

МЕТОДЫ РАБОТЫ С
БД СКУД ПО REST API

Версия 1.0

Листов 79

СОДЕРЖАНИЕ

1. Общие сведения	4
2. Настройка компьютеров и модуля редактирования	5
2.1. Настройка сервера (компьютера с VideoNet).....	5
2.2. Настройка клиента (компьютера с внешней системой)	9
2.3. Настройка модуля редактирования в VideoNet	12
3. Методы добавления данных в БД СКУД из внешней системы	14
3.1. Общие сведения.....	14
3.2. Добавление подразделения в БД СКУД.....	14
3.3. Добавление должности в БД СКУД	16
3.4. Добавление человека в БД СКУД.....	17
3.5. Добавление автомобиля в БД СКУД.....	20
4. Методы получения данных из БД СКУД.....	23
4.1. Общие сведения.....	23
4.2. Запрос списка групп доступа	23
4.3. Запрос данных о группе доступа	25
4.4. Запрос списка сотрудников.....	27
4.5. Запрос данных о человеке	31
4.6. Запрос представительской фотографии человека	33
4.7. Запрос всех фотографий, закреплённых за человеком	34
4.8. Запрос списка автомобилей	36
4.9. Запрос данных об автомобиле	39
4.10. Запрос данных о ключах	41
4.11. Запрос списка подразделений	44
4.12. Запрос списка должностей	46
5. Методы для изменения данных в БД СКУД.....	50
5.1. Общие сведения.....	50
5.2. Изменение параметров человека	50
5.3. Изменение параметров автомобиля	53
5.4. Назначение человеку прав доступа на объект	55

5.5. Аннулирование прав доступа на объект у человека.....	57
5.6. Назначение автомобилю прав доступа на объект.....	58
5.7. Аннулирование прав доступа на объект у автомобиля	61
5.8. Выдача ключей человеку	62
5.9. Аннулирование выданных ключей у человека	64
5.10. Выдача ключей автомобилю.....	65
5.11. Аннулирование выданных ключей у автомобиля	68
5.12. Блокировка доступа людям.....	69
5.13. Разблокировка доступа людям	70
6. Методы удаления данных из БД СКУД.....	73
6.1. Общие сведения.....	73
6.2. Удаление подразделения из БД СКУД.....	73
6.3. Удаление должности из БД СКУД.....	74
6.4. Удаление человека из БД СКУД	75
6.5. Удаление автомобиля из БД СКУД.....	76
Приложение Тестовое приложение.....	78
Исходный код на Python	78
Исходный код на C#.....	78

1. ОБЩИЕ СВЕДЕНИЯ

1.1. VideoNet предоставляет возможность обеспечивать изменение списка объектов доступа (людей и автомобилей) и назначение им групп доступа в базе данных СКУД сторонним приложениям (далее – Внешняя система) с помощью REST API.

1.1.1. Для передачи сообщений, Внешняя система должна представлять собой REST API клиент (описание настройки см. в разделе 2.2), для которого VideoNet будет выступать в качестве сервера (описание настройки см. в разделе 2.1).

1.2. Изменение списка объектов доступа (людей и автомобилей) и назначение им групп доступа реализовано с помощью Модуля редактирования БД СКУД по REST API (далее – модуль редактирования).

1.2.1. Для работы с БД СКУД модуль редактирования должен размещаться на основном сервере БД СКУД.

1.2.2. К одному компьютеру может быть подключён только один модуль редактирования.

1.2.3. Для работы с модулем редактирования на компьютере VideoNet должна быть лицензия «Модуль редактирования БД СКУД по REST API» (SM-ACS-API).

1.2.4. Существует возможность заблокировать модуль редактирования, в результате чего редактирование БД СКУД с помощью REST API производиться не будет.

1.3. Изменения, внесённые в БД СКУД с помощью модуля редактирования, протоколируются в журнале «Конфигурация СКУД».

1.4. В случае большого количества запросов к VideoNet может возникнуть ситуация при которой обработка запросов не будет возможна некоторое время. В таком случае, в ответ на запрос VideoNet вернёт код ошибки сервера 503. Запрос к VideoNet можно будет направить повторно через некоторое время.

2. НАСТРОЙКА КОМПЬЮТЕРОВ И МОДУЛЯ РЕДАКТИРОВАНИЯ

Внимание! Выполнение настроек, описанных в разделах 2.1 и 2.2, необходимо только для работы с использованием HTTPS. Если Вы хотите использовать HTTP – отключите проверку SSL-сертификата во внешней системе и выполните настройки, описанные в разделе 2.3.

Корректность выполнения настроек можно проверить при помощи тестового приложения (см. Приложение к документу).

2.1. Настройка сервера (компьютера с VideoNet)

2.1.1. На компьютере, на котором установлен VideoNet, запустите Windows PowerShell от имени администратора.

2.1.2. Введите следующую команду в Windows PowerShell для выпуска на сервере SSL-сертификата (далее – Сертификат):

```
$cert = New-SelfSignedCertificate -certstorelocation cert:\localmachine\my -  
dnsname Желаемое DNS имя
```

Например, если DNS должен быть *test.videonet.ru*, то команда должна выглядеть следующим образом: `$cert = New-SelfSignedCertificate -certstorelocation cert:\localmachine\my -dnsname test.videonet.ru`

2.1.3. Введите следующую команду для создания пароля:

```
$pwd = ConvertTo-SecureString -String "Желаемый пароль" -Force -  
AsPlainText
```

Пароль должен указываться в кавычках.

Пример команды: `$pwd = ConvertTo-SecureString -String "112233" -Force -AsPlainText`

2.1.4. Введите следующую команду для получения информации о сертификате:

```
$certpath = "Cert:\localMachine\my\$(($cert.Thumbprint))"
```

2.1.5. Введите следующую команду для экспорта созданного сертификата в желаемую папку:

`Export-PfxCertificate -Cert $certpath -FilePath Полный путь папки и имя файла.pfx -Password $pwd`

Например, для сохранения сертификата с именем «cert» в папку `F:\certificate` команда должна выглядеть следующим образом: `Export-PfxCertificate -Cert $certpath -FilePath F:\certificate\cert.pfx -Password $pwd`

В результате выполнения команды появится информация об экспорте (см. Рисунок 1). В указанную папку будет сохранён файл сертификата.

```
PS C:\Windows\system32> $cert = New-SelfSignedCertificate -certstorelocation cert:\localmachine\my -dnsname test.videonet.ru
PS C:\Windows\system32> $pwd = ConvertTo-SecureString -String "112233" -Force -AsPlainText
PS C:\Windows\system32> $certpath = "Cert:\localMachine\my\${$cert.Thumbprint}"
PS C:\Windows\system32> Export-PfxCertificate -Cert $certpath -FilePath D:\Work\cert.pfx -Password $pwd
```

Каталог: D:\Work

Mode	LastWriteTime	Length	Name
-a----	16.05.2024 11:15	2661	cert.pfx

Рисунок 1

2.1.6. После сохранения файла сертификата необходимо получить его отпечаток для добавления в конфигурационный файл VideoNet. Для этого введите следующую команду в Windows PowerShell:

`$cert.Thumbprint`

2.1.6.1. Скопируйте отобразившийся отпечаток сертификата (см. Рисунок 2).

```
PS C:\Windows\system32> $cert.Thumbprint
6E1E62D39F2FC3C553967276F47A6EB61EC7C72B
```

Рисунок 2

2.1.7. Откройте файл «VideoNet.config», находящийся в папке, в которую установлен VideoNet (по умолчанию: `C:\Program Files (x86)\SKYROS\VideoNet 9`).

2.1.8. Добавьте в файл в секцию «Behaviour» следующую строку (см. Рисунок 3) и сохраните его:

`<ETCRestServiceThumbprint>Отпечаток сертификата</ETCRestServiceThumbprint>`

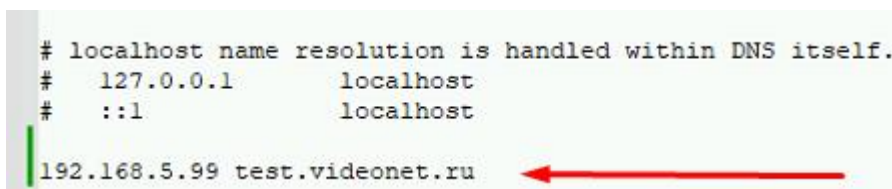
Например:

```
<ETCRestServiceThumbprint>42a800962e93c245d45500da08ab3f26e701ff8d</ETCRestServiceThumbprint>
```

```
<ETCRestServiceThumbprint>42a800962e93c245d45500da08ab3f26e701ff8d</ETCRestServiceThumbprint>  
</Behaviour>  
<!-- Отладочные параметры -->  
</VideoNetConfig>
```

Рисунок 3

2.1.9. Откройте файл «hosts», находящийся по адресу «C:\Windows\System32\drivers\etc\hosts» и добавьте в него IP-адрес компьютера и DNS имя, которое было указано при выполнении команды, описанной в п. 2.1.2 (см. Рисунок 4).



```
# localhost name resolution is handled within DNS itself.  
# 127.0.0.1      localhost  
# ::1           localhost  
192.168.5.99 test.videonet.ru
```

A red arrow points to the newly added line: 192.168.5.99 test.videonet.ru

Рисунок 4

2.1.10. После этого запустите сохранённый файл сертификата (например, «cert.pfx»). Откроется «Мастер импорта сертификатов» (см. Рисунок 5).

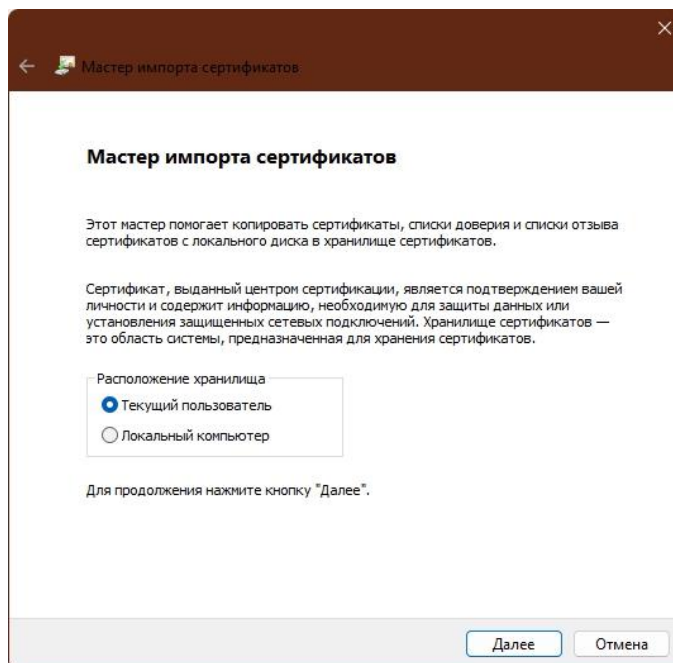


Рисунок 5

2.1.11. Выберите пункт «Локальный компьютер» и нажмите «Далее».

