



ViPNet Client 5 for Android SDK API

Справочник разработчика

Версия продукта: 5.3.0

© АО «ИнфоТеКС», 2025

ФРКЕ.00405-01 33 02

Версия продукта 5.3.0

Этот документ входит в комплект поставки продукта ViPNet, и на него распространяются все условия лицензионного соглашения.

Ни одна из частей этого документа не может быть воспроизведена, опубликована, сохранена в электронной базе данных или передана в любой форме или любыми средствами, такими как электронные, механические, записывающие или иначе, для любой цели без предварительного письменного разрешения АО «ИнфоТеКС».

ViPNet® является зарегистрированным товарным знаком АО «ИнфоТеКС».

Все названия компаний и продуктов, которые являются товарными знаками или зарегистрированными товарными знаками, принадлежат соответствующим владельцам.

АО «ИнфоТеКС»

125167, г. Москва, вн. тер. г. муниципальный округ Хорошевский, ул. Викторенко, д. 9, стр. 1, помещ. 47

Телефон: +7 (495) 737-6192, 8 (800) 250-0260 — бесплатный звонок из России (кроме Москвы)

Сайт: infotecs.ru

Служба поддержки: hotline@infotecs.ru

Содержание

Введение.....	4
О документе.....	4
Соглашения документа.....	4
О программе	4
Начало работы с ViPNet Client 5 for Android SDK API	5
Классы API.....	6
Описание контент-провайдера ViPNet Client 5 for Android	14
Пример	22
Обратная связь.....	23

Введение

О документе

Данный документ содержит инструкции по использованию SDK API ViPNet Client 5 for Android и описание функций этого API.

Документ предназначен для разработчиков, обеспечивающих взаимодействие приложения ViPNet Client 5 for Android с другими приложениями.

Соглашения документа

Обозначение	Описание
	Внимание! Содержит критически важную информацию
	Примечание. Содержит рекомендательную информацию
	Совет. Содержит полезные приёмы и хорошие практики
Название	Название элемента интерфейса: окна, вкладки, поля, кнопки, ссылки
Клавиша+Клавиша	Сочетание клавиш: нажмите первую клавишу и, не отпуская её, нажмите вторую
Меню > Команда	Последовательность элементов или действий
Код	Имя файла, путь, фрагмент кода или команда в командной строке



Примечание. В документе могут присутствовать снимки интерфейса из предыдущих версий продукта. Поэтому некоторые элементы интерфейса, которые не влияют на понимание текста, могут выглядеть не так, как в продукте.

О программе

ViPNet Client 5 for Android предназначен для защиты IP-трафика на мобильных устройствах с Android путём шифрования IP-пакетов. С помощью ViPNet Client 5, пользователи могут подключаться к узлам ViPNet и узлам, которые туннелируются координаторами ViPNet, по защищённым каналам и получать доступ к размещённым на этих узлах ресурсам: корпоративным веб-порталам, электронной почте, системе IP-телефонии, корпоративным серверам.

Начало работы с ViPNet Client 5 for Android SDK API

Минимальный уровень API Level (номер ревизии рабочего окружения) на мобильном устройстве для возможности взаимодействия с ViPNet Client 5 for Android через API — 26 (предоставляется Android версии 8.0 и выше).

Чтобы начать работу с ViPNet Client 5 for Android SDK API:

1 Поместите библиотеки `sdk-api.aar` и `sdk.aar` в каталог `libs` своего проекта.

2 В сборочный скрипт `build.gradle` добавьте следующие строки:

```
```groovy
repositories {
 flatDir {
 dirs 'libs'
 }
}
dependencies {
 implementation (name: 'sdk-api', ext: 'aar')
 implementation (name: 'sdk', ext: 'aar')
}
```
```

3 В своём проекте создайте объект `VpnSdkApi`, для этого добавьте строки следующего вида:

```
```kotlin
try {
 val clientApi = VpnSdkApi.from(appContext)
 //...

} catch (e: ClientNotInstalledException) {
 // если клиент не установлен
} catch (e: MultipleClientsInstalledException) {
 // если установлено несколько разных пакетов клиента
}
```
```



Внимание! В целях безопасности использовать SDK API ViPNet Client 5 for Android могут только приложения, подписанные сертификатом ИнфоТеКС.

В результате будет выполнено подключение к AIDL-сервису и контент-провайдеру ViPNet Client 5 for Android.



Внимание! Подключение к ViPNet Client 5 for Android осуществляется асинхронно, но на главном потоке, поэтому нельзя блокировать главный поток.

Все обращения к SDK API ViPNet Client 5 for Android синхронны, и их необходимо осуществлять не на главном потоке. В противном случае будет выброшено исключение `RpcOnMainThreadException`.

Объект `VpnSdkApi` автоматически следит за подключением к ViPNet Client 5 for Android и переподключается в случае необходимости (например, при обновлении версии ViPNet Client 5 for Android).

