Руководство пользователя

iRZ Агрегация

Содержание

1. Введение **3**
1.1. Описание документа 3
1.2. Термины и определения 3
2. Описание интерфейса **4**
2.1. Описание серверной части 4
2.2. Описание клиентской части 9
3. Описание настроек **15**
3.1. Aggregation Type 15
3.2. Aggregation Method 15
3.3. Client CC, Server CC 16
4. Кластер серверов **17**
4.1. Реализация кластера серверов 17
4.1.1. Описание рабочего узла 18
4.1.2. Управление нагрузкой 18
4.1.3. Обмен информацией внутри кластера 19
5. Контакты **20**

1. Введение

1.1. Описание документа

Настоящий документ является руководством пользователя **iRZ Агрегация**.

Данный документ содержит описания интерфейса и функциональных возможностей, доступных пользователю, а также описание кластерной архитектуры серверов агрегации.

Перед началом работы необходимо ознакомиться с положениями настоящего документа.

1.2. Термины и определения

Агрегация (**aggregation**) - Процесс объединения нескольких физических каналов связи в один логический канал для увеличения пропускной способности, повышения надежности и балансировки нагрузки в сети.

Соединение MPTCP - Набор из одного или более субпотоков. Соединение MPTCP - это и есть "агрегация" каналов.

Субпоток (**subflow**) - Физическая основа работы агрегации. Поток TCP-сегментов, передаваемых по индивидуальному маршруту, который составляет часть MPTCP-соединения. Субпоток запускается и прерывается так же, как и обычное TCP-соединение.

Маршрут - Последовательность каналов между отправителем и получателем, определенная в этом контексте двумя парами адрес-порт отправителя и получателя.

Планировщик (MPTCP Scheduler) - Отвечает за выбор субпотока, в который будет направлен и передан пакет.

Сервер агрегации - Приложение, запущенное на виртуальной машине с операционной системой irzOS (/service/atunnel-server).

Кластер - Совокупность нод, которые работают вместе как единое целое для обеспечения отказоустойчивости, масштабируемости и распределения нагрузки.

Нода кластера - Отдельный сервер агрегации в составе кластера. Ноды обмениваются между собой сервисной информацией (о других нодах и клиентах).

Inter Process Communication (IPC) - Механизм взаимодействия между процессами, используемый для обмена данными и сообщениями.

Контроль загрузки (Congestion Control) - Механизм, который управляет потоком данных в сети, чтобы избежать перегрузки и обеспечить оптимальную производительность.

2. Описание интерфейса

Ниже представлено описание интерфейса **/service/atunnel-server** и **/tunnel/atunnel**. Подробное описание других разделов, требуемых для настройки агрегации, представлено в irzOS. Руководство по интерфейсу командной строки

Подробное описание настроек Aggregation Type, Aggregation Method, Client CC, Server CC представлено в п.3 настоящего руководства.

2.1. Описание серверной части

Серверная часть представляет собой виртуальную машину с операционной системой irzOS и установленным пакетом **irzos-atunnel-server** для работы с агрегацией.

Убедитесь, что на вашем устройстве установлено ПО irzOS и соответствующий пакет. В разделе "Service" должен появиться раздел "Atunnel-Server".

Для настройки серверной части требуется:

1. Перейти в /service/atunnel-server
2. В открывшемся туннеле ввести настройки и нажать **Apply**

/ service atunnel:server

Apply Clients

Disabled	☐
Port	5555
Max Clients	1500
Aggregation Type	fullmesh
Aggregation Method	blest (default)
Uplink	...select one or more...
	port0
Tunnel Mode	tap
Tunnel IP	192.168.217.1/24
MTU	1500
Authorization Timeout	5
Keepalive Period	5
Debug	☐
Socket RX	524288
Socket TX	524288
Server Broadcast Pool	3072
Server Public Key	
Server Private Key	
Server Root Key	
Server Cipher Mode	tls
Cluster Public Key	
Cluster Private Key	
Cluster Root Key	
Cluster Connect Timeout	10
Cluster Reconnect Period	5
Cluster Keepalive Period	10
Cluster ID (IP:Port)	127.0.0.1:5550
Cluster Nodes	

connected	1
rx-tx	0.00KB/0.44KB
state	running
status	ready
uplink	port0/active

Описания настроек и статусов представлены в таблице ниже.