2.1.12. Проверьте имя файла в строке «Имя файла» и нажмите «Далее».

2.1.13. Введите пароль, указанный при выполнении команды, описанной в п. 2.1.3, в поле «Пароль».

2.1.14. Включите параметр «Пометить этот ключ как экспортируемый...» (см. Рисунок 6) и нажмите «Далее».

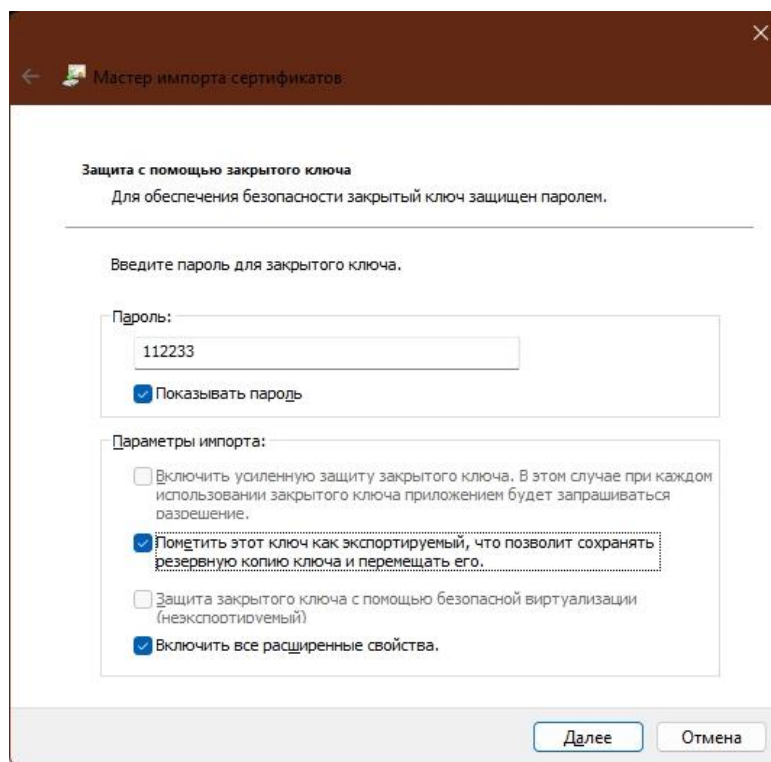


Рисунок 6

2.1.15. Выберите пункт «Поместить все сертификаты в следующее хранилище», а затем нажмите кнопку «Обзор». Откроется окно «Выбор хранилища сертификата».

2.1.16. В окне «Выбор хранилища сертификата» выберите пункт «Доверенные корневые центры сертификации» и нажмите «ОК» (см. Рисунок 7). Нажмите «Далее» в окне мастера импорта сертификатов.

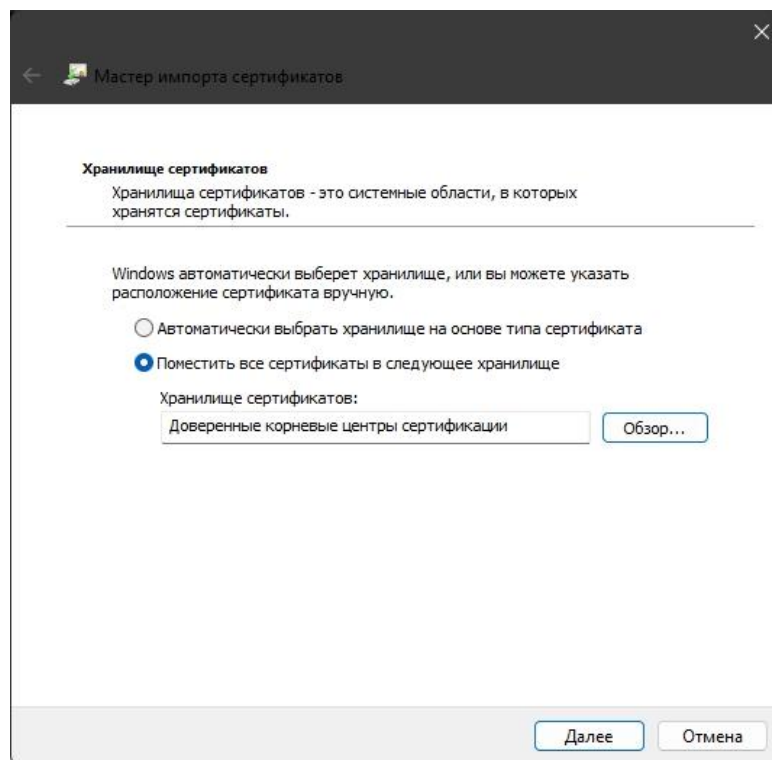


Рисунок 7

2.1.17. На шаге «Завершение мастера импорта сертификатов» нажмите «Готово». Отобразится сообщение «Импорт успешно выполнен».

На этом настройка сервера будет завершена, и можно будет запускать VideoNet.

2.2. Настройка клиента (компьютера с внешней системой)

2.2.1. Перенесите файл сертификата (с расширением «.pfx»), созданный на сервере (см. п. 2.1.5) на компьютер-клиент.

2.2.2. После этого запустите файл сертификата (например, «cert.pfx»). Откроется «Мастер импорта сертификатов» (см. Рисунок 5).

2.2.3. Выберите пункт «Локальный компьютер» и нажмите «Далее».

2.2.4. Проверьте имя файла в строке «Имя файла» и нажмите «Далее».

2.2.5. Введите пароль, указанный при выполнении команды, описанной в п. 2.1.3, в поле «Пароль».

2.2.6. Включите параметр «Пометить этот ключ как экспортируемый...» (см. Рисунок 6) и нажмите «Далее».

2.2.7. Выберите пункт «Поместить все сертификаты в следующее хранилище», а затем нажмите кнопку «Обзор». Откроется окно «Выбор хранилища сертификата».

2.2.8. В окне «Выбор хранилища сертификата» выберите пункт «Доверенные корневые центры сертификации» и нажмите «ОК» (см. Рисунок 7). Нажмите «Далее» в окне мастера импорта сертификатов.

2.2.9. На шаге «Завершение мастера импорта сертификатов» нажмите «Готово». Отобразится сообщение «Импорт успешно выполнен».

2.2.10. Повторите действия по п. 2.2.2-2.2.9, но в окне «Выбор хранилища сертификата» (см. п. 2.2.8) выберите пункт «Личное».

2.2.11. Откройте файл «hosts», находящийся по адресу «C:\Windows\System32\drivers\etc\hosts» и добавьте в него IP-адрес и DNS имя сервера, которое было указано при выполнении команды, описанной в п. 2.1.2 (см. Рисунок 4).

2.2.12. Запустите Windows PowerShell от имени администратора.

2.2.13. Выполните следующую команду, чтобы установить расширение Chocolatey:

```
Set-ExecutionPolicy Bypass -Scope Process -Force;  
[System.Net.ServicePointManager]::SecurityProtocol =  
[System.Net.ServicePointManager]::SecurityProtocol -bor 3072; iex ((New-Object  
System.Net.WebClient).DownloadString('https://community.chocolatey.org/install.ps1'))
```

2.2.14. Выполните следующую команду, чтобы установить OpenSSL.

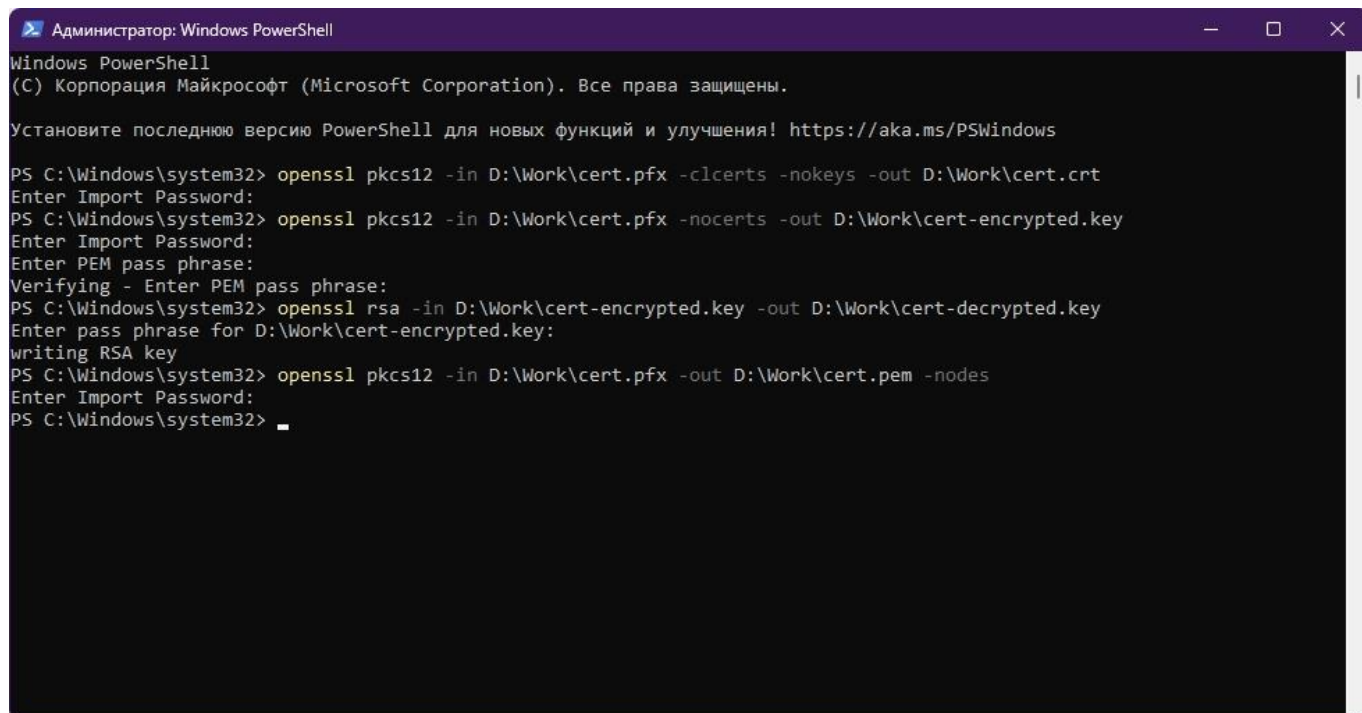
```
choco install openssl
```

2.2.15. Выполните следующие команды для генерации файлов из сертификата (см. Рисунок 8):

```
1. openssl pkcs12 -in Полный путь и имя сертификата.pfx -clcerts -nokeys -out  
Полный путь и имя сертификата.crt
```

```
Например, openssl pkcs12 -in D:\Work\cert.pfx -clcerts -nokeys -out  
D:\Work\cert.crt
```