После использования объекта `VpnSdkApi` необходимо освободить ресурсы и отключиться от ViPNet Client 5 for Android. Для этого служит метод `dispose()` (см. [Классы API](#)).

Объект `VpnSdkApi` также наследует интерфейс `AutoCloseable`, так что может использоваться в конструкциях `try with resources` в Java или `use` в Kotlin.

Классы API

Таблица 1. `VpnSdkApi`

| Функция | Описание |
|---|--|
| <pre>public static VpnSdkApi from(@NonNull Context context) throws ClientNotInstalledException, MultipleClientsInstalledException</pre> | Создаёт объект класса для доступа к API ViPNet Client 5 for Android, автоматически определяя имя установленного пакета ViPNet Client 5 for Android |
| <pre>public static VpnSdkApi from(Context context, String clientAppPackage) throws ClientNotInstalledException</pre> | Создаёт объект класса для доступа к API ViPNet Client 5 for Android согласно переданному пакету |
| <pre>public static List<String> getInstalledClientPackages(@NonNull Context context)</pre> | Возвращает список имён установленных пакетов ViPNet Client 5 for Android |
| <pre>public ClientStatus getClientStatus() throws RemoteException</pre> | Возвращает объект, описывающий текущее состояние ViPNet Client 5 for Android |
| <pre>public void setClientStatusChangeListener(@Nullable ClientStatusChangeListener statusChangeListener)</pre> | Установить обратный вызов на изменение статуса |
| <pre>public RpcResult installKeys(@NonNull Uri keysUri, @NonNull CharSequence password) throws RemoteException</pre> | Установить дистрибутив ключей из URI. К URI должен быть предоставлен доступ |

| Функция | Описание |
|--|---|
| <code>public RpcResult installKeys(@NonNull File keysFile, @NonNull CharSequence password) throws RemoteException, IOException</code> | Установить дистрибутив ключей из доступного файла. Будет создан файл-провайдер и вызван предыдущий метод |
| <code>public RpcResult removeKeys() throws RemoteException</code> | Удалить дистрибутив ключей |
| <code>public RpcResult startVpn(@Nullable CharSequence password) throws RemoteException</code> | Включить VPN-соединение |
| <code>public RpcResult stopVpn() throws RemoteException</code> | Отключить VPN-соединение |
| <code>public RpcResult getNodeInfo(long node, @NonNull NodeInfo info) throws RemoteException</code>

<code>public NodeInfo getNodeInfo(long node) throws RemoteException, BadRpcResultException</code> | <p>Получить информацию об узле. Может принимать предопределённые значения:</p> <ul style="list-style-type: none"> • <code>NODE_SELF</code> — собственный узел; • <code>NODE_COORDINATOR</code> — активный координатор (сервер IP-адресов). <p>Второй вариант может вернуть <code>BadRpcResultException</code></p> |

| Функция | Описание |
|--|---|
| <pre> public boolean isCipherSuiteSupported(long cipherSuite) throws RemoteException, BadRpcResultException </pre> | <p>Проверить, поддерживается ли указанный набор криптографических алгоритмов.</p> <pre> /** * Поддерживаемые наборы криптографических алгоритмов. */ public enum CipherType { UNKNOWN(-1), G28147_CFB_WITH_IMIT(0), // ГОСТ 28147 в режиме CFB с алгоритмом выработки произвольного ключа для протокола IPLir 4.1 (бывший eGost) G3412_MAGMA_AEAD(1), // ГОСТ 31.12-2015 Магма + AEAD G28147_CTR_WITH_IMIT_CMACKDF(2), // Набор #2 из рекомендаций ТК2 по стандартизации протокола IPLir 6.0 G28147_CTR_WITH_IMIT(5), // ГОСТ 28147 в режиме CTR с алгоритмом выработки произвольного ключа для протокола IPLir 4.1, используется при наличии поддержки SSSE3 (бывший eGostOfb) G28147_CFB_WITH_IMIT_ITCSKDFv2(8), // ГОСТ 28147 в режиме CFB с алгоритмом выработки произвольного ключа для протокола IPLir 4.2 (бывший eGost с UseSecondVersionOfGetDerivedKey) G28147_CTR_WITH_IMIT_ITCSKDFv2(13), // ГОСТ 28147 в режиме CTR с алгоритмом выработки произвольного ключа для протокола IPLir 4.2, используется при наличии поддержки SSSE3 (бывший eGostOfb с UseSecondVersionOfGetDerivedKey) G3412_KUZNYECHIK_CTR_WITH_CMACH(14); // ГОСТ 31.12-2015 Кузнечик в режиме CTR + CMACH </pre> |
| <pre> public RpcResult encryptBlob(long targetNodeId, @NonNull DataBuffer inputBuffer, @NonNull DataBuffer outputBuffer) throws RemoteException </pre> | <p>Зашифровать массив данных для узла targetNodeId.</p> |
| <pre> public DataBuffer encryptBlob(long targetNodeId, @NonNull DataBuffer inputBuffer) throws RemoteException, BadRpcResultException </pre> | <p>Второй вариант может вернуть BadRpcResultException</p> |