Поле настройки	Описание
Disabled	Включен или выключен интерфейс
Port	Номер порта, который будет ожидать подключения клиентов. Допустимый диапазон от 1 до 65535, по умолчанию 5555
Max Clients	Максимальное количество клиентов на сервере. Допустимый диапазон от 0 (запрещено подключение клиентов) до 1500 (по умолчанию)
Aggregation Type	Алгоритм для управления субпотоками в MPTCP. Выбор правильного алгоритма может существенно повлиять на производительность и отказоустойчивость: **fullmesh, binder**
Aggregation Method	Метод агрегации определяет, как данные будут распределяться между доступными субпотоками MPTCP: **lrf, roundrobin, redundant, blest (default), ecf**
Uplink	Список интерфейсов, через которые осуществляется агрегация
Tunnel Mode	Тип виртуального сетевого интерфейса: **TAP** - виртуальное L2 (Ethernet) устройство; **TUN** - виртуальное L3 (IP) устройство
Tunnel IP	IP-адрес, присвоенный туннельному интерфейсу и маска подсети
MTU	Максимальный размер передаваемого пакета данных через туннельный интерфейс, в байтах. Допустимый диапазон от 576 до 8192, по умолчанию 1500. Данный параметр не обязан совпадать на клиенте и сервере. Если MTU туннеля больше, чем MTU внешнего канала, большой пакет будет разделен
Authorization Timeout	Время ожидания авторизации клиента на сервере агрегации, в секундах. Отсчет идет от установки TCP-соединения до успешной авторизации. Допустимый диапазон от 1 до 600, по умолчанию 5
Keepalive Period	Период отправки keep-alive сообщений для проверки состояния MPTCP соединения, в секундах. При отсутствии входящего трафика клиент и сервер независимо друг от друга отправляют keep-alive сообщения с ожиданием ответа. Соединение разрывается после двух периодов без обмена. При включенной опции Debug факт разрыва записывается в лог. Допустимый диапазон от 1 до 600, по умолчанию 5 Данный параметр не обязан совпадать на клиенте и сервере
Debug	Режим отладки (увеличивает количество отладочной информации в логе: не только критические ошибки и ошибки, но и предупреждения, информационные сообщения)

Socket RX, Socket TX	Настройка размеров RX/TX буферов MPTCP сокета в ядре, в байтах. Задают максимальный размер TCP-окна , фактический размер окна постоянно регулируется автоматически в зависимости от состояния сети. Увеличение значений позволяет обрабатывать резкие перепады нагрузки, что особенно полезно на каналах с большой или изменяющейся задержкой. Допустимый диапазон от 262144 до 4194304, по умолчанию 524288
Server Broadcast Pool	Размер пула для широковещательных пакетов, в штуках (пакетах). Имеет два назначения: для сервисного обмена с клиентами и для отправки ARP бродкастов (в режиме TAP). Допустимый диапазон от 128 до 4096, по умолчанию 3072. Оптимальное значение: Max Clients * 2
Server Public Key	Путь к файлу, содержащему сертификат сервера (включает в себя публичный ключ). Файл должен быть в формате base64 с расширением .crt, .csr или .key. Поддерживаются форматы ключей и их длина, определяемые операционной системой (RSA / ECDSA / DSA)
Server Private Key	Путь к файлу, содержащему приватный ключ сервера. Файл должен быть в формате base64 с расширением .crt, .csr или .key. Поддерживаются форматы ключей и их длина, определяемые операционной системой (RSA / ECDSA / DSA)
Server Root Key	Путь к файлу, содержащему корневой сертификат удостоверяющего центра (iRZ). Файл должен быть в формате base64 с расширением .crt, .csr или .key. Поддерживаются форматы ключей и их длина, определяемые операционной системой (RSA / ECDSA / DSA). Данный параметр должен совпадать с параметром Client Root Key на клиенте
Server Cipher Mode	Режим шифрования соединения: **none** – шифрование не применяется, настройки Server Public Key, Server Private Key и Server Root Key не используются; **tls** – протокол для безопасной передачи данных, шифрует весь трафик. При включении проверяется корневой сертификат сервера. При использовании данного режима можно либо заполнить поля Server Public Key, Server Private Key и Server Root Key, либо при запуске сервера будет сгенерирован самоподписанный сертификат. Данный параметр должен совпадать на сервере и клиенте, иначе авторизация не пройдет

