbc#
vesbc# configure
vesbc(config)# esbc
vesbc(config-esbc)# trunk sip TRUNK_SSW
vesbc(config-esbc-trunk-sip)# sip transport TRANSPORT_SSW
vesbc(config-esbc-trunk-sip)# remote address 192.168.16.65
vesbc(config-esbc-trunk-sip)# remote port 5060
vesbc(config-esbc-trunk-sip)# media resource 0 MEDIA_SSW
```

8. Создать SIP-транк в сторону транкового шлюза:

```
vesbc#
vesbc# configure
vesbc(config)# esbc
vesbc(config-esbc)# trunk sip TRUNK_GATEWAY
vesbc(config-esbc-trunk-sip)# sip transport TRANSPORT_TRUNK_GATEWAY
vesbc(config-esbc-trunk-sip)# remote address 192.168.20.99
vesbc(config-esbc-trunk-sip)# remote port 5060
vesbc(config-esbc-trunk-sip)# media resource 0 MEDIA_TRUNK_GATEWAY
```

9. Создать таблицу маршрутизации и добавить туда правила, по которым вызовы, приходящие с транкового шлюза будут маршрутизироваться на SSW:

```
vesbc#
vesbc# configure
vesbc(config)# esbc
vesbc(config-esbc)# route-table TO_SSW
vesbc(config-esbc-route-table)# rule 0
vesbc(config-esbc-route-table-rule)# action direct-to-trunk TRUNK_SSW
```

10. Создать таблицу маршрутизации и добавить туда правила, по которым вызовы, приходящие с SSW будут маршрутизироваться на транковый шлюз:

```
vesbc#
vesbc# configure
vesbc(config)# esbc
vesbc(config-esbc)# route-table TO_TRUNK_GATEWAY
vesbc(config-esbc-route-table)# rule 0
vesbc(config-esbc-route-table-rule)# action direct-to-trunk TRUNK_GATEWAY
```

11. Привязать созданные таблицы маршрутизации к транкам:

```
vesbc#
vesbc# configure
vesbc(config)# esbc
vesbc(config-esbc)# trunk sip TRUNK_SSW
vesbc(config-esbc-trunk-sip)# route-table TO_TRUNK_GATEWAY
vesbc(config-esbc-trunk-sip)# exit
vesbc(config-esbc)# trunk sip TRUNK_GATEWAY
vesbc(config-esbc-trunk-sip)# route-table TO_SSW
```

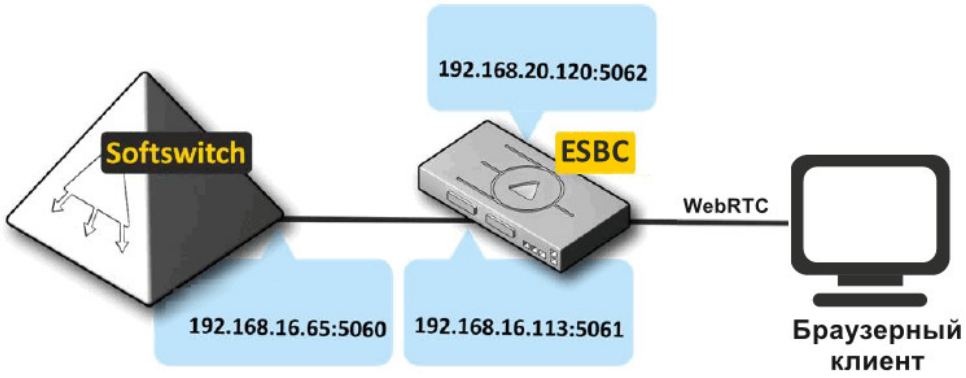
12. Применить конфигурацию и подтвердить изменения:

```
vesbc# commit
vesbc# confirm
```

⚠ В приведённой схеме описаны базовые настройки.

9.22.3 Настройка для SIP-абонентов, использующих WebRTC

Схема применения:



Описание:

SIP-абоненты (WEB, Desktop-клиенты) отправляют сообщения на IP-адрес 192.168.20.120 порт 5062 с помощью WebSocket Secure, ESBC отправляет по TCP данный трафик с IP-адреса 192.168.16.113 порт 5061 на адрес Softswitch (IP АТС/SIP-proxy и т. д) 192.168.16.65 порт 5060.

Для реализации данной схемы общий алгоритм настройки следующий:

1. Настроить сетевые интерфейсы на ESBC в сторону SSW и в сторону SIP-абонентов.
2. Создать SIP-транспорт в режиме TCP (only/prefer) в сторону SSW и SIP-транспорт в режиме WSS для SIP-абонентов.
3. Создать медиаресурсы для обоих направлений, назначить им диапазон портов для передачи голоса.
4. Создать медиапрофиль для SIP-абонентов и включить на нём шифрование DTLS-SRTP.
5. Создать абонентский интерфейс и SIP-транк.
6. Создать правила, по которым будут маршрутизироваться вызовы от абонентов до SSW.

Порядок конфигурирования ESBC:

1. Настроить IP-адрес на интерфейсе в сторону SSW:

```
vesbc# configure
vesbc(config)# interface gigabitethernet 1/0/1
vesbc(config-if-gi)# description "SSW"
vesbc(config-if-gi)# ip address 192.168.16.113/24
vesbc(config-if-gi)# ip firewall disable
```

2. Настроить IP-адрес на внешнем интерфейсе в сторону абонентов:

```
vesbc# configure
vesbc(config)# interface gigabitethernet 1/0/2
vesbc(config-if-gi)# description "ABONENTS"
vesbc(config-if-gi)# ip address 192.168.20.120/24
```

3. Создать SIP-транспорт в сторону SSW:

```
vesbc#
vesbc# configure
vesbc(config)# esbc
vesbc(config-esbc)# sip transport TRANSPORT_SSW
vesbc(config-esbc-sip-transport)# ip address 192.168.16.113
vesbc(config-esbc-sip-transport)# mode tcp-prefer
vesbc(config-esbc-sip-transport)# port 5061
```

4. Создать SIP-транспорт в сторону абонентов:

```
vesbc#
vesbc# configure
vesbc(config)# esbc
vesbc(config-esbc)# sip transport TRANSPORT_ABONENTS
vesbc(config-esbc-sip-transport)# ip address 192.168.20.120
vesbc(config-esbc-sip-transport)# mode wss
vesbc(config-esbc-sip-transport)# port 5062
```

⚠ Если абоненты используют WebSocket, а не WebSocket Secure, то необходимо выбрать **mode ws** в настройках SIP-транспорта для абонентов.

5. Создать медиаресурсы для согласования и передачи голоса на плече SSW --- ESBC:

```
vesbc#
vesbc# configure
vesbc(config)# esbc
vesbc(config-esbc)# media resource MEDIA_SSW
vesbc(config-esbc-media-resource)# ip address 192.168.16.113

#Указать диапазон портов, который будет выделяться на ESBC для передачи голоса. Данная команда
необязательная. Если ее не указывать, будет использоваться диапазон портов 8000-65535.
vesbc(config-esbc-media-resource)# port-range 1024-65535
```

6. Создать медиаресурсы для согласования и передачи голоса на плече ESBC --- Абонентский шлюз/SIP-абоненты:

```
vesbc#
vesbc# configure
vesbc(config)# esbc
vesbc(config-esbc)# media resource MEDIA_ABONENTS
vesbc(config-esbc-media-resource)# ip address 192.168.20.120
```

7. Создать медиапрофиль с шифрованием DTLS-SRTP для SIP-абонентов:

```
vesbc#
vesbc# configure
vesbc(config)# esbc
vesbc(config-esbc)# media profile MEDIA_PROFILE_ABONENTS
vesbc(config-esbc-media-profile)# srtp mode mandatory
vesbc(config-esbc-media-profile)# srtp keying dtls-srtp
```

8. Создать SIP-транк в сторону SSW:

```
vesbc#
vesbc# configure
vesbc(config)# esbc
vesbc(config-esbc)# trunk sip TRUNK_SSW
vesbc(config-esbc-trunk-sip)# sip transport TRANSPORT_SSW
vesbc(config-esbc-trunk-sip)# remote address 192.168.16.65
vesbc(config-esbc-trunk-sip)# remote port 5060
vesbc(config-esbc-trunk-sip)# media resource 0 MEDIA_SSW
```

9. Создать абонентский интерфейс в сторону абонентов:

```
vesbc#
vesbc# configure
vesbc(config)# esbc
vesbc(config-esbc)# user-interface sip ABONENTS
vesbc(config-esbc-user-interface-sip)# sip transport TRANSPORT_ABONENTS
vesbc(config-esbc-user-interface-sip)# media resource 0 MEDIA_ABONENTS
vesbc(config-esbc-user-interface-sip)# media profile MEDIA_PROFILE_ABONENTS

#Если абоненты находятся за NAT, выполнить команду:
vesbc(config-esbc-user-interface-sip)# nat comedia-mode on
```

10. Создать таблицу маршрутизации и добавить туда правила, по которым вызовы, приходящие с абонентов, будут маршрутизироваться на SSW:

```
vesbc#
vesbc# configure
vesbc(config)# esbc
vesbc(config-esbc)# route-table TO_SSW
vesbc(config-esbc-route-table)# rule 0
vesbc(config-esbc-route-table-rule)# action direct-to-trunk TRUNK_SSW
```

11. Привязать созданную таблицу маршрутизации к абонентскому интерфейсу:

```
vesbc#
vesbc# configure
vesbc(config)# esbc
vesbc(config-esbc)# user-interface sip ABONENTS
vesbc(config-esbc-user-interface-sip)# route-table TO_SSW
```

12. Применить конфигурацию и подтвердить изменения:

```
vesbc# commit
vesbc# confirm
```

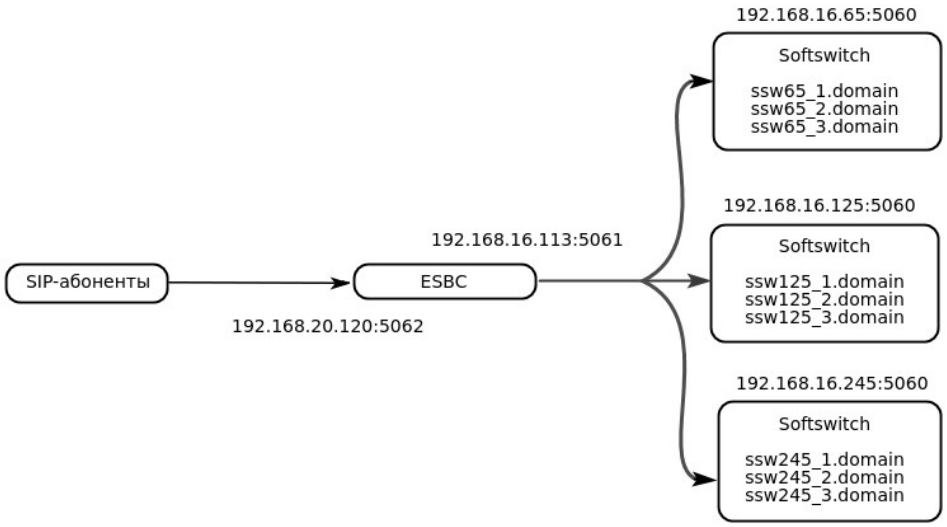
⚠ В приведённой схеме описаны базовые настройки.

9.22.4 Настройка динамического режима для SIP-транков

При использовании динамического режима фактический адрес назначения при вызове в транк определяется внешним сервисом.

⚠ В текущей версии ПО поддержана работа только с DNS в качестве внешнего сервиса.

Схема применения:



Описание:

SIP-абоненты (IP-телефон/VoIP-шлюз/Мобильный SIP-клиент и т. д.) отправляют SIP-запросы на IP-адрес 192.168.20.120 порт 5062.
ESBC должен смаршрутизировать запрос в зависимости от домена в hostname части RURI, полученного от абонента. Вызовы/регистрации могут быть смаршрутизированы на один из трёх Softswitch (IP АТС/ SIP-proxy и т. д), на каждом из них настроено несколько доменов.

Для реализации данной схемы общий алгоритм настройки следующий:

1. Настроить сетевые интерфейсы на ESBC в сторону SSW и в сторону SIP-абонентов.
2. Настроить DNS-сервер.
3. Создать SIP-транспорт в сторону SSW и SIP-транспорт для SIP-абонентов.
4. Создать медиаресурсы для обоих направлений, назначить им диапазон портов для передачи голоса.
5. Создать абонентский интерфейс и SIP-транк в динамическом режиме.
6. Создать правила, по которым будут маршрутизироваться вызовы от абонентов до SSW.

Порядок конфигурирования ESBC:

1. Настроить адрес внешнего сервиса, пример с DNS:

```
vesbc(config)# domain lookup enable
vesbc(config)# domain nameserver 192.168.20.100
vesbc(config)#
```

2. Настроить IP-адрес на интерфейсе в сторону SSW:

```
vesbc# configure
vesbc(config)# interface gigabitethernet 1/0/1
vesbc(config-if-gi)# description "SSW"
vesbc(config-if-gi)# ip address 192.168.16.113/24
vesbc(config-if-gi)# ip firewall disable
```

3. Настроить IP-адрес на внешнем интерфейсе в сторону абонентов:

```
vesbc# configure
vesbc(config)# interface gigabitethernet 1/0/2
vesbc(config-if-gi)# description "ABONENTS"
vesbc(config-if-gi)# ip address 192.168.20.120/24
```

4. Создать SIP-транспорт в сторону SSW:

```
vesbc#
vesbc# configure
vesbc(config)# esbc
vesbc(config-esbc)# sip transport TRANSPORT_SSW
vesbc(config-esbc-sip-transport)# ip address 192.168.16.113
vesbc(config-esbc-sip-transport)# port 5061
```

5. Создать SIP-транспорт в сторону абонентов:

```
vesbc#
vesbc# configure
vesbc(config)# esbc
vesbc(config-esbc)# sip transport TRANSPORT_ABONENTS
vesbc(config-esbc-sip-transport)# ip address 192.168.20.120
vesbc(config-esbc-sip-transport)# port 5062
```

6. Создать медиаресурсы для согласования и передачи голоса на плече SSW --- ESBC:

```
vesbc#
vesbc# configure
vesbc(config)# esbc
vesbc(config-esbc)# media resource MEDIA_SSW
vesbc(config-esbc-media-resource)# ip address 192.168.16.113

#Указать диапазон портов, который будет выделяться на ESBC для передачи голоса. Данная команда
необязательная. Если ее не указывать, будет использоваться диапазон портов 8000-65535.
vesbc(config-esbc-media-resource)# port-range 1024-65535
```

7. Создать медиаресурсы для согласования и передачи голоса на плече ESBC --- Абонентский шлюз/SIP-абоненты:

```
vesbc#
vesbc# configure
vesbc(config)# esbc
vesbc(config-esbc)# media resource MEDIA_ABONENTS
vesbc(config-esbc-media-resource)# ip address 192.168.20.120
```

8. Создать динамический SIP-транк, в качестве адреса указать подсеть, в которой находятся SSW:

```
vesbc#
vesbc# configure
vesbc(config)# esbc
vesbc(config-esbc)# trunk sip SSW_DYNAMIC
vesbc(config-esbc-trunk-sip)# sip transport TRANSPORT_SSW
vesbc(config-esbc-trunk-sip)# remote address 192.168.16.0/24
vesbc(config-esbc-trunk-sip)# remote port 5060
vesbc(config-esbc-trunk-sip)# dynamic-mode dns
vesbc(config-esbc-trunk-sip)# media resource 0 MEDIA_SSW
```

9. Создать абонентский интерфейс в сторону абонентов:

```
vesbc#
vesbc# configure
vesbc(config)# esbc
vesbc(config-esbc)# user-interface sip ABONENTS
vesbc(config-esbc-trunk-sip)# sip transport TRANSPORT_ABONENTS
vesbc(config-esbc-trunk-sip)# media resource 0 MEDIA_ABONENTS

#Если абоненты находятся за NAT, выполнить команду:
vesbc(config-esbc-user-interface-sip)# nat comedia-mode on
```

10. Создать таблицу маршрутизации и добавить туда правила, по которым вызовы, приходящие с абонентов будут маршрутизироваться на SIP-транк с динамическим режимом:

```
vesbc#
vesbc# configure
vesbc(config)# esbc
vesbc(config-esbc)# route-table TO_SSW
vesbc(config-esbc-route-table)# rule 0
vesbc(config-esbc-route-table-rule)# action direct-to-trunk SSW_DYNAMIC
```

11. Привязать созданную таблицу маршрутизации к абонентскому интерфейсу:

```
vesbc#
vesbc# configure
vesbc(config)# esbc
vesbc(config-esbc)# user-interface sip ABONENTS
vesbc(config-esbc-user-interface-sip)# route-table TO_SSW
```

12. Применить конфигурацию и подтвердить изменения:

```
vesbc# commit
vesbc# confirm
```

Пример обработки запроса:

SIP-абонент отправляет сообщение REGISTER, в hostname RURI указывает **ssw125_2.domain.** ESBC отправляет запрос на DNS-сервер (192.168.20.100) для определения адреса назначения, внешний сервис в ответ присылает адрес SIP-сервера (192.168.16.125), на который нужно отправить запрос. ESBC отправляет регистрацию на указанный адрес, подставляя в заголовки To и From **ssw125_2.domain**, последующие запросы с этого абонента при указании того же домена будут отправляться в транк 192.168.16.125:5060 без предварительного обращения к внешнему сервису.

Для определения входящего вызова из транка с динамическим режимом используется адрес/маска подсети из **remote address** и порт/диапазон портов из **remote port**.

⚠ В приведённой схеме описаны базовые настройки.

10 Управление интерфейсами

Алгоритм и примеры настройки функций управления интерфейсами см. в документации ESR.

⚠ Значения параметров для ESBC-3200 идентичны значениям для ESR-3200.

11 Управление туннелированием

Алгоритм и примеры настройки функций управления туннелированием см. в документации ESR.

⚠ Значения параметров для ESBC-3200 идентичны значениям для ESR-3200.

12 Управление функциями второго уровня (L2)

Алгоритм и примеры настройки управления функциями второго уровня (L2) см. в документации ESR.

⚠ Значения параметров для ESBC-3200 идентичны значениям для ESR-3200.

13 Управление QoS

Управление технологией Quality of Service (QoS) см. в документации ESR.

⚠ Значения параметров для ESBC-3200 идентичны значениям для ESR-3200.

14 Управление маршрутизацией

Алгоритм и примеры настройки функций управления маршрутизацией см. в документации ESR.

⚠ Значения параметров для ESBC-3200 идентичны значениям для ESR-3200.

15 Управление технологией MPLS

Управление технологией MPLS описано в документации ESR.

⚠ Значения параметров для ESBC-3200 идентичны значениям для ESR-3200.

16 Управление безопасностью

Алгоритм и примеры настройки функций управления безопасностью см. в документации ESR.

⚠ Значения параметров для ESBC-3200 идентичны значениям для ESR-3200.

17 Управление резервированием

Алгоритм настройки резервирования см. в документации ESR.

⚠ Значения параметров для ESBC-3200 идентичны значениям для ESR-3200.

18 Управление кластеризацией

Кластер организуется из двух одинаковых устройств. Работает в режиме Active-Standby, т. е. на Active запущены все модули ESBC, и он занимается обработкой сигнальных сообщений SIP и медиаданных (RTP-потоков). Устройство Standby не обрабатывает сигнальные сообщений SIP и медиаданные.

Резервирование соединения осуществляется протоколом VRRP.

Конфигурация, файлы ПО, зарегистрированные абоненты синхронизируются между устройствами в реальном времени. В случае обрыва соединения или отключения Active устройства, все существующие вызовы будут разрушены.

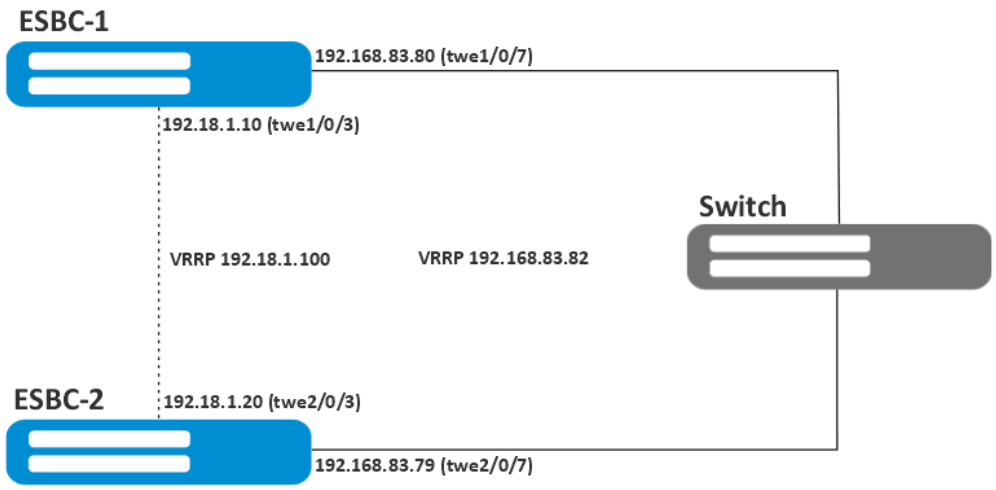
⚠ Время до начала обработки вызовов при failover зависит от вызывной нагрузки на устройство и составляет от 2 до 12 секунд.

Более подробное описание настройки кластера приведено в документации ESR. Ниже представлен пример настройки ESBC-3200 для обработки вызовов.

⚠ Значения параметров для ESBC-3200 идентичны значениям для ESR-3200.

Пример настройки HA кластера ESBC

Схема:



⚠ Настроить cluster можно двумя способами:
1) Настроить каждый unit отдельно;
2) Настроить один unit, а затем второй включить в cluster по ZTP.
Ниже приведён пример ручной настройки, настройка с ZTP описана в документации ESR.

18.1 Первичная настройка кластера

После включения устройства необходимо применить конфигурацию по умолчанию на устройствах, предназначенных для объединения в кластер:

ESBC-1,2

```
ESBC-3200# copy system:default-config system:candidate-config

Entire candidate configuration will be reset to default, all settings will be lost upon commit.

Do you really want to continue? (y/N): y

|****************************************| 100% (59B) Default configuration loaded
successfully.
```

Для более удобного и ясного восприятия рекомендуется переименовать устройства. В кластерной версии прошивки предусмотрена возможность указать имя устройства с привязкой к юниту. Устройство будет использовать только тот hostname, юнитом которого он является:

ESBC-1,2

```
ESBC-3200# configure
ESBC-3200(config)# hostname ESBC-1 unit 1
ESBC-3200(config)# hostname ESBC-2 unit 2
```

Чтобы изменить юнит устройства, выполните следующие команды:

ESBC-1

```
ESBC-1# set unit id 1
Unit ID will be 1 after reboot
ESBC-1# reload system
Do you really want to reload system now? (y/N): y
```

ESBC-2

```
ESBC-2# set unit id 2
Unit ID will be 2 after reboot
ESBC-2# reload system
Do you really want to reload system now? (y/N): y
```

Убедитесь в том, что настройки юнитов применились успешно:

ESBC-1

```
ESBC-1# show unit id
Unit ID is 1
Unit ID will be 1 after reboot
```

ESBC-2

```
ESBC-2# show unit id
Unit ID is 2
Unit ID will be 2 after reboot
```

В текущей схеме служебная информация по управлению кластером будет передаваться через выделенный линк синхронизации между интерфейсами twe1/0/3 и twe2/0/3.

ESBC-1,2

```
ESBC-1(config)# interface twentyfivegigabitethernet 1/0/3
ESBC-1(config-if-twe)# description "Network: SYNC"
ESBC-1(config-if-twe)# mode switchport
ESBC-1(config-if-twe)# exit
ESBC-1(config)# interface twentyfivegigabitethernet 2/0/3
ESBC-1(config-if-twe)# description "Network: SYNC"
ESBC-1(config-if-twe)# mode switchport
ESBC-1(config-if-twe)# exit
```

18.2 Настройка внешних сетевых интерфейсов

На обоих устройствах необходимо настроить IP-адрес и VRRP на внешних интерфейсах. В текущей схеме это интерфейсы twe1/0/7 и twe2/0/7.

ESBC-1,2

```
ESBC-1(config)# interface twentyfivegigabitethernet 1/0/7
ESBC-1(config-if-twe)# ip address 192.168.83.80/22
ESBC-1(config-if-twe)# vrrp
ESBC-1(config-if-twe)# vrrp id 10
ESBC-1(config-if-twe)# vrrp ip 192.168.83.82/22
ESBC-1(config-if-twe)# vrrp group 2
ESBC-1(config-if-twe)# exit
ESBC-1(config)# interface twentyfivegigabitethernet 2/0/7
ESBC-1(config-if-twe)# ip address 192.168.83.79/22
ESBC-1(config-if-twe)# vrrp
ESBC-1(config-if-twe)# vrrp id 10
ESBC-1(config-if-twe)# vrrp ip 192.168.83.82/22
ESBC-1(config-if-twe)# vrrp group 2
ESBC-1(config-if-twe)# exit
```

18.3 Настройка кластерного интерфейса

Для полноценной работы кластера требуется сконфигурировать кластерный интерфейс, который будет использоваться для передачи control plane трафика, необходимого для полноценного функционирования кластера. В качестве кластерного интерфейса назначен bridge. В качестве механизма, отвечающего за определение ролей устройств, участвующих в резервировании, назначен протокол VRRP. Настройки cluster-интерфейса должны быть идентичны для всех участников кластера. Так как кластер выполняет синхронизацию состояний между устройствами, необходимо создать зону безопасности SYNC (synchronization) и разрешить прохождение трафика протокола vrrp:

ESBC-1,2