2. `openssl pkcs12 -in Полный путь и имя сертификата.pfx -nocerts -out Полный путь и имя сертификата-encrypted.key`
3. `openssl rsa -in Полный путь и имя сертификата-encrypted.key -out Полный путь и имя сертификата-decrypted.key`
4. `openssl pkcs12 -in Полный путь и имя сертификата.pfx -out Полный путь и имя сертификата.pem -nodes`



```
Администратор: Windows PowerShell
Windows PowerShell
(C) Корпорация Майкрософт (Microsoft Corporation). Все права защищены.

Установите последнюю версию PowerShell для новых функций и улучшения! https://aka.ms/PSWindows

PS C:\Windows\system32> openssl pkcs12 -in D:\Work\cert.pfx -clcerts -nokeys -out D:\Work\cert.crt
Enter Import Password:
PS C:\Windows\system32> openssl pkcs12 -in D:\Work\cert.pfx -nocerts -out D:\Work\cert-encrypted.key
Enter Import Password:
Enter PEM pass phrase:
Verifying - Enter PEM pass phrase:
PS C:\Windows\system32> openssl rsa -in D:\Work\cert-encrypted.key -out D:\Work\cert-decrypted.key
Enter pass phrase for D:\Work\cert-encrypted.key:
writing RSA key
PS C:\Windows\system32> openssl pkcs12 -in D:\Work\cert.pfx -out D:\Work\cert.pem -nodes
Enter Import Password:
PS C:\Windows\system32> _
```

Рисунок 8

2.2.15.1. Когда Windows PowerShell запросит пароль, введите пароль, указанный при выполнении команды, описанной в п. 2.1.3.

2.2.15.2. В результате выполнения команд, описанных в п. 2.2.13, в папке появятся файлы с расширениями «.crt», «decrypted.key» и «.pem» (см. Рисунок 9).




 cert.pem	17.05.2024 12:18	Файл "PEM"	4 КБ
 cert-decrypted.key	17.05.2024 12:17	Файл "KEY"	2 КБ
 cert-encrypted.key	17.05.2024 12:15	Файл "KEY"	3 КБ
 cert.crt	17.05.2024 12:13	Сертификат безо...	2 КБ

Рисунок 9

2.3. Настройка модуля редактирования в VideoNet

2.3.1. Параметры модуля редактирования настраиваются в VideoNet на странице «Интеграция».

Страница «Интеграция» входит в группу страниц «Сеть VideoNet» среды «Конфигурирование».

2.3.1.1. Подробная информация о настройке модуля редактирования и работе со страницей «Интеграция» указана в Руководстве пользователя VideoNet.

2.3.2. Чтобы добавить модуль редактирования и обеспечить возможность его работы необходимо:

1) Выбрать в дереве элементов компьютер, который является основным сервером БД СКУД;

2) Установить флажок «Принимать запросы по REST API» в области свойств выбранного компьютера (см. Рисунок 10);

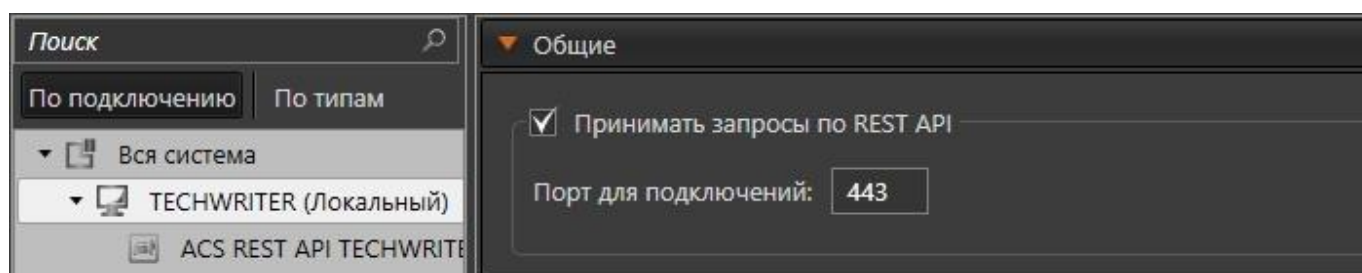


Рисунок 10

3) Установить необходимое значение параметра «Порт для подключений» в области свойств выбранного компьютера (см. Рисунок 10);

4) Нажать кнопку «Добавить» на панели инструментов страницы «Интеграция»;

5) Выбрать в выпадающем списке элемент «ACS REST API».

2.3.2.1. При добавлении модуль редактирования отобразится в дереве в виде элемента с определённым пользователем именем (по умолчанию «ACS REST API <Имя компьютера>»).

2.3.3. При выделении модуля редактирования в дереве элементов, в области свойств элементов (в правой части экрана) отобразятся его свойства.

2.3.3.1. Скриншот области свойств модуля редактирования представлен на Рисунок 11.



ACS REST API TECHWRITER	
▼ Общие	
Имя:	ACS REST API TECHWRITER
▼ Настройки	
Порт для подключений:	443

Рисунок 11

2.3.4. Чтобы заблокировать модуль редактирования и временно прекратить приём запросов необходимо выделить модуль в дереве элементов и нажать кнопку «Блокировать элемент» на панели инструментов.

3. МЕТОДЫ ДОБАВЛЕНИЯ ДАННЫХ В БД СКУД ИЗ ВНЕШНЕЙ СИСТЕМЫ

3.1. Общие сведения

3.1.1. Для добавления данных в БД СКУД Внешняя система должна отправить запрос методом POST.

3.1.2. В заголовке Authorization должны быть переданы параметры username и password (Basic авторизация).

3.1.2.1. Требуемые для авторизации параметры username и password передаются пользователям Внешней системы в административном порядке.

3.1.2.2. Пользователь VideoNet, username и password которого будут переданы для авторизации при запросе данных, должен иметь права на просмотр и редактирование БД СКУД.

3.1.2.3. В случае если username и password не будут соответствовать друг другу или будут неизвестны, Внешней системе будет передана ошибка с кодом 401.

3.1.2.4. В случае если пользователь, чей username и password использованы в запросе, не имеет прав для просмотра и редактирования БД СКУД, Внешней системе будет передана ошибка с кодом 403.

3.1.3. Параметры строки запроса отсутствуют.

3.2. Добавление подразделения в БД СКУД

3.2.1. Для добавления подразделения в БД СКУД необходимо отправить запрос методом:

POST <https://{IP:port}/acs/group>

где {IP:port} – IP-адрес и порт компьютера VideoNet, являющегося основным сервером БД СКУД.

Внимание! Добавленное таким образом подразделение будет наследовать доступ у родительского подразделения, который был назначен в среде СКУД системы VideoNet.

3.2.2. Тело запроса в формате JSON должно быть обязательно.

3.2.2.1. Описание параметров тела запроса представлено в Таблица 1.

Таблица 1

№	Имя атрибута	Описание поля	Тип	Обяз.	Примечание
1.	GroupName	Название группы (подразделения)	string	+	
2.	ParentGroupId	Уникальный идентификатор родительской группы (подразделения)	GUID	-	Указывается, если группа (подразделение) входит в состав другой группы.

3.2.2.2. Пример тела запроса в формате JSON:

```
{  
  "GroupName": "Бухгалтерия",  
  "ParentGroupId": "6F9619FF-8B86-D011-B42D-00CF4FC964FF"  
}
```

3.2.2.3. В случае если указанный ParentGroupId будет отсутствовать в БД СКУД, будет сформирован ответ с кодом 422 и телом ответа, содержащим значение некорректного ParentGroupId.

```
{  
  "FailedParentGroupId": "6F9619FF-8B86-D011-B42D-00CF4FC964FF"  
}
```

3.2.3. Пример URL с заполненными параметрами для выполнения метода:

<https://192.168.0.1:443/acs/group>

3.2.4. В результате успешного добавления в БД СКУД группы (подразделения) будет сформирован ответ с кодом 201 и телом ответа в формате JSON.

3.2.4.1. Описание структуры ответа представлено в Таблица 2.

Таблица 2

№	Имя атрибута	Описание поля	Тип	Обяз.	Примечание
1.	GroupId	Уникальный идентификатор группы (подразделения)	GUID	+	Возвращается назначенный GUID
2.	GroupName	Название группы (подразделения)	string	+	
3.	ParentGroupId	Уникальный идентификатор родительской группы (подразделения)	GUID	-	Указывается, если группа (подразделение) входит в состав другой группы.

3.2.4.2. Пример тела ответа с кодом 201 в формате JSON:

```
{
  "GroupId": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
  "GroupName": "Бухгалтерия",
  "ParentGroupId": "6F9619FF-8B86-D011-B42D-00CF4FC964FF"
}
```

3.3. Добавление должности в БД СКУД

3.3.1. Для добавления должности в БД СКУД необходимо отправить запрос методом:

POST <https://{IP:port}/acs/position>

где {IP:port} – IP-адрес и порт компьютера VideoNet, являющегося основным сервером БД СКУД.

3.3.2. Тело запроса в формате JSON должно быть обязательно.

3.3.2.1. Описание параметров тела запроса представлено в Таблица 3.

Таблица 3

№	Имя атрибута	Описание поля	Тип	Обяз.	Примечание
1.	PositionName	Название должности	string	+	
2.	PositionDescription	Описание должности	string	-	

3.3.2.2. Пример тела запроса в формате JSON:

```
{
  "PositionName": "Главный бухгалтер"
}
```

3.3.3. Пример URL с заполненными параметрами для выполнения метода:

<https://192.168.0.1:443/acs/position>

3.3.4. В результате успешного добавления в БД СКУД должности будет сформирован ответ с кодом 201 и телом ответа в формате JSON.