Cluster Public Key	Путь к файлу, содержащему публичный ключ кластера. Используется для аутентификации и шифрования коммуникаций между нодами кластера
Cluster Private Key	Путь к файлу, содержащему приватный ключ кластера. Используется для аутентификации и шифрования коммуникаций между нодами кластера. Должен храниться в безопасности
Cluster Root Key	Путь к файлу, содержащему корневой сертификат удостоверяющего центра (iRZ), который подписал сертификаты нод кластера
Cluster Connect Timeout	Время ожидания подключения к соседней ноде кластера, в секундах. Период времени после установки TCP-соединения, включает в себя время на SSL-подключение и авторизацию между нодами
Cluster Reconnect Period	Период повторных подключений к соседним нодам кластера, в секундах
Cluster Keepalive Period	Период отправки keep-alive сообщений для проверки состояния связи между нодами кластера. Вместе с keepalive-пакетами передаётся информация о подключенных клиентах между нодами
Cluster ID (IP:port)	Уникальный идентификатор ноды в кластере, состоящий из IP-адреса и порта для подключения к нему соседних нод кластера
Cluster Nodes	Список нод, входящих в кластер. Для каждой ноды вводится её IP:port (аналогично списку серверов на клиенте агрегации)

Запись системного лога настраивается в разделе /system/logging
По умолчанию лог сохраняется ТОЛЬКО в опреативной памяти, его можно видеть в разделе journal в веб-интерфейсе.
Чтобы записывать лог в файл, нужно в разделе /system/logging выбрать пункт **Log to file** и указать имя файла filename. Тогда лог будет записываться в /storage/file/filename

Файлы с SSL-ключами сервера (Server Public Key, Server Private Key, Server Root Key) изначально загружаются во временную папку /storage/file, далее их можно переместить в /storage/certificate через **Import**.

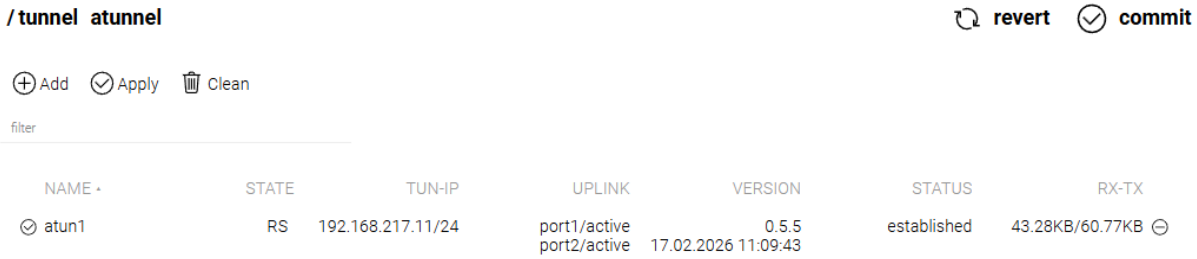
SSL-ключи кластера (Cluster Public Key, Cluster Private Key, Cluster Root Key) - для обмена информацией между нодами кластера. Если отсутствет корневой сертификат, будет использоваться самоподписанный. Если отсутвтует приватный ключ, то публичный и приватный ключи будут сгенерированы (RSA-2048 bit)

2.2. Описание клиентской части

Клиентская часть представляет собой роутер iRZ с операционной системой irzOS и установленным пакетом **irzos-atunnel-client** для работы с агрегацией.

Настройка клиентской части происходит в разделе /tunnel/atunnel

Все настроенные логические туннельные интерфейсы (далее - туннели) и их основная статусная информация отображаются в виде списка.



На роутере допускается создание нескольких туннелей atun, однако активным может быть только один. Несколько одновременно активных туннелей будут конфликтовать между собой.

Возможные значения статуса представлены в таблице ниже.