```
ESBC-1(config)# security zone SYNC
ESBC-1(config-zone)# exit
ESBC-1(config)#
ESBC-1(config)# security zone-pair SYNC self
ESBC-1(config-zone-pair)# rule 1
ESBC-1(config-zone-pair-rule)# action permit
ESBC-1(config-zone-pair-rule)# match protocol vrrp
ESBC-1(config-zone-pair-rule)# enable
ESBC-1(config-zone-pair-rule)# exit
ESBC-1(config-zone-pair)# exit
```

Далее перейдите к настройкам кластерного интерфейса:

ESBC-1,2

```
ESBC-1# configure
ESBC-1(config)# bridge 1
ESBC-1(config-bridge)# vlan 1
ESBC-1(config-bridge)# security-zone SYNC
ESBC-1(config-bridge)# ip address 192.18.1.10/24 unit 1
ESBC-1(config-bridge)# ip address 192.18.1.20/24 unit 2
ESBC-1(config-bridge)# vrrp id 1
ESBC-1(config-bridge)# vrrp group 2
ESBC-1(config-bridge)# vrrp ip 192.18.1.100/24
ESBC-1(config-bridge)# vrrp
ESBC-1(config-bridge)# enable
```

18.4 Настройка кластера

Для запуска кластера нужно только указать заранее настроенный кластерный интерфейс и юниты, которые будут выполнять роли Active и Standby.

Перейдите в настройку кластера:

ESBC-1,2

```
ESBC-1# configure
ESBC-1(config)# cluster
ESBC-1(config-cluster)# unit 1
ESBC-1(config-cluster-unit)# mac-address 68:13:e2:e1:28:90
ESBC-1(config-cluster-unit)# exit
ESBC-1(config-cluster)# unit 2
ESBC-1(config-cluster-unit)# mac-address 68:13:e2:e1:25:30
ESBC-1(config-cluster-unit)# exit
```

⚠ В качестве mac-address указывается системный MAC-адрес устройства, его можно узнать с помощью команды show system | include MAC.

Укажите кластерный интерфейс, созданный ранее, и активируйте кластер:

ESBC-1,2

```
ESBC-1(config-cluster)# cluster-interface bridge 1
ESBC-1(config-cluster)# enable
```

После выполнения этих шагов кластер будет успешно запущен. Текущее состояние кластера можно узнать, выполнив команду:

ESBC-1

```
ESBC-1# show cluster status
Unit   Hostname              Role         MAC address         State            IP address
----   --------------------  ----------   -----------------   --------------   ---------------
1*     ESBC-1                Active       68:13:e2:e1:28:90   Joined           192.18.1.10
2      ESBC-2                Standby      68:13:e2:e1:25:30   Joined           192.18.1.20
```

⚠ После включения кластера и установления юнитов в состояние Joined дальнейшая настройка кластера осуществляется путем настройки Active-юнита. Синхронизируются команды конфигурации, а также команды: *commit*, *confirm*, rollback, *restore*, *save*. В случае если конфигурирование осуществляется на Standby, то синхронизации не будет. Есть возможность отключения синхронизации командой *sync config disable*.

Для проверки работы протокола VRRP выполните следующую команду:

ESBC-1

```
ESBC-1# show vrrp
Virtual router   Virtual IP         Priority   Preemption   State    Synchronization group ID
--------------   -----------------  --------   ----------   ------   ------------------------
1                192.18.1.100/24      100        Enabled     Master   2
10               192.168.83.82/22     100        Enabled     Master   2
```

Также можно посмотреть состояние синхронизации различных подсистем в кластере, выполнив команду:

ESBC-1