3.3.4.1. Описание структуры ответа представлено в Таблица 4.

Таблица 4

№	Имя атрибута	Описание поля	Тип	Обяз.	Примечание
1.	PositionId	Уникальный идентификатор должности	GUID	+	Возвращается назначенный GUID
2.	PositionName	Название должности	string	+	
3.	PositionDescription	Описание должности	string	-	

3.3.4.2. Пример тела ответа с кодом 201 в формате JSON.

```
{
  "PositionId": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
  "PositionName": "Главный бухгалтер"
}
```

3.4. Добавление человека в БД СКУД

3.4.1. Для добавления человека в БД СКУД необходимо отправить запрос методом:

POST <https://{IP:port}/acs/person>

где {IP:port} – IP-адрес и порт компьютера VideoNet, являющегося основным сервером БД СКУД.

3.4.2. Тело запроса в формате JSON должно быть обязательно.

3.4.2.1. Описание параметров тела запроса представлено в Таблица 5.

Таблица 5

№	Имя атрибута	Описание поля	Тип	Обяз.	Примечание
1.	Surname	Фамилия человека	string	+	
2.	Name	Имя человека	string	-	
3.	Patronymic	Отчество человека	string	-	
4.	Number	Табельный номер	string	-	
5.	Comment	Комментарий к данным человека	string	-	
6.	PositionId	Уникальный идентификатор должности	GUID	-	
7.	GroupId	Уникальный идентификатор группы (подразделения)	GUID	-	

3.4.2.2. Пример тела запроса в формате JSON:

```
{
  "Surname ": "Петров",
  "Name ": "Петр",
  "Patronymic": "Петрович",
  "Number": "112"
}
```

3.4.2.3. В случае если запрос будет содержать Number, который уже есть в БД СКУД, будет сформирован ответ с кодом 422 и телом ответа в формате JSON с параметрами человека (см. Таблица 6), у которого такой же табельный номер.

```
{
  "PersonId": "6F9619FF-8B86-D011-B42D-00CF4FC964DD",
  "Surname ": "Семенов",
  "Name ": "Сергей",
  "Patronymic": "Петрович",
  "Number": "112"
}
```

3.4.2.4. В случае если указанные PositionId и/или GroupId будут отсутствовать в БД СКУД, будет сформирован ответ с кодом 422 и телом ответа в формате JSON, содержащий отсутствующие в БД СКУД значения PositionId и/или GroupId. Добавление человека в БД СКУД с параметрами запроса НЕ производится.

```
{
  "PositionId": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
  "GroupId": "8F9619FF-8B86-D011-B42D-00CF4FC964DD"
}
```

3.4.3. Пример URL с заполненными параметрами для выполнения метода:

<https://192.168.0.1:443/acs/person>

3.4.4. В результате успешного добавления в БД СКУД человека будет сформирован ответ с кодом 201 и телом ответа в формате JSON.

3.4.4.1. Описание структуры ответа представлено в Таблица 6.

Таблица 6

№	Имя атрибута	Описание поля	Тип	Обяз.	Примечание
1.	PersonId	Уникальный идентификатор человека	GUID	+	Возвращается назначенный GUID
2.	Surname	Фамилия человека	string	+	
3.	Name	Имя человека	string	-	Возвращается переданное в БД СКУД значение
4.	Patronymic	Отчество человека	string	-	
5.	Number	Табельный номер	string	-	
6.	Comment	Комментарий к данным человека	string	-	
7.	PositionId	Уникальный идентификатор должности	GUID	-	
8.	GroupId	Уникальный идентификатор группы (подразделения)	GUID	-	

3.4.4.2. Пример тела ответа с кодом 201 в формате JSON:

```
{
  "PersonId": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
```

```

    "Surname ": "Петров",
    "Name ": "Петр",
    "Patronymic": "Петрович",
    "Number": "112"
  }

```

3.5. Добавление автомобиля в БД СКУД

3.5.1. Для добавления автомобиля в БД СКУД необходимо отправить запрос методом:

POST <https://{IP:port}/acs/car>

где {IP:port} – IP-адрес и порт компьютера VideoNet, являющегося основным сервером БД СКУД.

3.5.2. Тело запроса в формате JSON должно быть обязательно.

3.5.2.1. Описание параметров тела запроса представлено в Таблица 7.

Таблица 7

№	Наименование атрибута	Наименование поля	Тип данных	Обяз.	Примечание
1.	CarNumber	Номер	string	+	
2.	Color	Цвет	int	-	Возможные значения см. в п. 3.5.2.2. При отсутствии параметра будет установлено значение 0
3.	Model	Модель	string	-	
4.	Comment	Комментарий к данным автомобиля	string	-	

3.5.2.2. Перечень возможных значений параметра Color (тип int), представлен в Таблица 8.

Таблица 8

№ п/п	Color	Название цвета
1.	0	Отсутствует значение цвета в параметрах автомобиля
2.	1	Черный
3.	2	Серебристый
4.	3	Белый
5.	4	Серый
6.	5	Синий
7.	6	Красный
8.	7	Зеленый
9.	8	Коричневый
10.	9	Бежевый
11.	10	Голубой
12.	11	Золотой
13.	12	Пурпурный
14.	13	Фиолетовый
15.	14	Желтый
16.	15	Оранжевый
17.	16	Розовый

3.5.2.3. Пример тела запроса в формате JSON:

```
{
  "CarNumber": "Y140KX178",
  "Color": 3,
  "Model": "VW Polo"
}
```

3.5.2.4. В случае если запрос будет содержать CarNumber, который уже есть в БД СКУД, будет сформирован ответ с кодом 422 и телом ответа в формате JSON с параметрами автомобиля (см. Таблица 9), у которого такой же номер. Добавление автомобиля в БД СКУД с параметрами запроса НЕ производится.

3.5.3. Пример URL с заполненными параметрами для выполнения метода:

<https://192.168.0.1:443/acs/car>

3.5.4. В результате успешного добавления в БД СКУД автомобиля будет сформирован ответ с кодом 201 и телом ответа в формате JSON.

3.5.4.1. Описание структуры ответа представлено в Таблица 9.

Таблица 9

№	Наименование атрибута	Наименование поля	Тип данных	Обяз.	Примечание
1.	CarId	Уникальный идентификатор автомобиля	GUID	+	Возвращается назначенный GUID
2.	CarNumber	Номер	string	+	
3.	Color	Цвет	int	+	Возможные значения см. в п. 3.5.2.2. Возвращается переданное в БД СКУД значение
4.	Model	Модель	string	-	Возвращается переданное в БД СКУД значение
5.	Comment	Комментарий к данным автомобиля	string	-	Возвращается переданное в БД СКУД значение

3.5.4.2. Пример тела ответа с кодом 201 в формате JSON:

```
{
  "CarId": "6F9619FF-8B86-D011-B42D-00CF4FC96400",
  "CarNumber": "Y140KX178",
  "Color": 3,
  "Model": " VW Polo"
}
```

4. МЕТОДЫ ПОЛУЧЕНИЯ ДАННЫХ ИЗ БД СКУД

4.1. Общие сведения

4.1.1. Внешняя система для получения данных из БД СКУД должна отправить запрос методом GET.

4.1.2. Порядок авторизации указан в п. 3.1.2.

4.1.3. Ответ на запрос может содержать более одной страницы.

4.1.3.1. Полное количество страниц ответов указывается в параметре ответа TotalPages.

4.1.3.2. Для обращения к необходимой странице ответа необходимо повторно обратиться с запросом, указав в значении параметра Page требуемый номер. Нумерация страниц начинается с 0.

Пример: TotalPages = 5. Для получения всех страниц ответа необходимо последовательно отправить запросы с Page = 0,1,2,3,4.

4.2. Запрос списка групп доступа

4.2.1. Для получения списка групп доступа необходимо отправить запрос методом:

GET `https://{IP:port}/acs/accessgroupslist`

где {IP:port} – IP-адрес и порт компьютера VideoNet, являющегося основным сервером БД СКУД.

4.2.2. Параметр строки запроса может отсутствовать.

4.2.2.1. Описание параметра строки запроса представлено в Таблица 10.

Таблица 10

№	Наименование атрибута	Описание	Тип данных	Обяз.	Примечание
1.	Page	Страница данных, которую необходимо вернуть.	int	-	От 0 и далее. При отсутствии параметра будет возвращена страница 0 (первая страница).

4.2.3. Пример URL с заполненными параметрами для выполнения метода:

<https://192.168.0.1:443/acs/accessgroupslist?Page=0>

4.2.4. В результате успешной обработки запроса будет сформирован ответ с кодом 200 и телом ответа в формате JSON.

4.2.4.1. Описание структуры ответа представлено в Таблица 11.

Таблица 11

№	Имя атрибута	Описание поля	Тип	Обяз.	Примечание
1.	TotalNumber	Общее количество записей (групп доступа) в БД СКУД	int	+	
2.	TotalNumberOnPage	Количество записей на текущей странице	int	+	
3.	TotalPages	Всего страниц с ответами	int	+	
4.	Page	Номер текущей страницы	int	+	Соответствует номеру запрошенной страницы
5.	AccessGroupsList	Массив с описанием групп доступа БД СКУД	AccessGroup	+	Множество записей о группах доступа. Описание блока AccessGroup см. в п. 4.2.4.2

4.2.4.2. Блок данных AccessGroup содержит параметры, описанные в Таблица 12.

Таблица 12

№	Наименование атрибута	Наименование поля	Тип данных	Обяз.	Примечание
1.	AccessGroupId	Уникальный идентификатор группы доступа	GUID	+	
2.	NameGroup	Имя группы доступа	string	+	

№	Наименование атрибута	Наименование поля	Тип данных	Обяз.	Примечание
3.	Comment	Комментарий к группе доступа	string	-	Если значение отсутствует в БД СКУД, то параметр может отсутствовать в ответе.

4.2.4.3. Пример тела ответа в формате JSON:

```
{
  "TotalNumber": 2,
  "TotalNumberOnPage": 2,
  "TotalPages ": 1,
  "Page": 0,
  "AccessGroupsList": [
    {
      "AccessGroupId": "6F9619FF-8B86-D011-B42D-00CF4FC964DD",
      "NameGroup": "Отдел продаж для сотрудников"
    },
    {
      "AccessGroupId": "6F9619FF-8B86-D011-B42D-00CF4FC964D9",
      "NameGroup": "Въезд на парковку",
      "Comment": "Группа только для сотрудников"
    }
  ]
}
```

4.3. Запрос данных о группе доступа

4.3.1. Для получения данных о группе доступа необходимо отправить запрос методом:

GET <https://{IP:port}/acs/accessgroup/{AccessGroupId}>

где {IP:port} – IP-адрес и порт компьютера VideoNet, являющегося основным сервером БД СКУД;

где {AccessGroupId} – уникальный идентификатор группы доступа.