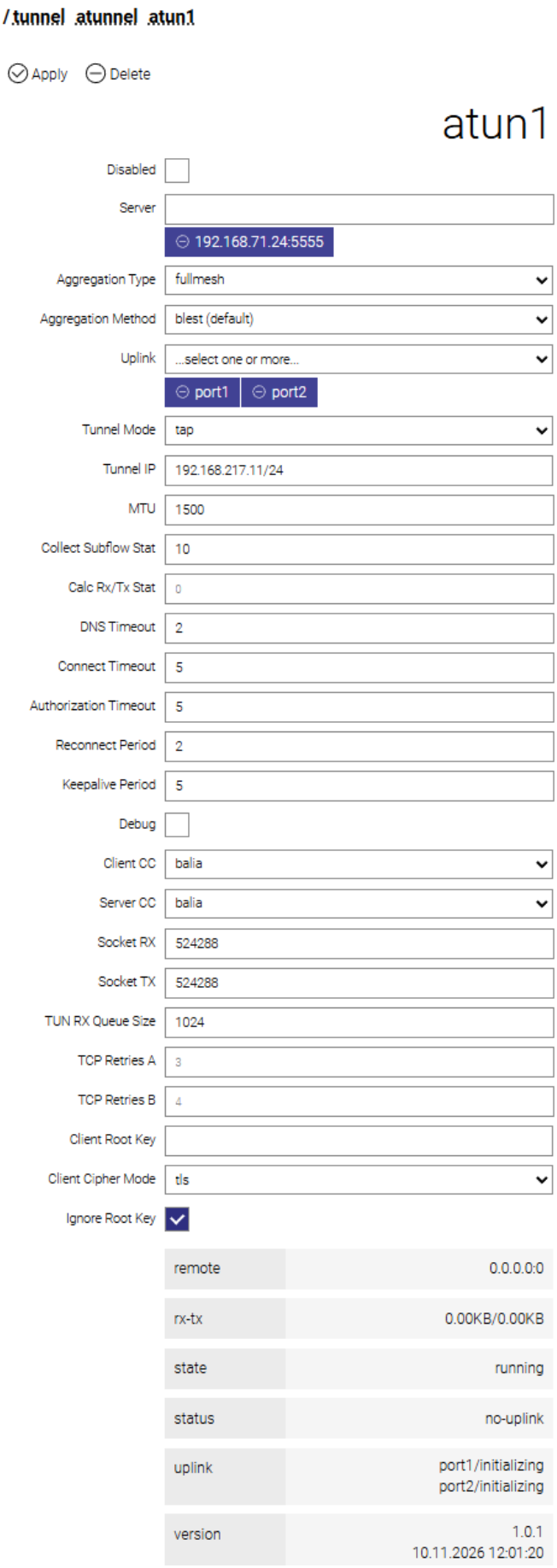
STATE	Состояние объекта, состоит из двух букв. 1 буква указывает на операционное состояние объекта: **R** = running, **X** = disabled, **E** = error 2 буква обозначает тип конфигурации объекта: **S** = static, **D** = dynamic
TUN-IP	IP-адрес туннеля на клиенте агрегации
UPLINK	Интерфейсы роутера-клиента, участвующие в агрегации, и их состояние: **unreachable** (недоступен) - интерфейс не поднят (link is down) или не включен в агрегацию (mptcp), **initializing** (подготовка) - по интерфейсу нет ни одного субпотока (это канал mptcp), **active** (в работе) - по интерфейсу есть активный(е) субпотоки
VERSION	Версия клиента агрегации и дата компиляции

STATUS	**no-uplink** - соединение не установлено, **authorizing** - происходит авторизация, **established** - соединение установлено
RX-TX	Объем полученных и переданных через туннель данных

Для создания нового туннеля нужно:

1. Нажать **Add**
2. Ввести имя туннеля (латинские буквы и цифры, при этом имя не должно начинаться с цифры)
3. Нажать **OK**

Далее откроется созданный туннель, где необходимо ввести настройки. Описания настроек представлены в таблице ниже.


/ tunnel atunnel atun1
Apply
Delete
atun1
Disabled
Server
192.168.71.24:5555
Aggregation Type
fullmesh
Aggregation Method
blest (default)
Uplink
...select one or more...
port1
port2
Tunnel Mode
tap
Tunnel IP
192.168.217.11/24
MTU
1500
Collect Subflow Stat
10
Calc Rx/Tx Stat
0
DNS Timeout
2
Connect Timeout
5
Authorization Timeout
5
Reconnect Period
2
Keepalive Period
5
Debug
Client CC
balia
Server CC
balia
Socket RX
524288
Socket TX
524288
TUN RX Queue Size
1024
TCP Retries A
3
TCP Retries B
4
Client Root Key
Client Cipher Mode
tls
Ignore Root Key
remote
0.0.0.0:0
rx-tx
0.00KB/0.00KB
state
running
status
no-uplink
uplink
port1/initializing
port2/initializing
version
1.0.1
10.11.2026 12:01:20