```
ESBC-1# show cluster sync status
System part               Synced
----------------------    ------
candidate-config          Yes
running-config            Yes
SW version                Yes
licence                   Yes
licence (After reboot)    Yes
date                      Yes
E-SBC version             Yes
```

19 Управление удаленным доступом

Алгоритм и примеры настройки функций управления удаленным доступом см. в документации ESR.

⚠ Значения параметров для ESBC-3200 идентичны значениям для ESR-3200.

20 Управление сервисами

Алгоритм и примеры настройки функций управления сервисами см. в документации ESR.

⚠ Значения параметров для ESBC-3200 идентичны значениям для ESR-3200.

21 Мониторинг

Данный раздел см. в документации ESR.

⚠ Значения параметров для ESBC-3200 идентичны значениям для ESR-3200.

22 Управление BRAS (Broadband Remote Access Server)

Данный раздел см. в документации ESR.

⚠ Значения параметров для ESBC-3200 идентичны значениям для ESR-3200.

23 Управление лицензированием

- Виды лицензий ESBC
  - vESBC
  - ESBC-3200
- Способы получения лицензии
- Статусы лицензий
- ELM
  - Алгоритм работы с сервером ELM
  - Получение лицензии для vESBC через ELM
  - Получение лицензии для ESBC-3200 через ELM
- Загрузка и активация файловой лицензии

23.1 Виды лицензий ESBC

23.1.1 vESBC

Название лицензии	Функционал
ESBC-LIMIT-MAX-CALLS	Ограничение одновременно установленных сессий на ESBC.
ESBC-LIMIT-MAX-CPS	Ограничение количества вызовов в секунду на ESBC.
ESBC-VIRTUAL-LIMIT-NET	Ограничение скорости полосы пропускания виртуального ESBC.
ESBC-VIRTUAL-LIMIT-DEFAULT	Неизменяемый параметр, предоставляется при выдаче любой лицензии vESBC. Увеличивает лимиты RIB для: • BGP до 65000 • OSPF до 500000 • IS-IS до 500000 • RIP до 10000 Активирует учетную запись techsupport для доступа в режим shell.

23.1.2 ESBC-3200

Название лицензии	Функционал
ESBC	Требуется для активации функционала ESBC, поставляется вместе с устройством в заводской комплектации.
ESBC-LIMIT-MAX-CALLS	Ограничение одновременно установленных сессий на ESBC.
ESBC-LIMIT-MAX-CPS	Ограничение количества вызовов в секунду на ESBC.

23.2 Способы получения лицензии

	Online ELM	Offline ELM	File
vESBC	✓	✓	✗
ESBC-3200	✓	✗	✓

23.3 Статусы лицензий

Active	Лицензия активна.
Candidate	Лицензия будет применена после перезагрузки.
Unsupported	Лицензия не поддерживается в рамках текущей версии ПО или вообще не поддерживается устройством.

23.4 ELM

Сервер лицензий Eltex License Manager (далее — ELM), осуществляющего функцию лицензирования программных и аппаратных продуктов компании «Элтекс». ELM используется в процессе активации лицензии и последующей эксплуатации для подтверждения легитимности приобретенного программного обеспечения и предоставления прав на его использование.

Существует 2 варианта работы с ELM:

- *Online ELM* — сервер лицензий расположен в компании «Элтекс». Установка дополнительного ПО не требуется. Центральный сервер лицензий доступен по адресу https://elm.eltex-co.ru:8099, к которому необходимо обеспечить доступ.
- *Offline ELM* — сервер лицензий устанавливается на стороне заказчика. Подходит для эксплуатации в закрытом контуре. Подробная информация об Offline ELM доступна в официальной документации.

23.4.1 Алгоритм работы с сервером ELM

- При штатной работе ESBC обращается к серверу ELM один раз в час для подтверждения статуса лицензии.
- Если при обращении к серверу возникнет ошибка, и ответ не будет получен, то лицензия на системе будет активна в течение 4 часов, при этом частота обращений к серверу увеличится до одного раза в 15 минут.
- Если по истечении 4 часов ESBC так и не получит подтверждения лицензии, то существующая лицензия будет сброшена.
- Если ESBC получил лицензионные параметры от сервера ELM, то при последующих перезагрузках он будет стартовать уже с применёнными параметрами, но должен подтвердить лицензию в течение 15 минут. Обращение к серверу ELM будет сразу после загрузки системы. Если в течение 15 минут ответ от сервера не будет получен, лицензия будет сброшена.

23.4.2 Получение лицензии для vESBC через ELM

✗ При отсутствии подключения vESBC к ELM пропускная способность устройства равна 1 Мбит/с, обработка вызовов отключена.

Для получения лицензии с сервера ELM необходимо настроить serial-number и указать licence-key.

⚠ serial-number и licence-key предоставляются при заказе vESBC.

Шаг 1. Задайте серийный номер:

```
vesbc# set serial-number ESBCXXXXXXX
```

Шаг 2. Перезагрузите устройство:

❌ Серийный номер изменится только после перезагрузки. Не выполняйте дальнейшие шаги до задания серийного номера. После 10 попыток подключения к серверу лицензирования с некорректными учётными данными ваш IP-адрес будет автоматически заблокирован системой защиты сервера лицензирования.

Шаг 3. Настройте подключение к серверу лицензирования:

```
vesbc# configure
vesbc(config)# licence-manager
vesbc(config-licence-manager)# host address elm.eltex-co.ru
vesbc(config-licence-manager)# licence-key ELM-LICENCEKEY
vesbc(config-licence-manager)# enable
vesbc(config-licence-manager)# end
```

❌ Вместо ELM-LICENCEKEY необходимо ввести ключ, полученный при заказе vESBC.

Шаг 4. Примените конфигурацию.

После применения конфигурации и обмена данными с сервером лицензирования станет доступна лицензия, которая расширит возможности вашего устройства.

ℹ Для принудительного запроса к серверу лицензирования можно использовать команду *update licence-manager licence*.

Шаг 5. Используя команду *show licence-manager status*, проверьте статус подключения к ELM-серверу:

```
vesbc# show licence-manager status
ELM server type:                root
Last request status:            success
Last request to licence server: 2025-04-17 10:24:22
Next request to licence server: 2025-04-17 10:24:43
```

Шаг 6. Используя команду *show licence,* проверьте наличие лицензий на устройстве:

```
vesbc# show licence
Feature                        Source     State      Value          Valid from  Expiries
---------------------------   --------   --------   -------------   ----------  -------
ESBC-LIMIT-MAX-CALLS            ELM       Active     50000             --          --
ESBC-LIMIT-MAX-CPS              ELM       Active     1000              --          --
ESBC-VIRTUAL-LIMIT-DEFAULT      ELM       Active     true              --          --
ESBC-VIRTUAL-LIMIT-NET          ELM       Active     10000000000       --          --
```

23.4.3 Получение лицензии для ESBC-3200 через ELM

При отсутствии лицензии на ESBC-3200 обработка вызовов отключена.

Для того чтобы получить лицензию с помощью Eltex Licence Manager, необходимо выполнить следующие шаги:

Шаг 1. Настройте подключение к серверу лицензирования:

```
ESBC-3200# configure
ESBC-3200(config)# licence-manager
ESBC-3200(config-licence-manager)# host address elm.eltex-co.ru
ESBC-3200(config-licence-manager)# enable
ESBC-3200(config-licence-manager)# end
```

Шаг 2. Примените конфигурацию.

После применения конфигурации и обмена данными с сервером лицензирования станет доступна лицензия, которая расширит возможности вашего устройства.

Для принудительного запроса к серверу лицензирования можно использовать команду *update licence-manager licence*.

Шаг 3. Используя команду *show licence-manager status*, проверьте статус подключения к ELM-серверу:

```
ESBC-3200# show licence-manager status
ELM server type:                root
Last request status:            success
Last request to licence server: 2025-04-17 10:24:22
Next request to licence server: 2025-04-17 10:24:43
```

Шаг 4. Используя команду *show licence,* проверьте наличие лицензий на устройстве:

```
ESBC-3200# show licence
Feature                          Source     State          Value     Valid from    Expiries
-------------------------------  --------   -----------    -------   -----------   ---------
ESBC                             Boot       Active         true         --            --
ESBC                             Boot       Candidate      true         --            --
ESBC-LIMIT-MAX-CALLS             ELM        Active         8500         --            --
ESBC-LIMIT-MAX-CPS               ELM        Active         400          --            --
```

23.5 Загрузка и активация файловой лицензии

Загрузка файловой лицензии через web-интерфейс описана в разделе Управление через web-интерфейс настоящего руководства.

Для загрузки лицензии через CLI введите одну из нижеописанных команд. В качестве параметра *<server>* должен быть указан IP-адрес используемого сервера. Для обновления с FTP- или SCP-сервера потребуется ввести имя пользователя (параметр *<user>*) и пароль (параметр *<password>*). В качестве параметра *<file_name>* укажите имя файла лицензии, помещенного на сервер (при использовании SCP нужно указать полный путь − параметр *<folder>*). После ввода команды устройство скопирует файл во внутреннюю память, проверит целостность данных и сохранит его.

TFTP:

```
ESBC-3200# copy tftp://<server>:/<file_name> system:licence
```

FTP:

```
ESBC-3200# copy ftp://[<user>[:<password>]@]<server>:/<file_name> system:licence
```

SCP:

```
ESBC-3200# copy scp://[<user>[:<password>]@]<server>://<folder>/<file_name> system:licence
```

SFTP:

```
ESBC-3200# copy sftp://[<user>[:<password>]@]<server>:/<file_name> system:licence
```

Пример загрузки лицензии через SCP:

```
ESBC-3200# copy scp://adm:password123@192.168.16.168://home/tftp/licence system:licence
|****************************************| 100% (670B) Licence loaded successfully. Please
reboot system to apply changes.
```

Для активации лицензии необходимо перезагрузить устройство:

```
ESBC-3200# reload system
```

После перезагрузки проверьте, что лицензия активирована:

```
Feature                 Source     State         Value    Valid from    Expiries
----------------------  --------   -----------   -------  ------------  --------
ESBC                    Boot       Active        true         --          --
ESBC                    Boot       Candidate     true         --          --
ESBC-LIMIT-MAX-CALLS    File       Active        8500         --          --
ESBC-LIMIT-MAX-CALLS    File       Candidate     8500         --          --
ESBC-LIMIT-MAX-CPS      File       Active        400          --          --
ESBC-LIMIT-MAX-CPS      File       Candidate     400          --          --
```

24 Часто задаваемые вопросы

Не удалось получить маршруты по BGP и/или OSPF, сконфигурированных в VRF. Соседство успешно устанавливается, но в записи маршрутов в RIB отказано %ROUTING-W-KERNEL: Can not install route. Reached the maximum number of BGP routes in the RIB

Необходимо выделить ресурс RIB для VRF, по умолчанию он равен нулю. Делаем это в режиме конфигурирования VRF:

```
esbc(config)# ip vrf <NAME>
esbc(config-vrf)# ip protocols ospf max-routes 12000
esbc(config-vrf)# ip protocols bgp max-routes 1200000
esbc(config-vrf)# end
```

Закрываются сессии SSH/Telnet, проходящие через пограничный контроллер сессий ESBC

Для поддержания сессии активной необходимо настроить передачу keepalive-пакетов. Опция отправки keepalive настраивается в клиенте SSH, например, для клиента PuTTY раздел “Соединение”.

В свою очередь, на пограничном контроллере сессий можно выставить время ожидания до закрытия неактивных сессий TCP (в примере выставлен 1 час):

```
esbc(config)# ip firewall sessions tcp-estabilished-timeout 3600
```

На интерфейсе был отключен firewall (ip firewall disable). После внесения этого интерфейса в security zone, удаления из конфигурации ip firewall disable и применения изменений – доступ для активных сессий с данного порта не закрылся согласно правилам security zone-pair

Изменения в конфигурации Firewall будут действовать только для новых сессий, сброса активных сессий в Firewall не происходит. Отчистить активные сессии в firewall можно командой:

```
esbc# clear ip firewall session
```

Как полностью очистить конфигурацию ESBC и как сбросить на заводскую конфигурацию?

Очистка конфигурации происходит путем копирования пустой конфигурации в candidate-config и применения его в running-config.

```
esbc# copy system:default-config system:candidate-config
```

Процесс сброса на заводскую конфигурацию аналогичен.

```
esbc# copy system:factory-config system:candidate-config
```

В случае невозможности аутентификации на пограничном контроллере сессий (неизвестен логин/пароль) конфигурацию можно сбросить к заводской следующим образом:

1. дождаться полной загрузки устройства
2. зажать функциональную кнопку "F" на 15 секунд
3. отпустить функциональную кнопку "F"
4. дождаться полной загрузки устройства с заводской конфигурацией

Как привязать subinterface к созданным VLAN?

При создании саб-интерфейса VLAN создается и привязывается автоматически (прямая зависимость индекс sub – VID).

```
esbc(config)# interface gigabitethernet 1/0/1.100
```

После применения можно наблюдать информационные сообщения:

```
2016-07-14T012:46:24+00:00 %VLAN: creating VLAN 100
```

Есть ли функционал в пограничном контроллере серии ESBC для анализа трафика?

В пограничных контроллерах сессий серии ESBC реализована возможность анализировать трафик на интерфейсах из CLI. Сниффер запускается командой monitor.

```
esbc# monitor gigabitethernet 1/0/1
```

Как настроить ip prefix-list 0.0.0.0/0?

Ниже приведен пример конфигурации префикс-листа, разрешающего прием маршрута по умолчанию.

```
esbc(config)# ip prefix-list eltex
esbc(config-pl)# permit 0.0.0.0/0
```

Проблема прохождения асинхронного трафика

В случае организации сети с асинхронной маршрутизацией, Firewall будет запрещать "неправильный (ошибочный)" входящий трафик (не открывающий новое соединение и не принадлежащий никакому установленному соединению) из соображений безопасности.

Разрешающее правило в Firewall не решит поставленную задачу для подобных схем.

Решить задачу можно, отключив Firewall на входном интерфейсе:

```
esbc(config-if-gi)# ip firewall disable
```

Как можно сохранить локальную копию конфигурации пограничного контроллера сессий?

Если необходимо скопировать текущую running или candidate – конфигурацию на самом пограничном контроллере сессий – можно воспользоваться командой copy с указанием в качестве источника копирования "system:running-config" или "system:candidate-config", а в качестве назначения – файл в разделе "flash:data/".