4.3.1.1. В случае если запрос будет содержать AccessGroupId, который отсутствует в БД СКУД, будет сформирован ответ с кодом 404.

4.3.2. Параметры строки запроса отсутствуют.

4.3.3. Пример URL с заполненными параметрами для выполнения метода:

<https://192.168.0.1:443/acs/accessgroup/6F9619FF-8B86-D011-B42D-00CF4FC964DD>

4.3.4. В результате успешной обработки запроса будет сформирован ответ с кодом 200 и телом ответа в формате JSON.

4.3.4.1. Описание структуры ответа представлено в Таблица 13.

Таблица 13

№	Имя атрибута	Описание поля	Тип	Обяз.	Примечание
1.	AccessGroup	Описание группы	AccessGroup	+	Описание блока AccessGroup см. в п. 4.2.4.2
2.	AccessPointsList	Список устройств доступа, входящих в группу доступа	AccessPoint	+	Описание структуры AccessPoint представлено в п. 4.3.4.2

4.3.4.2. Описание структуры AccessPoint, передающей в теле запроса параметры точки доступа, представлено в Таблица 14.

Таблица 14

№	Имя атрибута	Описание поля	Тип	Обяз.	Примечание
1.	AccessPointId	Уникальный идентификатор устройства доступа	GUID	+	
2.	AccessPointName	Имя устройства доступа	string	+	
3.	ScheduleTemplates	Массив из уникальных идентификаторов шаблонов расписаний, с которыми связано устройство доступа в данной группе доступа	ScheduleTemplateId	+	

4.3.4.3. Пример тела ответа в формате JSON:

```
{
  "AccessGroup": {
    "AccessGroupId": "6F9619FF-8B86-D011-B42D-00CF4FC964DD",
    "NameGroup": "Отдел продаж для сотрудников"
  },
  "AccessPointsList": [
    {
      "AccessPointId": "6F9619FF-8B86-D011-B42D-00CF4FC944D5",
      "AccessPointName": "Вход в столовую",
      "ScheduleTemplates": [ "6F9619FF-8B86-D011-B42D-00CF4FC944B3" ]
    },
    {
      "AccessPointId": "6F9619FF-8B86-D011-B42D-00CF4FC944G7",
      "AccessPointName": "Вход в совещалку",
      "ScheduleTemplates": [ "6F9619FF-8B86-D011-B42D-00CF4FC944B3" ]
    },
    ....
  ]
}
```

4.4. Запрос списка сотрудников

4.4.1. Для получения списка сотрудников необходимо отправить запрос методом:

GET <https://{IP:port}/acs/personslist>

где {IP:port} – IP-адрес и порт компьютера VideoNet, являющегося основным сервером БД СКУД.

4.4.2. Параметр строки запроса может отсутствовать.

4.4.2.1. Описание параметра строки запроса представлено в Таблица 15.

Таблица 15

№	Наименование атрибута	Описание	Тип данных	Обяз.	Примечание
1.	Page	Страница данных, которую необходимо вернуть.	int	-	От 0 и далее. При отсутствии параметра будет возвращена страница 0 (первая страница).

4.4.3. Пример URL с заполненными параметрами для выполнения метода:

<https://192.168.0.1:443/acs/personslist?Page=0>

4.4.4. В результате успешной обработки запроса будет сформирован ответ с кодом 200 и телом ответа в формате JSON.

4.4.4.1. Описание структуры ответа представлено в Таблица 16.

Таблица 16

№	Имя атрибута	Описание поля	Тип	Обяз.	Примечание
1.	TotalNumber	Общее количество записей (сотрудников) в БД СКУД	int	+	
2.	TotalNumberOnPage	Количество записей на текущей странице	int	+	
3.	TotalPages	Всего страниц с ответами	int	+	
4.	Page	Номер текущей страницы	int	+	Соответствует номеру запрошенной страницы
5.	PersonsList	Массив с описанием сотрудников	PersonResult	+	Множество записей о сотрудниках. Описание блока PersonResult см. в п. 4.4.4.2

4.4.4.2. Описание структуры PersonResult, передающей в теле запроса параметры человека, представлено в Таблица 17.

Таблица 17

№	Имя атрибута	Описание поля	Тип	Обяз.	Примечание
1.	PersonId	Уникальный идентификатор человека	GUID	+	
2.	Name	Имя человека	string	-	
3.	Patronymic	Отчество человека	string	-	
4.	Surname	Фамилия человека	string	+	
5.	Number	Табельный номер	string	-	
6.	PositionId	Уникальный идентификатор должности	GUID	-	
7.	PositionName	Название должности	string	-	
8.	GroupId	Уникальный идентификатор группы (подразделения), в которую определен человек	GUID	-	
9.	GroupName	Название группы (подразделения), в которую определен человек	string	-	
10.	Comment	Комментарий к данным человека	string	-	
11.	PersonalPhoneNumber	Номер личного телефона	string	-	
12.	WorkPhoneNumber	Номер рабочего телефона	string	-	
13.	Email	Адрес электронной почты	string	-	Если почтовых адресов несколько, то они будут отделены друг от друга точкой с запятой

4.4.4.3. Пример тела ответа в формате JSON:

```
{
  "TotalNumber": 1200,
  "TotalNumberOnPage": 100,
  "TotalPages ": 11,
  "Page": 0,
```

```
"PersonsList": [  
  {  
    "PersonId": "5615E6F9-79C3-4623-9B18-21C194FEC2CC",  
    "Name": "Сергей",  
    "Patronymic": "Иванович",  
    "Surname": "Петров",  
    "Number": "3541",  
    "PositionId": "55A0A0A1-570F-4A89-AA86-0AE7AB62E4FD",  
    "PositionName": "Охранник",  
    "GroupId": "372082C9-49C7-4A6D-959D-AD8B480D9FF9",  
    "GroupName": "Отдел безопасности"  
  },  
  {  
    "PersonId": "5615E6F9-79C3-4623-9B18-21C194FEC2CC",  
    "Name": "Сергей",  
    "Patronymic": "Иванович",  
    "Surname": "Петров",  
    "Number": "3541",  
    "PositionId": "55A0A0A1-570F-4A89-AA86-0AE7AB62E4FD",  
    "PositionName": "Охранник",  
    "GroupId": "372082C9-49C7-4A6D-959D-AD8B480D9FF9",  
    "GroupName": "Отдел безопасности"  
  }  
  .....  
  .....  
]  
}
```

4.5. Запрос данных о человеке

4.5.1. Для получения данных о человеке необходимо отправить запрос методом:

GET <https://{IP:port}/acs/person/{PersonId}>

где {IP:port} – IP-адрес и порт компьютера VideoNet, являющегося основным сервером БД СКУД;

где {PersonId} – уникальный идентификатор человека.

4.5.1.1. В случае если запрос будет содержать PersonId, который отсутствует в БД СКУД, будет сформирован ответ с кодом 404.

4.5.2. Параметры строки запроса отсутствуют.

4.5.3. Пример URL с заполненными параметрами для выполнения метода:

<https://192.168.0.1:443/acs/person/5615E6F9-79C3-4623-9B18-21C194FEC2CC>

4.5.4. В результате успешной обработки запроса будет сформирован ответ с кодом 200 и телом ответа в формате JSON.

4.5.4.1. Описание структуры ответа представлено в Таблица 18.

Таблица 18

№	Имя атрибута	Описание поля	Тип	Обяз.	Примечание
1.	PersonDescription	Блок описания параметров человека	PersonResult	+	Описание блока данных представлено в п. 4.4.4.2.
2.	Blocked	Признак блокировки доступа	bool	+	Признак блокировки доступа человеку. true – доступ блокирован; false – доступ разрешен по определенным для человека группам доступа

№	Имя атрибута	Описание поля	Тип	Обяз.	Примечание
3.	AccessGroups	Массив из GUID групп доступа, по которым необходимо предоставлять доступ	AccessGroupId	-	Отсутствие массива или элементов в нем означает, что у человека нет доступа на объект.
4.	KeysData	Массив с номерами ключей, закреплённых за человеком	KeyId	-	Массив может отсутствовать, если человеку не выданы ключи.

4.5.4.2. Пример тела ответа в формате JSON:

```
{
  "PersonDescription":
  {
    "PersonId": "5615E6F9-79C3-4623-9B18-21C194FEC2CC",
    "Name": "Сергей",
    "Patronymic": "Иванович",
    "Surname": "Петров",
    "Number": "3541",
    "PositionId": "55A0A0A1-570F-4A89-AA86-0AE7AB62E4FD",
    "PositionName": "Охранник",
    "GroupId": "372082C9-49C7-4A6D-959D-AD8B480D9FF9",
    "GroupName": "Отдел безопасности"
  },
  "Blocked": false,
  "AccessGroups": [
    "6F9619FF-8B86-D011-B42D-00CF4FC964DD", "6F9619FF-8BB6-D011-B42D-00CF4FC964FF"
  ]
}
```



```
"KeysData": [ "F24D3D59-0BF8-49B5-A156-FC03939AD0F9", " DC6B701B-1C48-4235-AFAB-B7C88BEF8C69" ]
```

4.6. Запрос представительской фотографии человека

4.6.1. Для получения представительской фотографии человека, размещённой в свойствах человека на закладке «Общая информация» в среде СКУД, необходимо отправить запрос методом:

GET <https://{IP:port}/acs/person/mainphoto/{PersonId}>

где {IP:port} – IP-адрес и порт компьютера VideoNet, являющегося основным сервером БД СКУД;

где {PersonId} – уникальный идентификатор человека.

4.6.1.1. В случае если запрос будет содержать PersonId, который отсутствует в БД СКУД, будет сформирован ответ с кодом 404.

4.6.1.2. В случае если запрос будет содержать PersonId, который есть в БД СКУД, но у человека нет представительской фотографии, будет сформирован ответ с кодом 404 и телом ответа в формате JSON. Пример тела ответа в формате JSON:

```
{  
  "PersonId": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",  
  "MainPhoto": 0  
}
```

4.6.2. Параметры строки запроса отсутствуют.