| Функция | Описание |
|--|---|
| <pre>public RpcResult encryptBlobEx(long cipherSuite, long targetNodeId, @NonNull DataBuffer inputBuffer, @NonNull DataBuffer outputBuffer) throws RemoteException</pre> | Зашифровать массив данных для узла targetNodeId с помощью набора криптографических алгоритмов cipherSuite. |
| <pre>public DataBuffer encryptBlobEx(long cipherSuite, long targetNodeId, @NonNull DataBuffer inputBuffer) throws RemoteException, BadRpcResultException</pre> | Второй вариант может вернуть BadRpcResultException |
| <pre>public RpcResult decryptBlob(long sourceNodeId, @NonNull DataBuffer inputBuffer, @NonNull DataBuffer outputBuffer) throws RemoteException</pre> | Расшифровать массив данных от узла sourceNodeId. |
| <pre>public DataBuffer decryptBlob(long targetNodeId, @NonNull DataBuffer inputBuffer) throws RemoteException, BadRpcResultException</pre> | Второй вариант может вернуть BadRpcResultException |
| <pre>public RpcResult decryptBlobEx(long cipherSuite, long sourceNodeId, @NonNull DataBuffer inputBuffer, @NonNull DataBuffer outputBuffer) throws RemoteException</pre> | Расшифровать массив данных от узла sourceNodeId с помощью набора криптографических алгоритмов cipherSuite. |
| <pre>public DataBuffer decryptBlobEx(long cipherSuite, long targetNodeId, @NonNull DataBuffer inputBuffer) throws RemoteException, BadRpcResultException</pre> | Второй вариант может вернуть BadRpcResultException |
| <pre>public boolean isServiceConnected()</pre> | Возвращает статус подключения к ViPNet Client 5 for Android |
| <pre>public String getClientPackage()</pre> | Возвращает имя пакета приложения ViPNet Client 5 for Android, с которым ассоциирован объект |
| <pre>public ContentProviderDetails getContentProviderDetails()</pre> | Возвращает объект, предоставляющий информацию для доступа к контент-провайдерам ViPNet Client 5 for Android |
| <pre>public void dispose()</pre> | Прекратить использовать объект API.
Закрывает подключение и освобождает выделенные ресурсы |
| <pre>public boolean isDisposed()</pre> | Проверить, прекращено ли использование текущего объекта API |

| Функция | Описание |
|---|---|
| <code>public Intent getMakeReportIntent(@NotNull File outputReportFileInCache)</code> | Создать интент на запуск активности для сбора отчёта в ViPNet Client 5 for Android. Активности нужно вызывать через <code>startActivityForResult()</code> .
Файл с отчётом должен находиться в кэше (он будет передан из ViPNet Client 5 for Android через файл-провайдер) |
| <code>public CharSequence getClientAppId() throws RemoteException</code> | Получить уникальный идентификатор приложения ViPNet Client 5 for Android (в рамках инфраструктуры Android).
Для синхронизации статистики при отправке в централизованные системы |
| <code>public boolean isRestrictedUiMode() throws RemoteException</code> | Узнать состояние режима блокировки пользовательского интерфейса приложения ViPNet Client 5 for Android |
| <code>public void setRestrictedUiMode(boolean enabled)</code> | Установить режим блокировки пользовательского интерфейса приложения ViPNet Client 5 for Android.
В этом режиме пользователь не может влиять на состояние VPN |
| <code>public SharedFileInfo getSharedFileInfo(@NotNull String fileName)</code> | Получить информацию о файле, доступном через файл-провайдер ViPNet Client 5 for Android (см. Описание контент-провайдера ViPNet Client 5 for Android) |
| <code>public boolean isFullTunnelMode() throws RemoteException</code> | Узнать состояние режима полного туннеля ViPNet Client |
| <code>public void setFullTunnelMode(boolean enabled) throws RemoteException</code> | Установить режим режима полного туннеля ViPNet Client.
В этом режиме весь трафик направляется через VPN.
Данный вызов предназначен для использования в специальных окружениях, где недопустим открытый трафик |
| <code>public void hideSoftwareUpdate(String pkg) throws RemoteException</code> | Скрыть в интерфейсе ViPNet Client информацию о доступном обновлении |