```
esbc# copy system:candidate-config flash:data/temp.txt
```

Также существует возможность копирования ранее сохраненных конфигурационных файлов (автоматически из раздела flash:backup/ или вручную из раздела flash:data/) в candidate-конфигурацию:

```
esbc# copy flash:data/temp.txt system:candidate-config
esbc# copy flash:backup/config_20190918_164455 system:candidate-config
```

25 Приложение A. Packet Flow

- Порядок обработки входящего/исходящего трафика сетевыми службами пограничного контроллера сессий ESBC
- Порядок обработки транзитного трафика сетевыми службами пограничного контроллера сессий ESBC

25.1 Порядок обработки входящего/исходящего трафика сетевыми службами пограничного контроллера сессий ESBC

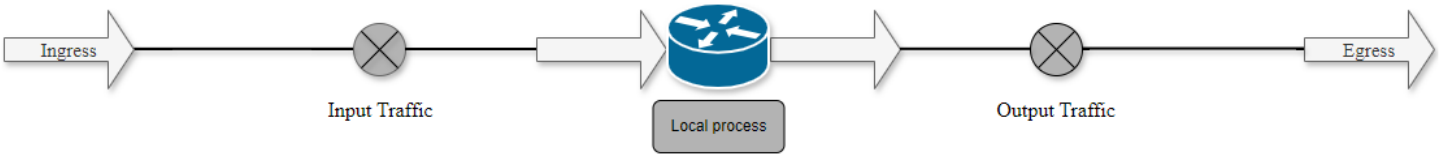


Таблица 1 – Порядок обработки входящего трафика

Шаг	Описание
1	Выполнение функций ACL на входящем трафике
2	Выполнение функций QoS (ограничение полосы пропускания, классификация и т. д.)
3	Выполнение функций DOS defense[1]. На данном этапе выполняются функции защиты от DDOS из разделов firewall screen dos-defense, firewall screen spy-blocking, firewall screen suspicious-packets
4	Инспектирование пакета сервисом IDS/IPS в режиме service-ips monitor[1]
5	Отключение функций Firewall командой ip firewall disable. Разрешение трафика исключает проверки на этапах 6, 13, 15
6	Выполнение правил между зонами any/self
7	Выполнение дефрагментации пакета
8	Выполнение начальных функций BRAS (инициализация соединений, сессий)[1]
9	Выполнение HTTP/HTTPs прокси[1]
10	Функции Destination NAT[1]
11	Routing Decision (FIB)
12	Выполнение функций DOS defense[1]. На этапе данном этапе выполняются специфичные функции защиты от DDOS из разделов firewall screen dos-defense, firewall screen spy-blocking, firewall screen suspicious-packets: ip firewall screen suspicious-packets large-icmp ip firewall screen dos-defense winnuke ip firewall screen spy-blocking port-scan
13	Выполнение правил между пользовательскими зонами / self

14	Разрешение служебного трафика кластера[1]
15	Передача пакета в DPI[1]
16	Передача пакета в Netflow/sFlow (Ingress)[1]
17	Передача пакета в Antispam[1]
18	IPsec (decode)[1]. После выполнения этого шага происходит переход к п.3

Таблица 2 – Порядок обработки исходящего трафика

Шаг	Описание
1	Local Policy Based Routing[1]
2	Route Decision
3	Передача пакета в DPI[1]
4	tcp adjust-mss[1]
5	Netflow/sFlow (Egress)[1]
6	BRAS (для исходящих пакетов)[1]
7	Выполнение функций Source NAT[1]
8	IPsec (encode)[1]
9	Выполнение фрагментации пакетов
10	Выполнение функций QoS (ограничение полосы пропускания, классификация и т. д.)

[1]Данная функция выполняется только при наличии необходимых настроек.

25.2 Порядок обработки транзитного трафика сетевыми службами пограничного контроллера сессий ESBC

Таблица 3 – Порядок обработки транзитного трафика

Шаг	Описание
1	Выполнение функций ACL на входящем трафике
2	Выполнение функций QoS (ограничение полосы пропускания, классификация и т. д.)
3	Выполнение функций DOS defense[1]. На данном этапе выполняются функции защиты от DDOS из раздела firewall screen dos-defense, firewall screen spy-blocking, firewall screen suspicious-packets
4	Отключение функций Firewall командой ip firewall disable. Разрешение трафика исключает проверки на этапах 5, 15, 16
5	Выполнение правил между пользовательскими зонами / any
6	Выполнение дефрагментации пакета
7	Выполнение начальных функций BRAS (инициализация соединений, сессий)[1]
8	Разрешение трафика, исходящий порт которого voice-port. Разрешение трафика включается только в случае наличия настроенного voice-port.
9	Выполнение HTTP/HTTPS прокси[1]
10	Функции Destination NAT[1]
11	Policy Based Routing
12	Routing Decision (FIB)
Если пакет перед передачей необходимо обработать протоколом более высокого уровня, выполняются следующие действия:	
12.1	Выполнение функций DOS defense[1]. На данном этапе выполняются специфичные функции защиты от DDOS из разделов firewall screen dos-defense, firewall screen spy-blocking, firewall screen suspicious-packets: ip firewall screen suspicious-packets large-icmp ip firewall screen dos-defense winnuke ip firewall screen spy-blocking port-scan

Шаг	Описание
12.2	Передача пакета в DPI[1]
12.3	Передача пакета в Netflow/sFlow (Ingress)[1]
12.4	Передача пакета в Antispam[1]
12.5	IPsec (decode)[1]. После выполнения этого шага происходит переход к п.3
13	tcp adjust-mss[1]
14	Выполнение функций DOS defense[1]. На данном этапе выполняются специфичные функции защиты от DDOS из разделов firewall screen dos-defense, firewall screen spy-blocking, firewall screen suspicious-packets: ip firewall screen suspicious-packets large-icmp ip firewall screen dos-defense winnuke ip firewall screen spy-blocking port-scan
15	Выполнение правил между специальными зонами, any/any
16	Передача пакета в DPI[1]
17	Разрешение трафика, исходящий порт которого voice-port. Разрешение трафика включается только в случае наличия настроенного voice-port.
18	Netflow/sFlow (Egress)[1]
19	Инспектирование пакета сервисом IPS/IDS в режиме service-ips inline[1]
20	BRAS (для исходящих пакетов)[1]
21	Выполнение функций Source NAT[1]
22	IPsec (encode)[1]
Если необходимо шифрование, то после этого процесса, выполняются следующие операции:	
22.1	Передача пакета в DPI[1]
22.2	tcp adjust-mss[1]
22.3	Netflow/sFflow (Egress)[1]
22.4	BRAS (для исходящих пакетов)
22.5	Выполнение функций Source NAT[1]

Шаг	Описание
23	Выполнение фрагментации пакетов
24	Выполнение функций QoS (ограничение полосы пропускания, классификация и т. д.)

⚠ [1]Данная функция выполняется только при наличии необходимых настроек.

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, 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