4.6.3. Пример URL с заполненными параметрами для выполнения метода:

<https://192.168.0.1:443/acs/person/mainphoto/6F9619FF-8B86-D011-B42D-00CF4FC964FF>

4.6.4. В результате успешной обработки запроса будет сформирован ответ с кодом 200 и телом ответа в формате JSON.

4.6.4.1. Описание структуры ответа представлено в Таблица 19.

Таблица 19

№	Имя атрибута	Описание поля	Тип	Обяз.	Примечание
1.	PersonId	Уникальный идентификатор человека	GUID	+	
2.	PhotoId	Уникальный идентификатор фотографии	GUID	+	
3.	Image	Изображение в стандарте Base64	string	+	

4.6.4.2. Пример тела ответа в формате JSON:

```
{
  "PersonId": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
  "PhotoId": "ef75425f-8b04-4073-b683-ca9e86cba70a",
  "Image": "iVBORw0KGgoAAAANSUhEUgA
AAAOAAAAKCAYAAACNM+s+9AAAAABmJLR0QA/wD/AP+gvaeTAAAB3RJ
TUUH1ggDCwMADQ4NnwAAAFVJREFUGJWNkMEJADEIBEc bSDkXUnfSg
nBVeZ8LSAjiwjyEQXSFEIcHGP9oAi+H0Bymgx9MhxbFdZE2a0s9kT
Zdw01ZhhYkABS wgmf1Z6r1SNyff4BZ+ZUExcNUQUAAAAASUVORK5
CYII="
}
```

4.7. Запрос всех фотографий, закреплённых за человеком

4.7.1. Для получения всех фотографий человека, сохранённых в БД СКУД, необходимо отправить запрос методом:

GET <https://{IP:port}/acs/person/allphotos/{PersonId}>

где {IP:port} – IP-адрес и порт компьютера VideoNet, являющегося основным сервером БД СКУД;

где {PersonId} – уникальный идентификатор человека.

4.7.1.1. В случае если запрос будет содержать PersonId, который отсутствует в БД СКУД, будет сформирован ответ с кодом 404.

4.7.1.2. В случае если запрос будет содержать PersonId, который есть в БД СКУД, но у человека нет закреплённых фотографий, будет сформирован ответ с кодом 422 и телом ответа в формате JSON. Пример тела ответа в формате JSON:

```
{
  "PersonId": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
  "TotalPhotos": 0
}
```

4.7.1. Параметры строки запроса отсутствуют.

4.7.2. Пример URL с заполненными параметрами для выполнения метода:

<https://192.168.0.1:443/acs/person/allphotos/6F9619FF-8B86-D011-B42D-00CF4FC964FF>

4.7.3. В результате успешной обработки запроса будет сформирован ответ с кодом 200 и телом ответа в формате JSON.

4.7.3.1. Описание структуры ответа представлено в Таблица 20.

Таблица 20

№	Имя атрибута	Описание поля	Тип	Обяз.	Примечание
1.	PersonId	Уникальный идентификатор человека	GUID	+	
2.	TotalPhotos	Количество фото, закреплённых за человеком	int	+	
3.	Photos	Массив из фотографий человека	Photo	+	Множество фотографий должно соответствовать параметру TotalPhotos. Описание блока Photo см. в п. 4.7.3.2

4.7.3.2. Блок данных Photo содержит параметры, описанные в Таблица 21.

Таблица 21

№	Наименование атрибута	Описание	Тип данных	Обяз.	Примечание
1.	PhotoId	Уникальный идентификатор фотографии в БД СКУД	GUID	+	

№	Наименование атрибута	Описание	Тип данных	Обяз.	Примечание
2.	Image	Изображение фотографии в стандарте Base64	string	+	

4.7.3.3. Пример тела ответа в формате JSON:

```
{
  "PersonId": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
  "TotalPhotos": 2,
  "Photos": [
    { "PhotoId": "ef75425f-8b04-4073-b683-ca9e86cba70a",
      "Image": "iVBORw0KGgoAAAANSUhEUgAAAAoAAAAKCA=" },
    { "PhotoId": "6394a1ea-de74-405b-87e2-b08f4e8b3da2",
      "Image": "iVBORw0KGgoAAAANSUhEUgAAAAoOOOOOGFGHHJJKKHGHGJG=" }
  ]
}
```

4.8. Запрос списка автомобилей

4.8.1. Для получения списка автомобилей необходимо отправить запрос методом:

GET <https://{IP:port}/acs/carslist>

где {IP:port} – IP-адрес и порт компьютера VideoNet, являющегося основным сервером БД СКУД.

4.8.2. Параметр строки запроса может отсутствовать.

4.8.2.1. Описание параметра строки запроса представлено в Таблица 22.

Таблица 22

№	Наименование атрибута	Описание	Тип данных	Обяз.	Примечание
1.	Page	Страница данных, которую необходимо вернуть.	int	-	От 0 и далее. При отсутствии параметра будет возвращена страница 0 (первая страница)

4.8.3. Пример URL с заполненными параметрами для выполнения метода:

<https://192.168.0.1:443/acs/carslist?Page=0>

4.8.4. В результате успешной обработки запроса будет сформирован ответ с кодом 200 и телом ответа в формате JSON.

4.8.4.1. Описание структуры ответа представлено в Таблица 23.

Таблица 23

№	Имя атрибута	Описание поля	Тип	Обяз.	Примечание
1.	TotalNumber	Общее количество записей (автомобилей) в БД СКУД	int	+	
2.	TotalNumberOnPage	Количество записей на текущей странице	int	+	
3.	TotalPages	Всего страниц с ответами	int	+	
4.	Page	Номер текущей страницы	int	+	Соответствует номеру запрошенной страницы
5.	CarsList	Массив с описанием автомобилей	CarResult	+	Множество записей об автомобилях. Описание блока CarResult см. в п. 4.8.4.2

4.8.4.2. Описание структуры CarResult, передающей в теле запроса параметры автомобиля, представлено в Таблица 24.

Таблица 24

№	Наименование атрибута	Наименование поля	Тип данных	Обяз.	Примечание
1.	CarId	Уникальный идентификатор автомобиля	GUID	+	
2.	CarNumber	Номер	string	+	
3.	Color	Цвет	int	+	Возможные значения см. в п. 3.5.2.2.
4.	Model	Модель	string	-	Значение обязательно при наличии данных в БД СКУД
5.	Comment	Комментарий к данным автомобиля	string	-	Значение обязательно при наличии данных в БД СКУД

4.8.4.1. Пример тела ответа в формате JSON:

```

{
  "TotalNumber": 1200,
  "TotalNumberOnPage": 1000,
  "TotalPages ": 2,
  "Page": 0,
  "CarsList": [
    {
      "CarId": "6F9619FF-8B86-D011-B42D-00CF4FC96400",
      "CarNumber": "Y140KX178",
      "Color": 3,
      "Model": " VW Polo"
    },
    {
      "CarId": "6F9619FF-8B86-D011-B42C-00CF4FC96415",
      "CarNumber": "Y145YX178",
      "Color": 0,

```

```

    "Model": "VW Polo"
  },
  .....
  .....
  ]
}

```

4.9. Запрос данных об автомобиле

4.9.1. Для получения данных об автомобиле необходимо отправить запрос методом:

GET <https://{IP:port}/acs/car/{CarId}>

где {IP:port} – IP-адрес и порт компьютера VideoNet, являющегося основным сервером БД СКУД;

где {CarId} – уникальный идентификатор автомобиля.

4.9.1.1. В случае если запрос будет содержать CarId, который отсутствует в БД СКУД, будет сформирован ответ с кодом 404.

4.9.2. Параметры строки запроса отсутствуют.

4.9.3. Пример URL с заполненными параметрами для выполнения метода:

<https://192.168.0.1:443/acs/car/6F9619FF-8B86-D011-B42D-00CF4FC964FF>

4.9.4. В результате успешной обработки запроса будет сформирован ответ с кодом 200 и телом ответа в формате JSON.

4.9.4.1. Описание структуры ответа представлено в Таблица 25.

Таблица 25

№	Имя атрибута	Описание поля	Тип	Обяз.	Примечание
1.	Car	Блок описания параметров автомобиля	CarResult	+	Описание блока данных CarResult представлено в п. 4.8.4.2

№	Имя атрибута	Описание поля	Тип	Обяз.	Примечание
2.	Blocked	Признак блокировки доступа	bool	+	Признак блокировки доступа автомобилю. true – доступ блокирован; false – доступ разрешён по определённым группам доступа.
3.	AccessGroups	Массив из GUID групп доступа, по которым необходимо предоставлять доступ	AccessGroupId	-	Отсутствие массива или элементов в нем означает, что у автомобиля нет доступа на объект.
4.	KeysData	Массив с номерами ключей, закрепленных за автомобилем	KeyId	-	Массив может отсутствовать, если автомобилю не выданы ключи.

4.9.4.2. Пример тела ответа в формате JSON:

```
{
  "Car":{
    "CarId":"6F9619FF-8B86-D011-B42D-00CF4FC96400",
    "CarNumber":"Y140KX178",
    "Color":3,
    "Model":"VW Polo"
  },
  "Blocked":false,
  "AccessGroups":[
    "6F9619FF-8B86-D011-B42D-00CF4FC964DD", "6F9619FF-8BB6-D011-B42D-00CF4FC964FF"
  ],
}
```



```
"KeysData":[ "6AFA3248-2A3B-4DB5-9552-F58947666605" ]
}
```

4.10. Запрос данных о ключах

4.10.1. Для получения списка ключей необходимо отправить запрос методом:

GET <https://{IP:port}/acs/keyslist>

где {IP:port} – IP-адрес и порт компьютера VideoNet, являющегося основным сервером БД СКУД.

4.10.2. Параметр строки запроса может отсутствовать.

4.10.2.1. Описание параметра строки запроса указано в Таблица 26.

Таблица 26

№	Наименование атрибута	Описание	Тип данных	Обяз.	Примечание
1.	Page	Страница данных, которую необходимо вернуть.	int	-	От 0 и далее. При отсутствии параметра будет возвращена страница 0 (первая страница).

4.10.3. Пример URL с заполненными параметрами для выполнения метода:

<https://192.168.0.1:443/acs/keyslist?Page=1>

4.10.4. В результате успешной обработки запроса будет сформирован ответ с кодом 200 и телом ответа в формате JSON.

4.10.4.1. Описание структуры ответа представлено в Таблица 27.