Параметр	Описание
Disabled	Включен или выключен интерфейс
Server	Список серверов агрегации в формате <host:port>, где host - IP адрес или доменное имя
Aggregation Type	Алгоритм для управления субпотоками в MPTCP: **fullmesh, binder**
Aggregation Method	Метод распределения данных между доступными субпотоками MPTCP: **lrf, roundrobin, redundant, blest (default), ecf**
Uplink	Список интерфейсов, через которые осуществляется агрегация
Tunnel Mode	Тип виртуального сетевого интерфейса: **TAP** - виртуальное L2 (Ethernet) устройство; **TUN** - виртуальное L3 (IP) устройство
Tunnel IP	IP-адрес, присвоенный туннельному интерфейсу и маска подсети
MTU	Максимальный размер передаваемого пакета данных через туннельный интерфейс, в байтах. Допустимый диапазон от 576 до 8192, по умолчанию 1500. Данный параметр не обязан совпадать на клиенте и сервере. Если MTU туннеля больше, чем MTU внешнего канала, большой пакет будет разделен
Collect Subflow Stat	Период опроса статистической информации по субпотокам у сервера, влияет на скорость отображения актуальной информации в полях "status" и "uplink", при этом дополнительно нагружает MPTCP сокет. Значение 0 отключает опрос (может потребоваться для отладки)
Calc Rx/Tx Stat	Период для расчета статистики передачи данных в миллисекундах. Значение 0 отключает расчет статистики
DNS Timeout	Время ожидания ответа DNS сервера для получения IP-адреса сервера агрегации по его доменному имени, в секундах
Connect Timeout	Общее время ожидания подключения к серверу агрегации, включая ответ DNS-сервера (если требуется), TCP соединение, авторизацию, в секундах

Authorization Timeout	Время ожидания авторизации клиента на сервере агрегации, в секундах. Отсчет идет от установки TCP-соединения до успешной авторизации. Допустимый диапазон от 1 до 600, по умолчанию 5
Reconnect Period	Период повторных подключений к серверам агрегации, в случае если ни с одним из серверов не удалось соединиться, в секундах
Keepalive Period	Период отправки keep-alive сообщений для проверки состояния MPTCP соединения, в секундах. При отсутствии входящего трафика клиент и сервер независимо друг от друга отправляют keep-alive сообщения с ожиданием ответа. Соединение разрывается после двух периодов без обмена. При включенной опции Debug факт разрыва записывается в лог. Допустимый диапазон от 1 до 600, по умолчанию 5 Данный параметр не обязан совпадать на клиенте и сервере
Debug	Режим отладки (увеличивает количество отладочной информации в логе: не только критические ошибки и ошибки, но и предупреждения, информационные сообщения)
Client CC, Server CC	Алгоритм TCP congestion control (правление перегрузкой TCP) на стороне клиента/сервера. Значения: **balia, lia, reno, bbr, cubic, westwood, vegas, veno, lp, illinois, olia, wvegas**
Socket RX, Socket TX	Настройка размеров RX/TX буферов MPTCP сокета в ядре, в байтах. Задают максимальный размер TCP-окна , фактический размер окна постоянно регулируется автоматически в зависимости от состояния сети. Увеличение значений позволяет обрабатывать резкие перепады нагрузки, что особенно полезно на каналах с большой или изменяющейся задержкой. Допустимый диапазон от 262144 до 4194304, по умолчанию 524288
TUN RX Queue Size	Длина входящей очереди TUN интерфейса (в блоках, размером с MTU)
TCP Retries A	Параметр ядра, определяющий максимальное количество повторных передач неподтвержденных пакетов, после которого TCP уведомляет сетевой уровень о проблеме с соединением и необходимости смены маршрута. Допустимый диапазон от 1 до 16, по умолчанию 3
TCP Retries B	Параметр ядра, определяющий максимальное количество повторных передач при экспоненциально увеличивающемся таймере (RTO), после которого соединение будет принудительно разорвано. Допустимый диапазон от 1 до 16, по умолчанию 4