Таблица 2. ClientStatus

| Поле | Описание |
|---|----------------------|
| <code>public boolean isKeysInstalled</code> | Установлены ли ключи |
| <code>public boolean isVpnEnabled</code> | Включён ли VPN |

| Поле | Описание |
|--|--|
| <code>public boolean isConnected</code> | Статус подключения к активному координатору (характеризует наличие связи) |
| <code>public boolean isInTransition</code> | ViPNet Client 5 for Android находится в состоянии перехода между «статическими» состояниями «отсутствуют ключи», «выключен» и «включён». Например, при включении, выключении, установке ключей, обновлении справочников и ключей. Рекомендуется дождаться состояния, когда данное поле будет <code>false</code> для продолжения работы с ViPNet Client 5 for Android во избежание возможных ошибок |
| <code>public boolean isSuspended</code> | ViPNet Client 5 for Android находится в состоянии «работа приостановлена», это возможно при отсутствии сетевых интерфейсов (режим полёта) или когда устройство находится в режиме Doze и ViPNet Client 5 for Android не находится в списке исключений энергосбережения. Трафик не проходит |
| <code>public boolean isError</code> | Критическая ошибка в работе ViPNet Client 5 for Android |
| <code>public boolean isAutoStop</code> | Включена ли настройка автоматического выключения VPN |
| <code>public boolean isUserTrafficRestricted</code> | <p>Заблокирован ли пользовательский трафик в следующих случаях:</p> <ul style="list-style-type: none"> • Просрочена лицензия ViPNet. • На устройстве получены права root или не пройдена проверка SafetyNet при минимальном уровне полномочий пользователя ViPNet |
| <code>public boolean isCompromised</code> | Обнаружены ли проблемы с безопасностью окружения на устройстве (не пройдена проверка SafetyNet или на устройстве получены права root) |
| <code>public String internalStatus</code> | Внутреннее представление состояния ViPNet Client 5 for Android (идентификатор состояния). Для справки и журналирования |
| <code>public boolean isAuthenticationRequired</code> | Требуется ли многофакторная аутентификация для доступа к корпоративным ресурсам |

Таблица 3. *RpcResult*

| Поле | Описание |
|---|--|
| <code>public ResultCode code</code> | <p>Содержит код ошибки:</p> <ul style="list-style-type: none"> • <code>SUCCESS</code>, • <code>UNKNOWN</code>, • <code>KEYS_ALREADY_INSTALLED</code>, • <code>KEYS_NOT_INSTALLED</code>, • <code>KEYS_INSTALLATION_FAILED</code>, • <code>BAD_PASSWORD</code>, • <code>NODE_NOT_FOUND</code>, • <code>TIMEOUT</code>, • <code>INTERRUPTED</code>, • <code>ENCRYPT_ERROR</code>, • <code>DECRYPT_ERROR</code>, • <code>CRYPT_UNAVAILABLE</code>, • <code>BUFFER_TOO_SMALL</code>, • <code>BUFFER_TOO_LARGE</code>, • <code>BAD_PASSWORD</code>, • <code>USER_NOT_FOUND</code>, • <code>APP_LOCKED</code> |
| <code>public String message</code> | Дополнительная информация об ошибке (может отсутствовать) |
| <code>public boolean isSuccessfull()</code> | Проверить, что результат <code>SUCCESS</code> |

Таблица 4. *DataBuffer*

| Поле | Описание |
|----------------------------------|------------------------------|
| <code>public byte[] bytes</code> | Массив данных для шифрования |

Таблица 5. *NodeInfo*

| Поле | Описание |
|---------------------------------|---------------------------------------|
| <code>public long id</code> | Идентификатор узла ViPNet (ViPNet Id) |
| <code>public String ip</code> | IPv4-адрес узла в виде строки |
| <code>public String name</code> | Название узла ViPNet |

Таблица 6. *SharedFileInfo*

| Функция | Описание |
|--|---|
| <code>public String getName()</code> | Получить имя файла |
| <code>public Uri getUri()</code> | Получить URI доступа к файлу через файл-провайдер |
| <code>public long getSize()</code> | Получить размер файла |
| <code>public long lastModified()</code> | Получить время последней модификации файла (аналогично <code>File.lastModified()</code>) |
| <code>public boolean exists()</code> | <code>true</code> , если файл существует. |
| <code>public ParcelFileDescriptor
openFileDescriptor(@NonNull Context context)
throws FileNotFoundException</code> | Получить дескриптор для чтения информации из файла |
| <code>public InputStream openInputStream(@NonNull
Context context) throws FileNotFoundException</code> | Получить поток чтения из файла |