Таблица 27

№	Имя атрибута	Описание поля	Тип	Обяз.	Примечание
1.	TotalNumber	Общее количество записей (ключей) в БД СКУД	int	+	
2.	TotalNumberOnPage	Количество записей на текущей странице	int	+	

№	Имя атрибута	Описание поля	Тип	Обяз.	Примечание
3.	TotalPages	Всего страниц с ответами	int	+	
4.	Page	Номер текущей страницы	int	+	Соответствует номеру запрошенной страницы
5.	KeysList	Массив с описанием ключей	KeyResult	+	Множество записей о ключах. Описание блока KeyResult см. в п. 4.10.4.2

4.10.4.2. Описание структуры KeyResult, передающей параметры ключа, представлено в Таблица 28.

Таблица 28

№	Имя атрибута	Описание поля	Тип	Обяз.	Примечание
1.	KeyId	Уникальный идентификатор ключа	GUID	+	
2.	KeyCode	Код ключа	string	+	
3.	KeyType	Тип ключа	int	+	Может принимать значение 1 – 1-Wire; 2 – Wiegand-26; 3 - Wiegand-34.
4.	Description	Описание ключа	string	-	
5.	Guest	Признак гостевого ключа	bool	-	Признак гостевого ключа: true – гостевой ключ, обязателен для ответа; false – обычный ключ.
6.	Object	Уникальный идентификатор назначенного ключу объекта (человек или автомобиль)	GUID	-	

№	Имя атрибута	Описание поля	Тип	Обяз.	Примечание
7.	ObjectType	Тип назначенного ключу объекта	int	-	Тип назначенного ключу объекта: 1-Человек; 2 -Автомобиль.

4.10.4.3. Пример тела ответа в формате JSON:

```
{
  "TotalNumber": 1002,
  "TotalNumberOnPage": 2,
  "TotalPages ": 2,
  "Page": 1,
  "KeysList": [
    {
      "KeyId": "F24D3D59-0BF8-49B5-A156-FC03939AD0F9",
      "KeyCode": "5/34234",
      "KeyType": 2,
      "Guest": true,
      "Object": "5615E6F9-79C3-4623-9B18-21C194FEC2CC",
      "ObjectType": 1
    },
    {
      "KeyId": "6AFA3248-2A3B-4DB5-9552-F58947666605",
      "KeyCode": "5/46212",
      "KeyType": 2,
      "Object": "6F9619FF-8B86-D011-B42D-00CF4FC96400",
      "ObjectType": 2
    }
  ]
}
```

4.11. Запрос списка подразделений

4.11.1. Для получения списка подразделений необходимо отправить запрос методом:

GET <https://{IP:port}/acs/groupslist>

где {IP:port} – IP-адрес и порт компьютера VideoNet, являющегося основным сервером БД СКУД.

4.11.2. Параметры строки запроса могут отсутствовать.

4.11.2.1. Описание параметра строки запроса указано в Таблица 29.

Таблица 29

№	Наименование атрибута	Описание	Тип данных	Обяз.	Примечание
1.	Page	Страница данных, которую необходимо вернуть.	int	-	От 0 и далее. При отсутствии параметра будет возвращена страница 0 (первая страница).
2.	ParentGroupId	Уникальный идентификатор родительской группы (подразделения)	GUID	-	Указывается, если необходимо получить перечень дочерних подразделений.

4.11.2.2. В случае если запрос будет содержать ParentGroupId, который отсутствует в БД СКУД, будет сформирован ответ с кодом 422 и телом ответа в формате JSON с ошибочным ParentGroupId. Пример тела ответа в формате JSON:

```
{
  "FailedParentGroupId": "6F9619FF-8B86-D011-B42D-00CF4FC964DD"
}
```

4.11.3. Пример URL с заполненными параметрами для выполнения метода:

<https://192.168.0.1:443/acs/groupslist?ParentGroupId=6F9619FF-8B86-D011-B42D-00CF4FC964FF>

4.11.4. В результате успешной обработки запроса будет сформирован ответ с кодом 200 и телом ответа в формате JSON.

4.11.4.1. Описание структуры ответа представлено в Таблица 30.

Таблица 30

№	Имя атрибута	Описание поля	Тип	Обяз.	Примечание
1.	TotalNumber	Общее количество записей (подразделений) в ответе	int	+	
2.	TotalNumberOnPage	Количество записей на текущей странице	int	+	
3.	TotalPages	Всего страниц с ответами	int	+	
4.	Page	Номер текущей страницы	int	+	Соответствует номеру запрошенной страницы
5.	ParentGroupId	Уникальный идентификатор родительской группы (подразделения)	GUID	-	Указывается, только если параметр передавался в запросе
6.	GroupsList	Массив с описанием подразделений	GroupResult	+	Множество записей о подразделениях. Описание блока GroupResult см. в п. 4.11.4.2

4.11.4.2. Описание структуры GroupResult, передающей в теле ответа параметры подразделения, представлено в Таблица 31.

Таблица 31

№	Наименование атрибута	Наименование поля	Тип данных	Обяз.	Примечание
1.	GroupId	Уникальный идентификатор группы (подразделения)	GUID	+	

№	Наименование атрибута	Наименование поля	Тип данных	Обяз.	Примечание
2.	GroupName	Название группы (подразделения)	string	+	
3.	ParentGroupId	Уникальный идентификатор родительской группы (подразделения)	GUID	-	Указывается только если группа (подразделение) входит в состав другой группы.

4.11.4.3. Пример тела ответа в формате JSON:

```
{
  "TotalNumber": 2,
  "ParentGroupId": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
  "GroupsList": [
    {
      "GroupId": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
      "GroupName": "Бухгалтерия",
      "ParentGroupId": "6F9619FF-8B86-D011-B42D-00CF4FC964FF"
    },
    {
      "GroupId": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
      "GroupName": "Планово-экономический отдел",
      "ParentGroupId": "6F9619FF-8B86-D011-B42D-00CF4FC964FF"
    }
  ]
}
```

4.12. Запрос списка должностей

4.12.1. Для получения списка должностей необходимо отправить запрос методом:

GET <https://{IP:port}/acs/positionslist>

где {IP:port} – IP-адрес и порт компьютера VideoNet, являющегося основным сервером БД СКУД.

4.12.2. Параметр строки запроса может отсутствовать.

4.12.2.1. Описание параметра строки запроса представлено в Таблица 32.

Таблица 32

№	Наименование атрибута	Описание	Тип данных	Обяз.	Примечание
1.	Page	Страница данных, которую необходимо вернуть.	int	-	От 0 и далее. При отсутствии параметра будет возвращена страница 0 (первая страница).

4.12.3. Пример URL с заполненными параметрами для выполнения метода:

<https://192.168.0.1:443/acs/positionslist>

4.12.4. В результате успешной обработки запроса будет сформирован ответ с кодом 200 и телом ответа в формате JSON.

4.12.4.1. Описание структуры ответа представлено в Таблица 33.

Таблица 33

№	Имя атрибута	Описание поля	Тип	Обяз.	Примечание
1.	TotalNumber	Общее количество записей (должностей) в БД СКУД	int	+	
2.	TotalNumberOnPage	Количество записей на текущей странице	int	+	
3.	TotalPages	Всего страниц с ответами	int	+	
4.	Page	Номер текущей страницы	int	+	Соответствует номеру запрошенной страницы

№	Имя атрибута	Описание поля	Тип	Обяз.	Примечание
5.	PositionsList	Массив с описанием должностей	PositionsResult	+	Множество записей о должностях. Описание блока PositionsResult см. в п. 4.12.4.2

4.12.4.2. Описание структуры PositionsResult, передающей в теле ответа параметры должности, представлено в Таблица 34.

Таблица 34

№	Наименование атрибута	Наименование поля	Тип данных	Обяз.	Примечание
1.	PositionId	Уникальный идентификатор должности	GUID	+	
2.	PositionName	Название должности	string	+	
3.	PositionDescription	Описание должности	string	-	

4.12.4.3. Пример тела ответа в формате JSON:

```
{
  "TotalNumber": 2,
  "PositionsList": [
    {
      "PositionId": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
      "PositionName": "Начальник бухгалтерии",
      "PositionDescription": "Высшее образование"
    },
    {
      "PositionId": "8474B03C-8DC4-45E9-91DC-EFBEB023E934",
      "PositionName": "Техник"
```



```
}  
  ]  
}
```

5. МЕТОДЫ ДЛЯ ИЗМЕНЕНИЯ ДАННЫХ В БД СКУД

5.1. Общие сведения

5.1.1. Внешняя система для добавления данных в БД СКУД должна отправить запрос методом PUT.

5.1.2. Порядок авторизации указан в п. 3.1.2.

5.2. Изменение параметров человека

5.2.1. Для изменения параметров человека в БД СКУД необходимо отправить запрос методом:

PUT `https://{IP:port}/acs/person/{PersonId}`

где {IP:port} – IP-адрес и порт компьютера VideoNet, являющегося основным сервером БД СКУД;

где {PersonId} – Уникальный идентификатор человека.

5.2.2. Параметры строки запроса отсутствуют.

5.2.3. Тело запроса в формате JSON должно быть обязательно.

5.2.3.1. Описание параметров тела запроса представлено в Таблица 35.

Таблица 35

№	Имя атрибута	Описание поля	Тип	Обяз.	Примечание
1.	Surname	Фамилия человека	string	+	
2.	Name	Имя человека	string	-	
3.	Patronymic	Отчество человека	string	-	
4.	Number	Табельный номер	string	-	
5.	Comment	Комментарий к данным человека	string	-	
6.	PositionId	Уникальный идентификатор должности	GUID	-	
7.	GroupId	Уникальный идентификатор группы (подразделения)	GUID	-	

5.2.3.2. Пример тела запроса в формате JSON:

```
{  
  "Surname ": "Семенов",  
  "Name ": "Петр",  
  "Patronymic": "Петрович",  
  "Number": "112"  
}
```

5.2.3.3. В случае если запрос будет содержать Number, который уже есть в БД СКУД у другого человека, будет сформирован ответ с кодом 422 и телом ответа в формате JSON с параметрами человека (см. Таблица 36), у которого такой же табельный номер. Пример тела ответа в формате JSON:

```
{  
  "PersonId": "6F9619FF-8B86-D011-B42D-00CF4FC964DD",  
  "Surname ": "Семенов",  
  "Name ": "Сергей",  
  "Patronymic": "Петрович",  
  "Number": "112"  
}
```

5.2.3.4. В случае если указанные PositionId и/или GroupId будут отсутствовать в БД СКУД, будет сформирован ответ с кодом 422 и телом ответа в формате JSON, содержащим отсутствующие в БД СКУД значения PositionId и/или GroupId. Изменение параметров человека в БД СКУД НЕ производится. Пример тела ответа в формате JSON:

```
{  
  "PositionId": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",  
  "GroupId": "8F9619FF-8B86-D011-B42D-00CF4FC964DD"  
}
```

5.2.4. Пример URL с заполненными параметрами для выполнения метода:

<https://192.168.0.1:443/acs/person>

5.2.5. В результате успешного изменения параметров человека в БД СКУД будет сформирован ответ с кодом 201 и телом ответа в формате JSON.