Client Root Key	Путь к файлу, содержащему корневой сертификат удостоверяющего центра (iRZ). Применяется для проверки сертификата сервера при установке TLS-соединения без использования опции Ignore Root Key. Файл должен быть в формате base64 с расширением .crt, .csr или .key. Поддерживаются форматы ключей и их длина, определяемые операционной системой (RSA / ECDSA / DSA). Данный параметр должен совпадать с параметром Server Root Key на сервере
Client Cipher Mode	Режим шифрования соединения: **none** – шифрование не применяется; **tls** – протокол для безопасной передачи данных, шифрует весь трафик. При включении проверяется корневой сертификат сервера. Данный параметр должен совпадать на сервере и клиенте, иначе авторизация не пройдет
Ignore Root Key	Игнорировать проверку корневого сертификата сервера. Применяется при использовании автоматически сгенерированных самоподписанных сертификатов: активирован режим шифрования **tls**, но не заполнены поля Server Public Key, Server Private Key, Server Root Key на сервере



Запись системного лога настраивается в разделе /system/logging
По умолчанию лог сохраняется ТОЛЬКО в опреативной памяти, его можно видеть в разделе journal в веб-интерфейсе.
Чтобы записывать лог в файл, нужно в разделе /system/logging выбрать пункт **Log to file** и указать имя файла filename. Тогда лог будет записываться в /storage/file/filename

3. Описание настроек

3.1. Aggregation Type

Алгоритмы для управления субпотоками в MPTCP:

Fullmesh – создает субпотоки MPTCP (subflow) между всеми доступными интерфейсами. Обеспечивает высокую пропускную способность, но может создавать избыточную нагрузку на сеть, особенно при большом количестве интерфейсов. Подходит для сетей с высокой пропускной способностью и низкой задержкой.

Binder – изначально создает только один субпоток. Дополнительные субпотоки создаются только при необходимости, например, при обнаружении потери пакетов или перегрузке сети. Это позволяет экономить ресурсы сети, но может привести к снижению производительности в случае внезапного увеличения нагрузки. Рекомендуется для мобильных сетей или сетей с ограниченными ресурсами.

3.2. Aggregation Method

Методы распределения данных между доступными субпотоками MPTCP:

Lrf (Lowest-RTT-First) – всегда использует субпоток с наименьшим временем отклика (RTT), остальные субпотоки – в режиме ожидания или резервирования.

Roundrobin (циклический) – распределяет данные равномерно между всеми доступными субпотоками в циклическом порядке.

Redundant – отправляет трафик по всем доступным субпотокам (дублируя), обеспечивая максимальную отказоустойчивость. Используется для случаев, когда важна быстрая доставка за счёт избыточной передачи трафика. Но может снизить общую пропускную способность, особенно в сетях с высокой задержкой.

Blest (default) (Basic Low Extra Delay Background Transport) – стандартный планировщик, начиная с ядра Linux 4.14. Более сложный алгоритм, который стремится минимизировать задержку и максимально использовать пропускную способность, динамически адаптируясь к условиям сети. Он использует один primary субпоток для передачи большей части данных и secondary субпоток для передачи небольшого объема данных с целью оценки их качества. Если secondary субпоток оказывается лучше, чем primary, он становится новым primary. Этот планировщик обычно обеспечивает наилучшую производительность в разнообразных сетевых условиях.

Ecf (Earliest Completion First) – минимизирует время завершения каждого пакета, отправляя их по пути с наименьшим ожидаемым временем завершения. Алгоритм учитывает задержку и пропускную способность каждого пути, чтобы предсказать, какой из них доставит пакеты быстрее всего.

3.3. Client CC, Server CC

Алгоритмы TCP congestion control (правление перегрузкой TCP) на стороне клиента/сервера:

Lia (MPTCP) – базовая логика MultiPath TCP. Увеличивает окно пропорционально пропускной способности каждого пути для справедливости.

Reno – loss-based классика. При потере пакета уменьшает окно вдвое (Fast Recovery), затем линейный рост.

Bbr – model-based (Google). Явно оценивает BtlBw и RTprop, задает скорость для минимизации очередей и потерь.

Cubic – loss-based стандарт (Linux). После потери использует кубическую функцию для роста окна, фокусируясь на высоких скоростях.

