

# CDN 2.4

## Описание

CDN 2.4 дает следующие дополнительные возможности по сравнению с [CDN 2.1](#), [2.2](#) и [2.3](#):

- мониторинг состояния служебных соединений (сигналинга) в CDN
- управление состоянием служебных соединений в CDN

## Мониторинг состояния служебных соединений (сигналинга) в CDN

В CDN 2.4 на каждом узле можно проверить состояние служебных соединений (сигналинга) с другими узлами CDN при помощи REST API или CLI. Таким образом, можно контролировать связность CDN с бэкенда.

### Использование REST API

[REST API](#) запрос `/cdn/show_nodes` возвращает текущее состояние и статистику соединения с каждым узлом CDN

```
[
  {
    "version": "2.3",
    "role": "TRANSCODER",
    "inboundConnected": true,
    "outboundConnected": false,
    "globalState": "ACTIVE",
    "processingState": "NEW_STREAMS_ALLOWED",
    "id": "192.168.1.64"
  },
  {
    "version": "2.4",
    "role": "EDGE",
    "inboundConnected": true,
    "outboundConnected": false,
    "globalState": "ACTIVE",
    "processingState": "NEW_STREAMS_ALLOWED",
    "connectionStats": {
      "lastRtt": 2,
      "lastSn": 24,
      "lastMessageSentMs": 1592359761801,
      "lastACKReceivedMs": 1592359761802,
      "maxRtt": 1224,
    }
  }
]
```

```

    "inFlightSize": 0
  },
  "id": "192.168.1.65"
}
]

```

## Параметры

Параметр	Описание	Пример
inboundConnected	Установлено входящее соединение с данного узла	`true`
outboundConnected	Установлено исходящее соединение с данным узлом	`false`
lastRtt	Время доставки последнего пакета, мс	`2`
lastSn	Номер последнего пакета в последовательности	`24`
lastMessageSentMs	Метка времени последнего отправленного сообщения, мс	`1592359761801`
lastACKReceivedMs	Метка времени последнего подтверждения, мс	`1592359761802`
maxRtt	Максимальное время доставки пакета, мс	`1224`
inFlightSize	Размер текущего отправляемого пакета, байт	`0`

Отметим, что для соединений с узлами предыдущих версий CDN доступно только состояние соединения: установлено или нет.

## Использование CLI

Команда [интерфейса командной строки v 2](#)

```
cdn show-connections
```

отображает статистику соединений (только для узлов CDN 2.4)

```

Ip           State  Inbound Outbound Last rtt Max rtt Last SN In flight
-----
192.168.1.39 ACTIVE false   true    1       2057   127    0

```

## Управление состоянием служебных соединений в CDN

В некоторых случаях, например, при потерях на канале между узлами, связность определенного узла с CDN может быть нарушена. Для восстановления связности может потребоваться сброс и повторная установка соединений сигналинга. В предыдущих версиях CDN это можно было сделать только перезапуском узла. В CDN 2.4 добавлена возможность сброса соединений при помощи REST API. Эта операция не затрагивает медиапотoki, которые уже транслируются через данный узел.

### Сброс соединений при помощи REST API

REST-запрос должен быть HTTP/HTTPS POST запросом в таком виде:

- HTTP: `http://test.flashphoner.com:8081/rest-api/cdn/connection/reset_outbound`
- HTTPS: `https://test.flashphoner.com:8444/rest-api/cdn/connection/reset_outbound`

Здесь:

- `test.flashphoner.com` - адрес WCS-сервера
- `8081` - стандартный REST / HTTP порт WCS-сервера
- `8444` - стандартный HTTPS порт
- `rest-api` - обязательная часть URL
- `/cdn/connection/reset_outbound` - используемый REST-метод

REST запросы необходимо отправлять на сервер, на котором необходимо сбросить соединение

### REST-методы и статусы ответа

REST метод	Тело запроса	Статусы ответа	Описание
<code>^/cdn/connection/reset_inbound`</code>	<pre>{    "nodeId": "1 92.168.1.65 " }</pre>	200 OK 404 Node not found	Сбросить входящее соединение с указанным узлом

REST метод	Тело запроса	Статусы ответа	Описание
<code>`/cdn/connection/reset_outbound`</code>	<pre>{   "nodeId": "192.168.1.65" }</pre>	200 OK 404 Node not found	Сбросить исходящее соединение с указанным узлом
<code>`/cdn/connection/reset_all`</code>		200 OK	Сбросить все соединения со всеми узлами (добавлено в сборке <a href="#">5.2.1222</a> )

### Параметры

Параметр	Описание	Пример
nodeId	Идентификатор (IP-адрес) узла, соединение с которым должно быть сброшено	<code>`192.168.1.65`</code>

## Тонкая настройка служебных соединений

### Настройки сброса неактивных соединений

Если сервер не получает пакеты по служебному соединению, оно может быть сброшено. Эта возможность включается настройками для входящих и исходящих соединений соответственно

```
cdn_inbound_ws_read_socket_timeout=true
cdn_outbound_ws_read_socket_timeout=true
```

По умолчанию, интервал контроля приема пакетов установлен в 60 секунд

```
cdn_inbound_ws_read_socket_timeout_sec=60
cdn_outbound_ws_read_socket_timeout_sec=60
```

## Обратная совместимость с CDN 2.0-2.3

1. Для соединений с узлами предыдущих версий, статистика не отображается

2. При этом, соединение с узлом предыдущей версии может быть сброшено