Таблица 7. *ContentProviderDetails*

| Функция | Описание |
|--|---|
| <code>ExportDbUriProvider
getExportDbUriProvider()</code> | Класс предоставляет URI таблиц базы данных экспорта информации контент-провайдера ViPNet Client 5 for Android |
| <code>StatusInfoUriProvider
getStatusInfoUriProvider()</code> | Класс предоставляет URI таблиц базы данных состояния контент-провайдера ViPNet Client 5 for Android |
| <code>SharedFilesUriProvider
getSharedFilesUriProvider()</code> | Класс предоставляет URI файлов, доступных через файл-провайдер ViPNet Client 5 for Android |
| <code>ClientStatusInfo getClientStatusInfo()</code> | Запрашивает информацию из контент-провайдера состояния ViPNet Client 5 for Android. Побочный эффект — запуск процесса ViPNet Client 5 for Android |
| <code>ExtraInfo getExtraInfo()</code> | Запрашивает информацию из таблицы ExtraInfo из контент-провайдера клиента |
| <code>List<ClientProfileInfo>
getClientProfilesInfo()</code> | Запрашивает информацию из таблицы ClientProfileInfo из контент-провайдера клиента |

Описание контент-провайдера ViPNet Client 5 for Android

Для получения сторонним приложением информации о сети ViPNet, о конфигурации и текущем статусе приложения ViPNet Client 5 for Android вы можете использовать контент-провайдер ViPNet Client 5 for Android. При этом доступ к данным приложения ViPNet Client 5 for Android предоставляется только в режиме чтения.

Для обращения к данным приложения ViPNet Client 5 for Android объявите блок `<queries>` в файле манифеста своего приложения с именем пакета приложения ViPNet Client 5:

```
<queries>

    <package android:name="ru.vipnet.client" />

</queries>
```



Примечание. После подключения библиотеки `sdk-api.aar` к вашему проекту это разрешение добавляется автоматически.

База данных ExportDb для доступа к информации о сети ViPNet и конфигурации приложения ViPNet Client 5 for Android

Базовый URI для обращения к базе данных: `content://ru.vipnet.client.provider/`

Обслуживаемые таблицы:

- AlgSettings — настройки обработки прикладных протоколов. URI для доступа к таблице: `content://ru.vipnet.client.provider/AlgSettings`.

Таблица 8. Атрибуты таблицы AlgSettings

Название	Значения
Service	Идентификатор прикладного протокола: - dns - ftp - h.323 - sccp - sip
Transport	Транспортный протокол, с использованием которого работает прикладной протокол: - tcp - udp

Название	Значения
Port	Номер порта для прикладного протокола (например, 5060 для протокола SIP)

Пример реальных значений:

Table:  AlgSettings

	Service	Transport	Port
	Filter	Filter	Filter
1	sip	udp	5060
2	sip	tcp	5060
3	h.323	udp	1719
4	h.323	tcp	1720
5	ftp	tcp	21
6	dns	udp	53
7	sccp	tcp	2000

Рисунок 1. Пример содержимого таблицы AlgSettings

- DbInfo — содержит идентификатор последнего обновления информации во всей базе данных ExportDb. URI для доступа к таблице: `content://ru.vipnet.client.provider/DbInfo`.

Таблица 9. Атрибуты таблицы DbInfo

Название	Значения
ModificationStamp	Целочисленное значение, увеличивающееся при каждом изменении базы данных ExportDb. Каждая изменяемая запись в базе данных при этом получает то же самое значение ModificationStamp. Это позволяет синхронизировать только изменившиеся значения при кешировании содержимого контент-провайдера в локальную базу. Существует проблема удаления записей, так как фактически ModificationStamp может увеличиться при удалении, но записей с соответствующим значением в таблицах не будет. Для решения этой проблемы служит UpdateGuid в таблице StatusInfo контент-провайдера статуса приложения ViPNet Client 5 for Android (см. описание ниже). Если значение этого атрибута меняется, то необходимо выполнить полную синхронизацию, так как данные могли быть удалены. В противном случае достаточно синхронизировать по выборке из записей с более актуальными ModificationStamp.

Пример реальных значений:

Table:  DbInfo

ModificationStamp
Filter
1 7

Рисунок 2. Пример содержимого таблицы DbInfo

- `ClientInfo` — информация о версии приложения ViPNet Client 5 for Android, лицензии и базовых параметрах сети ViPNet. Эта таблица содержит не более одной строки. URI для доступа к таблице: `content://ru.vipnet.client.provider/ClientInfo`.

Таблица 10. Атрибуты таблицы `ClientInfo`

Название	Значения
<code>_id</code>	Идентификатор записи — 1 (в таблице всегда не более одной строки).
<code>Version</code>	Текущая версия приложения ViPNet Client 5 for Android.
<code>License</code>	Время окончания срока действия лицензии (в формате unixtime) или 0 (если лицензия бессрочная).
<code>OwnId</code>	Идентификатор локального узла с приложением ViPNet Client 5 for Android (строка — шестнадцатеричное число).
<code>CreateDateDst</code>	Время создания дистрибутива ключей (в формате unixtime).
<code>NameNetwork</code>	Имя сети ViPNet.
<code>UserPrivileges</code>	Полномочия пользователя, назначенные узлу в настройках роли (прикладной задачи) «VPN Client для мобильных устройств» в управляющем ПО ViPNet.
<code>ModificationStamp</code>	Идентификатор последнего обновления записи в таблице <code>ClientInfo</code> .