Westwood – loss/delay-based. Оценивает доступную полосу по ACK и устанавливает окно на основе этой оценки при потере.

Vegas – delay-based. Регулирует окно, поддерживая небольшую постоянную очередь (вычисляемую по RTT).

Wvegas – модификация Vegas. Потоки получают пропорциональную долю полосы согласно назначенному весу.

Balia – гибрид Vegas и Westwood. Оптимизирует пропускную способность и справедливость в гетерогенных сетях.

Veno – гибрид Vegas и Reno. Использует рост RTT для раннего снижения скорости, но при потере действует как Reno.

Lp – для низкоприоритетного трафика. Сильно замедляет рост окна при признаках перегрузки (по RTT).

Illinois – delay-based для сетей с высокой BDP. Использует адаптивные шаги увеличения/уменьшения окна на основе RTT.

Olia (MPTCP) – улучшенный LIA. Гарантирует справедливость к обычному TCP и эффективно агрегирует полосу на путях.

4. Кластер серверов

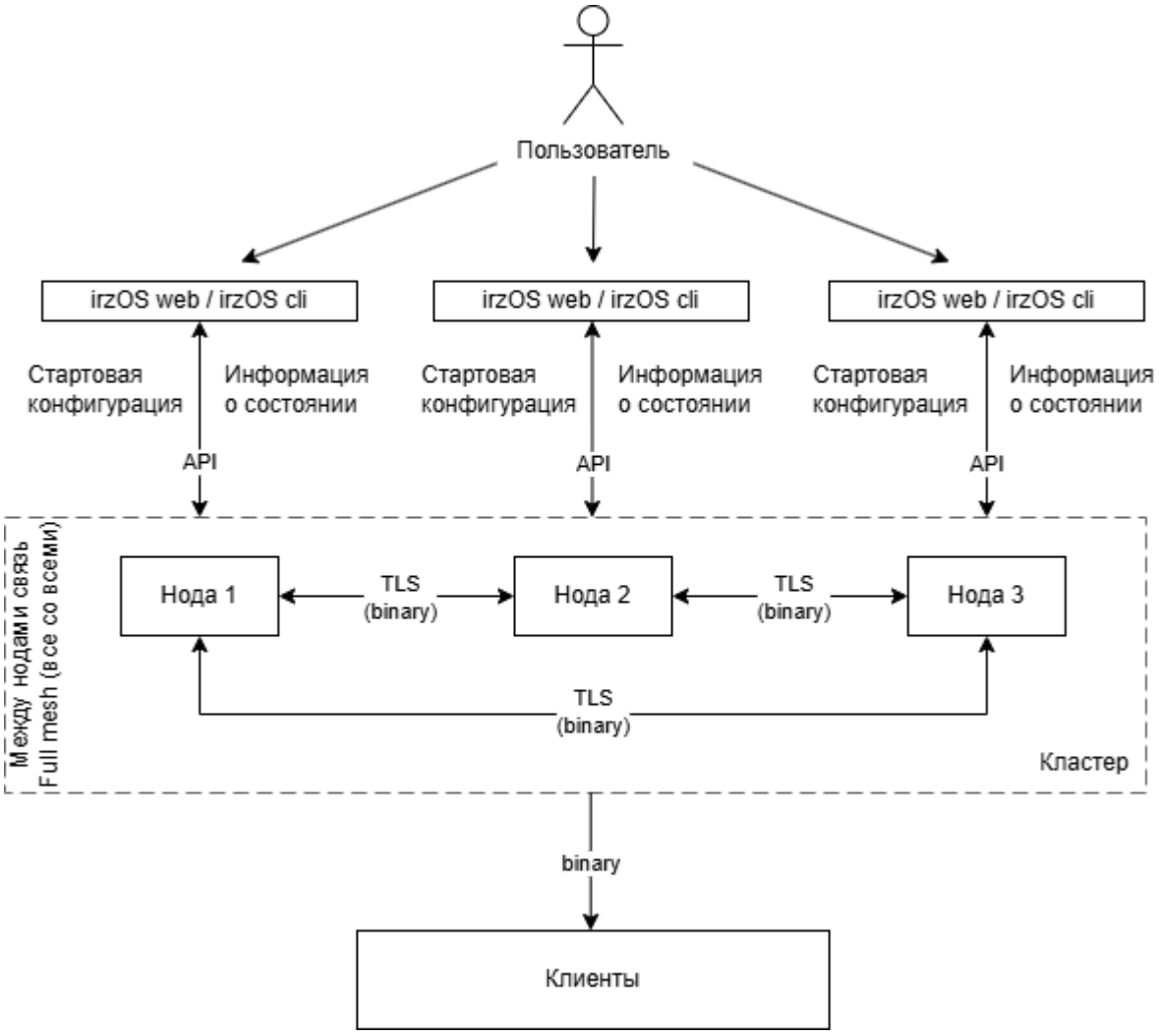
Сервера агрегации могут быть объединены в кластер.

Преимущества кластерной архитектуры для серверов агрегации:

- горизонтальное масштабирование
- распределение нагрузки
- отказоустойчивость на программном и аппаратном уровне

4.1. Реализация кластера серверов

Стартовая конфигурация кластера задается пользователем при настройке каждого из серверов.



4.1.1. Описание рабочего узла

У каждой ноды есть собственный ID.

Каждая нода хранит следующую информацию:

- ID (IP:port)
- адреса соседних нод
- список клиентов, подключенных к собственному серверу агрегации, и информацию о них
- списки клиентов, подключенных к другим серверам агрегации, и информацию о них

4.1.2. Управление нагрузкой

Для каждой ноды может быть указано максимальное количество клиентов (параметр **max-clients**).

Параметр **max-clients** не является обязательным. По умолчанию ему присваивается значение **1500**.

В настройках каждого роутера-клиента в параметре **Servers** указываются IP-адреса всех нод – серверов агрегации в составе кластера (Нода 1, Нода 2, Нода 3 на схеме).

При подключении клиент отправляет запросы на все доступные ноды и подключается к ток, которая ответила первой.

Если к ноде уже подключено **максимальное** количество клиентов, то в момент авторизации она отклонит запрос на подключение. Тогда клиент будет подключаться к следующей доступной ноде.

Первой отвечает нода с самой низкой загрузкой либо с самым быстрым каналом связи.

4.1.3. Обмен информацией внутри кластера

В процессе работы сервер отправляет своей ноде кластера сообщения об изменениях списка клиентов (подключился или отключился клиент с таким-то ID). Эта информация хранится на ноде.

Ноды обмениваются между собой информацией о подключенных клиентах. Таким образом, каждая нода хранит информацию обо всех клиентах и серверах, к которым они подключены.

Информация между нодами всегда передается в зашифрованном виде. Для аутентификации и шифрования коммуникаций между нодами используется SSL/TLS. TLS используется версии 1.2 в режиме совместимости с TLS 1.3

Любая нода ретранслирует запросы, требующие исполнения на сервере, к своему серверу или к другой ноде, если запрос к её серверу. Например, запросы детальной информации о клиенте (MPTCP_INFO), которая не хранится на ноде, либо запросы на действие (например, дисконнект клиента).

Информация о появлении новой ноды в кластере также будет передана остальным нодам.

Ноды обмениваются keepalive-сообщениями для контроля состояния соединения. Вместе с keepalive-пакетами передаётся информация о подключенных клиентах между нодами. При потере соединения данные о ноде удаляются.

5. Контакты

Новые версии прошивок, документации и сопутствующего программного обеспечения можно получить, обратившись по следующим контактам:

Санкт-Петербург

Сайт компании в Интернете	www.radiofid.ru
Тел. в Санкт-Петербурге	+7 (812) 318 18 19
e-mail	support@radiofid.ru
Telegram	@irzhelpbot

Наши специалисты всегда готовы ответить на все Ваши вопросы, помочь в установке, настройке и устранении проблемных ситуаций при эксплуатации оборудования.

В случае возникновения проблемной ситуации при обращении в техническую поддержку следует указывать версию программного обеспечения, используемого в роутере. Также рекомендуется к письму прикрепить журналы запуска проблемных сервисов, снимки экранов настроек и любую другую полезную информацию. Чем больше информации будет предоставлено сотруднику технической поддержки, тем быстрее он сможет разобраться в сложившейся ситуации.



Перед обращением в техническую поддержку настоятельно рекомендуется обновить программное обеспечение роутера до актуальной версии.



Нарушение условий эксплуатации (ненадлежащее использование роутера) лишает владельца устройства права на гарантийное обслуживание.