Пример реальных значений:

Table: ClientInfo							New Record		Delete Record
_id	Version	License	OwnId	CreateDateDst	NameNetwork	UserPrivileges	ModificationStamp		
Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter		
1 1	2.17.0.0-20181115175016	1546992000	15FD00B6	1515671266	Infotecs, Release Control, ...	2	3		

Рисунок 3. Пример содержимого таблицы `ClientInfo`

- `NodeInfo` — информация об узлах сети ViPNet, с которыми связан клиент ViPNet Client 5 for Android. URI для доступа к таблице: `content://ru.vipnet.client.provider/NodeInfo`. URI для доступа к записи по идентификатору узла ViPNet (`_id`): `content://ru.vipnet.client.provider/NodeInfo/#`.



Примечание. Чтобы оптимизировать получение полного списка узлов вместе со списком ролей (прикладных задач), вы можете использовать следующий URI для одновременного доступа к таблицам `NodeInfo` и `NodeTask`:

`content://ru.vipnet.client.provider/NodeInfo_NodeTask`.

Таблица 11. Атрибуты таблицы *NodeInfo*

Название	Значения
<code>_id</code>	Идентификатор узла ViPNet (строка — шестнадцатеричное число)
<code>NetworkNumber</code>	Десятичный номер сети ViPNet, в которой находится узел
<code>Name</code>	Имя узла
<code>Address</code>	IP-адрес узла (строка)
<code>CoordinatorId</code>	Идентификатор координатора ViPNet (строка — шестнадцатеричное число), за которым стоит данный узел, или 00000000 (если значение неизвестно)
<code>Broadcast</code>	Не используется, значение всегда 0
<code>ModificationStamp</code>	Идентификатор последнего обновления этой записи в таблице <i>NodeInfo</i> и связанных записей в таблице <i>NodeTask</i>

Пример реальных значений:

Table: <i>NodeInfo</i>							
	<code>_id</code>	<code>NetworkNumber</code>	<code>Name</code>	<code>Address</code>	<code>CoordinatorId</code>	<code>Broadcast</code>	<code>ModificationStamp</code>
	Filter	Filter	Filter	Filter	Filter	Filter	Filter
1	15FD00B7	5629	Song Shen	11.0.0.7	00000000	0	4
2	15FD0941	5629	Push Server	11.0.0.9	15FD000A	0	5
3	15FD00B8	5629	Paresh R. Prajapati	11.0.0.8	00000000	0	4
4	15FD00B6	5629	Neil Nelson	7.0.182.7	15FD000A	0	4
5	15FD00B5	5629	Mary Miles	11.0.0.6	00000000	0	4
6	15FD00B4	5629	Leslie Lawson	11.0.0.5	00000000	0	4

Рисунок 4. Пример содержимого таблицы *NodeInfo*

- *NodeTask* — информация о ролях (прикладных задачах) узлов сети ViPNet, с которыми связан клиент ViPNet Client 5 for Android. URI для доступа к таблице:
`content://ru.vipnet.client.provider/NodeTask`.

Таблица 12. Атрибуты таблицы *NodeTask*

Название	Значения
<code>NodeId</code>	Идентификатор узла ViPNet (строка — шестнадцатеричное число).
<code>Task</code>	Идентификатор роли (прикладной задачи) (десятичное число).
<code>Authority</code>	ASCII-код символа, обозначающего уровень полномочий (0, 1, 2, 3, h и другие уровни) роли (прикладной задачи). Например, уровню полномочий 2 соответствует код 32.

Пример реальных значений:

Table:  NodeTask

	NodeId 	Task	Authority
	Filter	Filter	Filter
5	15FD0941	130	32
6	15FD00B8	23	32
7	15FD00B8	79	32
8	15FD00B8	101	32
9	15FD00B7	23	32
10	15FD00B7	79	32
11	15FD00B7	101	32
12	15FD00B6	23	50

Рисунок 5. Пример содержимого таблицы NodeTask

- NodeTunnels — информация о туннелируемых IP-адресах. URI для доступа к таблице: `content://ru.vipnet.client.provider/NodeTunnels`.

Таблица 13. Атрибуты таблицы NodeTunnels

Название	Значения
NodeId	Идентификатор узла ViPNet (строка — шестнадцатеричное число), который туннелирует указанный диапазон IP-адресов
TunnelStartReal	Начало туннелируемого диапазона IP-адресов, адресация по реальным IP-адресам (в виде числа unsigned integer)
TunnelEndReal	Конец туннелируемого диапазона IP-адресов, адресация по реальным IP-адресам (в виде числа unsigned integer)
TunnelStartVirtual	Начало туннелируемого диапазона IP-адресов, адресация по виртуальным IP-адресам (в виде числа unsigned integer)
TunnelEndVirtual	Конец туннелируемого диапазона IP-адресов, адресация по виртуальным IP-адресам (в виде числа unsigned integer)
TunnelStartRealText	Начало туннелируемого диапазона IP-адресов, адресация по реальным IP-адресам (в виде строки IP-адреса)
TunnelEndRealText	Конец туннелируемого диапазона IP-адресов, адресация по реальным IP-адресам (в виде строки IP-адреса)
TunnelStartVirtualText	Начало туннелируемого диапазона IP-адресов, адресация по виртуальным IP-адресам (в виде строки IP-адреса)
TunnelEndVirtualText	Конец туннелируемого диапазона IP-адресов, адресация по виртуальным IP-адресам (в виде строки IP-адреса)

Пример реальных значений:

Table: NodeTunnels								
	Nodeid	TunnelStartReal	TunnelEndReal	TunnelStartVirtual	TunnelEndVirtual	TunnelStartRealText	TunnelEndRealText	TunnelStartVirtualText
	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter
1	15FD000A	167772200	167772200	184549387	184549387	10.0.0.40	10.0.0.40	11.0.0.11
2	15FD000A	167772264	167772264	184549388	184549388	10.0.0.104	10.0.0.104	11.0.0.12
3	15FD000A	167772283	167772283	184549389	184549389	10.0.0.123	10.0.0.123	11.0.0.13
4	15FD000A	167772390	167772390	184549390	184549390	10.0.0.230	10.0.0.230	11.0.0.14
5	15FD000A	167772405	167772405	184549391	184549391	10.0.0.245	10.0.0.245	11.0.0.15
6	15FD000A	167772410	167772410	184549392	184549392	10.0.0.250	10.0.0.250	11.0.0.16
7	15FD000E	3232236034	3232236034	184549393	184549393	192.168.2.2	192.168.2.2	11.0.0.17

Рисунок 6. Пример содержимого таблицы NodeTunnels

База данных StatusInfoDb для доступа к информации о текущем статусе приложения ViPNet Client 5 for Android

Базовый URI для обращения к базе данных: `content://ru.vipnet.client.status_provider/`

Обслуживаемые таблицы:

- **StatusInfo** — текущий статус приложения ViPNet Client 5 for Android. URI для доступа к таблице: `content://ru.vipnet.client.status_provider/StatusInfo`.

Таблица 14. Атрибуты таблицы StatusInfo

Название	Значения
VpnStatus	Статус приложения ViPNet Client 5 for Android: • ENABLED — ViPNet Client 5 for Android подключён к сети ViPNet (VPN-соединение включено). • DISABLED — ViPNet Client 5 for Android не подключён к сети ViPNet (VPN-соединение выключено)
UpdateGuid	Возможные значения: • none — ключи ViPNet не установлены. • значения вида <code>UUID.randomUUID().toString()</code>, если ключи ViPNet установлены. Значение меняется при каждом обновлении справочников и ключей ViPNet или других событиях, которые могут привести к удалению части информации из базы <code>ExportDb</code>. Изменение значения этого атрибута сигнализирует о необходимости полной синхронизации базы <code>ExportDb</code> с локальной базой

- **ClientStatusInfo** — расширенный статус приложения ViPNet Client 5 for Android. URI для доступа к таблице: `content://ru.vipnet.client.status_provider/ClientStatusInfo`.

Таблица 15. Атрибуты таблицы *ClientStatusInfo*

Название	Значения
IsKeysInstalled	Возможные значения: • 1 — ключи ViPNet установлены; • 0 — ключи ViPNet не установлены
IsVpnEnabled	Возможные значения: • 1 — VPN включён; • 0 — VPN выключен. Дублирует поле VpnStatus таблицы StatusInfo.
IsConnected	Возможные значения: • 1 — связь с координатором установлена; • 0 — связь с координатором отсутствует
IsInTransition	Возможные значения: • 1 — ViPNet Client 5 for Android находится в переходном состоянии (запуск, остановка, применение обновлений и тому подобное); • 0 — ViPNet Client 5 for Android находится в состоянии готовности приёма команд
IsSuspended	Возможные значения: • 1 — ViPNet Client 5 for Android находится в состоянии, когда у него нет доступа к сети (режим полёта, отсутствие подключений, режим энергосбережения); • 0 — ViPNet Client 5 for Android находится в рабочем состоянии
IsError	Возможные значения: • 1 — произошла фатальная ошибка, требуется вмешательство; • 0 — ViPNet Client 5 for Android работает штатно
IsAutoStop	Возможные значения: • 1 — настройка автоматического выключения VPN включена; • 0 — настройка автоматического выключения VPN выключена
IsUserTrafficRestricted	Возможные значения: • 1 — пользовательский трафик заблокирован (просрочена лицензия ViPNet или на устройстве получены права root либо не пройдена проверка SafetyNet при минимальном уровне полномочий пользователя ViPNet); • 0 — пользовательский трафик не заблокирован

Название	Значения
IsCompromised	Возможные значения: 1 — обнаружена проблема с безопасностью окружения на устройстве (не пройдена проверка SafetyNet или на устройстве получены права root); 0 — проблем с безопасностью окружения на устройстве не обнаружено
InternalStatus	Строковое представление текущего состояния ViPNet Client 5 for Android (справочная информация, формат и значения могут меняться, на параметр нельзя ориентироваться в коде проверок)
isAuthenticationRequired	Возможные значения: 1 — требуется многофакторная аутентификация для доступа к корпоративным ресурсам. Аутентификация выполняется в UI ViPNet Client 5 for Android; 0 — аутентификация не требуется

- SoftwareUpdateInfo — отражает наличие готовых для установки обновлений ПО, присланных через инфраструктуру ViPNet. URI для доступа к таблице:
content://ru.vipnet.client.status_provider/SoftwareUpdateInfo.

Таблица 16. Атрибуты таблицы SoftwareUpdateInfo

Название	Значения
ApkName	Имя файла *.apk с обновлением ПО
Version	Версия приложения, которая содержится в обновлении
AppName	Название приложения, для которого поступило обновление
PackageName	Название пакета, который содержится в обновлении
Uri	URI для доступа к файлу *.apk через файл-провайдер

Файл-провайдер для доступа к публичным справочникам ViPNet Client 5 for Android (shared files)

Базовый URI для обращения к файл-провайдеру:

content://ru.vipnet.client.sharedFilesProvider/shared/

Для доступа к конкретному файлу необходимо получать информационный объект, возвращаемый функцией `SharedFileInfo VpnSdkApi.getSharedFileIntent(@NotNull String fileName)`, которой передаётся имя этого файла.

Список доступных файлов определён в классе `SharedFiles`.

Таблица 17. Доступные файлы справочников

Название файла	Константа в коде	Описание
sip.contacts	SharedFiles.SIP_CONTACTS	JSON-файл со списком SIP-абонентов для приложения ViPNet CSS Connect

SharedFileInfo содержит информацию о том, существует ли файл, его размер, время его последней модификации и функции получения доступа к данным `openFileDescriptor` и `openInputStream`.

Пример

В следующем коде происходит установка обработчика изменения статуса ViPNet Client 5 for Android, проверка, установлены ли ключи, их удаление при наличии и установка новых ключей:

```
```kotlin
VpnSdkApi.from(context).use { clientApi ->
 // использование clientApi

 clientApi.setClientStatusChangeListener {
 // статус изменился
 }

 // удалить ключи, если они уже установлены
 if (clientApi.clientStatus.isKeysInstalled) clientApi.removeKeys()

 // установить ключи
 val installResult = clientApi.installKeys(dstFile, dstPassword)

 // получить информацию о собственном узле
 val selfNodeInfo = clientApi.getNodeInfo(NodeInfo.NODE_SELF)
 //...
}
```
```

ClientStatus содержит важную информацию о текущем состоянии ViPNet Client 5 for Android. Значение может быть получено двумя способами: через AIDL-интерфейс вызовом `clientApi.clientStatus` или через контент-провайдер вызовом `clientApi.contentProviderDetails.clientStatus`.

С точки зрения использования API следует обратить внимание на значение `clientStatus.isInTransition`, которое сообщает о том, что ViPNet Client 5 for Android находится в переходном состоянии и может быть не готов обрабатывать команды. Поэтому, перед очередным вызовом API следует дождаться, когда это значение сбросится.

При установке ключей происходит передача файла дистрибутива ключей через `FileProvider`. Вы можете воспользоваться уже реализованным в SDK API ViPNet Client 5 for Android, тогда для установки нужно передавать объект `File`, который должен находиться в месте, доступном для чтения. Или использовать альтернативный `FileProvider` и передавать объект `Uri`, а также позаботиться о том, чтобы на этот `Uri` были выданы разрешения на доступ.

Обратная связь

Контактная информация

- Единый многоканальный телефон:
+7 (495) 737-6192,
8 (800) 250-0-260 — бесплатный звонок из России (кроме Москвы).
- Служба поддержки: hotline@infotecs.ru.
Форма для обращения в службу поддержки через сайт.
Канал поддержки в Telegram: t.me/vhd21
Телефон для клиентов с расширенной поддержкой: +7 (495) 737-6196.
- Отдел продаж: soft@infotecs.ru.

Дополнительная информация на сайте ИнфоТеКС

- [О продуктах ViPNet](#).
- [О решениях ViPNet](#).
- [Часто задаваемые вопросы](#).
- [Форум пользователей продуктов ViPNet](#).

Если вы обнаружили уязвимости в продуктах компании, сообщите о них по адресу security-notifications@infotecs.ru. Распространение информации об уязвимостях продуктов компании ИнфоТеКС регулируется [политикой ответственного разглашения](#).