5.2.5.1. Описание структуры ответа представлено в Таблица 36.

Таблица 36

№	Имя атрибута	Описание поля	Тип	Обяз.	Примечание
1.	PersonId	Уникальный идентификатор человека	GUID	+	Возвращаются сохранённые в БД СКУД значения. Если значения отсутствуют, то параметры могут отсутствовать в ответе.
2.	Surname	Фамилия человека	string	+	
3.	Name	Имя человека	string	-	
4.	Patronymic	Отчество человека	string	-	
5.	Number	Табельный номер	string	-	
6.	Comment	Комментарий к данным человека	string	-	
7.	PositionId	Уникальный идентификатор должности	GUID	-	
8.	GroupId	Уникальный идентификатор группы (подразделения)	GUID	-	

5.2.5.2. Пример тела ответа с кодом 201 в формате JSON:

```
{
  "PersonId": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
  "Surname ": "Семенов",
  "Name ": "Сергей",
  "Patronymic": "Петрович",
  "Number": "112"
}
```

5.3. Изменение параметров автомобиля

5.3.1. Для изменения параметров автомобиля в БД СКУД необходимо отправить запрос методом:

PUT `https://{IP:port}/acs/car/{CarId}`

где {IP:port} – IP-адрес и порт компьютера VideoNet, являющегося основным сервером БД СКУД;

где {CarId} – уникальный идентификатор автомобиля.

5.3.2. Параметры строки запроса отсутствуют.

5.3.3. Тело запроса в формате JSON должно быть обязательно.

5.3.3.1. Описание параметров тела запроса представлено в Таблица 37.

Таблица 37

№	Наименование атрибута	Наименование поля	Тип данных	Обяз.	Примечание
1.	CarNumber	Номер	string	+	
2.	Color	Цвет	int	-	Возможные значения см. в п. 3.5.2.2. При отсутствии параметра будет установлено значение 0
3.	Model	Модель	string	-	
4.	Comment	Комментарий к данным автомобиля	string	-	

5.3.3.2. Пример тела запроса в формате JSON:

```
{
  "CarNumber": "Y140KX178",
  "Color": 4,
  "Model": " VW Polo"
}
```

5.3.3.3. В случае если запрос будет содержать CarId, который уже есть в БД СКУД, данные будут заменены на данные из запроса.

5.3.3.4. В случае если запрос будет содержать CarNumber, который уже есть в БД СКУД у другого автомобиля, будет сформирован ответ с кодом 422 и телом ответа в формате JSON с параметрами автомобиля (см. Таблица 38), у которого такой же номер. Изменение параметров автомобиля в БД СКУД НЕ производится.

5.3.4. Пример URL с заполненными параметрами для выполнения метода:

<https://192.168.0.1:443/acs/car/6F9619FF-8B86-D011-B42D-00CF4FC96400>

5.3.5. В результате успешного изменения параметров автомобиля в БД СКУД будет сформирован ответ с кодом 201 и телом ответа в формате JSON.

5.3.5.1. Описание структуры ответа представлено в Таблица 38.

Таблица 38

№	Наименование атрибута	Наименование поля	Тип данных	Обяз.	Примечание
1.	CarId	Уникальный идентификатор автомобиля	GUID	+	
2.	CarNumber	Номер	string	+	
3.	Color	Цвет	int	+	Возможные значения см. в п. 3.5.2.2. Возвращается сохранённое в БД СКУД значение.
4.	Model	Модель	string	-	Возвращается сохранённое в БД СКУД значение. Если значение отсутствует, то параметр может отсутствовать в ответе.
5.	Comment	Комментарий к данным автомобиля	string	-	

5.3.5.2. Пример тела ответа с кодом 201 в формате JSON:

```
{
  "CarId": "6F9619FF-8B86-D011-B42D-00CF4FC96400",
  "CarNumber": "Y140KX178",
  "Color": 4,
  "Model": "VW Polo"
```

}

5.4. Назначение человеку прав доступа на объект

5.4.1. Для назначения человеку прав доступа на объект необходимо отправить запрос методом:

PUT `https://{IP:port}/acs/person/{PersonId}/add_accessgroup`

где {IP:port} – IP-адрес и порт компьютера VideoNet, являющегося основным сервером БД СКУД;

где {PersonId} – Уникальный идентификатор человека.

5.4.1.1. В случае если запрос будет содержать PersonId, который отсутствует в БД СКУД, будет сформирован ответ с кодом 404.

5.4.2. Параметры строки запроса отсутствуют.

5.4.3. Тело запроса в формате JSON должно быть обязательно.

5.4.3.1. Описание параметров тела запроса представлено в Таблица 39.

Таблица 39

№	Имя атрибута	Описание поля	Тип	Обяз.	Примечание
1.	AccessGroupId	GUID группы доступа, по которой необходимо предоставлять доступ	GUID	+	Группа доступа добавляется к уже ранее назначенным для объекта. Аннулирование назначенных объекту групп доступа производится по отдельному запросу (см. 5.5).
2.	Loading	Признак необходимости загрузки контроллеров, участвующих в предоставлении доступа	bool	-	Если параметр отсутствует в запросе, то его значение будет определено как false. Подробнее о параметре см. п. 5.4.3.2.

5.4.3.2. Прогрузка контроллеров изменениями прав доступа осуществляется по определенным в системе правилам. При этом запрет доступа обеспечивается сразу после его аннулирования.

При изменении прав доступа прогрузка контроллеров может быть отложена и произведена в определенное в системе время (раз в сутки).

Для предоставления доступа по изменённым правилам как можно скорее в запросе параметру `Loading` необходимо установить значение `true`. Тогда контроллеры, которых касаются изменения доступа, будут прогружены в ближайшее время (от 15 до 90 секунд с момента обработки запроса).

5.4.3.3. Пример тела запроса в формате JSON:

```
{  
  "AccessGroupId": "6F9619FF-8B86-D011-B42D-00CF4FC964DD",  
  "Loading": true  
}
```

5.4.3.4. В случае если запрос будет содержать `AccessGroupId`, который отсутствует в БД СКУД, будет сформирован ответ с кодом 422 и телом ответа в формате JSON с ошибочным `AccessGroupId`. Пример тела ответа в формате JSON:

```
{  
  "FailedAccessGroupId": "6F9619FF-8B86-D011-B42D-00CF4FC964DD"  
}
```

5.4.4. Пример URL с заполненными параметрами для выполнения метода:

https://192.168.0.1:443/acs/person/6F9619FF-8B86-D011-B42D-00CF4FC964FF/add_accessgroup

5.4.5. В результате назначения человеку прав доступа на объект будет сформирован ответ с кодом 201 и телом ответа в формате JSON.

5.4.5.1. Описание структуры ответа представлено в Таблица 40.

Таблица 40

№	Имя атрибута	Описание поля	Тип	Обяз.	Примечание
1.	PersonId	Уникальный идентификатор человека	GUID	+	
2.	Loading	Признак необходимости загрузки контроллеров, участвующих в предоставлении доступа	bool	+	Соответствует значению параметра в запросе
3.	AccessGroups	Массив из GUID групп доступа, по которым определён доступ	AccessGroupId	+	Указываются группы доступа, назначенные объекту

5.4.5.2. Пример тела ответа с кодом 201 в формате JSON.

```
{
  "PersonId": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
  "Loading": true,
  "AccessGroups":[
    "6F9619FF-8B86-D011-B42D-00CF4FC964DD", "6F9619FF-8BB6-D011-B42D-00CF4FC964FF"
  ]
}
```

5.5. Аннулирование прав доступа на объект у человека

5.5.1. Для аннулирования (отмены назначения) всех назначенных групп доступа (прав доступа) у человека в БД СКУД необходимо отправить запрос методом:

PUT https://{IP:port}/acs/person/{PersonId}/cancel_accessgroups

где {IP:port} – IP-адрес и порт компьютера VideoNet, являющегося основным сервером БД СКУД;

где {PersonId} – Уникальный идентификатор человека.

5.5.1.1. В случае если запрос будет содержать PersonId, который отсутствует в БД СКУД, будет сформирован ответ с кодом 404.

Внимание! Будет аннулирован только тот доступ, который был назначен непосредственно человеку. Доступ, который назначен в среде СКУД системы VideoNet для подразделений и должностей, и наследуется человеком, отменен не будет. Для отмены наследуемого доступа необходимо воспользоваться средой СКУД системы VideoNet.

5.5.2. Параметры строки запроса отсутствуют.

5.5.3. Тело запроса в формате JSON отсутствует.

5.5.4. Пример URL с заполненными параметрами для выполнения метода:

https://192.168.0.1:443/acs/person/6F9619FF-8B86-D011-B42D-00CF4FC964FF/cancel_accessgroups

5.5.5. В результате успешного аннулирования прав доступа на объект у человека будет сформирован ответ с кодом 201 и телом ответа в формате JSON.

5.5.5.1. Описание структуры ответа представлено в Таблица 41.

Таблица 41

№	Имя атрибута	Описание поля	Тип	Обяз.	Примечание
1.	PersonId	Уникальный идентификатор человека, у которого аннулирован доступ	GUID	+	

5.5.5.2. Пример тела ответа с кодом 201 в формате JSON:

```
{
  "PersonId": "6F9619FF-8B86-D011-B42D-00CF4FC964FF"
}
```

5.6. Назначение автомобилю прав доступа на объект

5.6.1. Для назначения автомобилю прав доступа на объект необходимо отправить запрос методом:

PUT https://{IP:port}/acs/car/{CarId}/add_accessgroup

где {IP:port} – IP-адрес и порт компьютера VideoNet, являющегося основным сервером БД СКУД;

где {CarId} – уникальный идентификатор автомобиля.

5.6.1.1. В случае если запрос будет содержать CarId, который отсутствует в БД СКУД, будет сформирован ответ с кодом 404.

5.6.2. Параметры строки запроса отсутствуют.

5.6.3. Тело запроса в формате JSON должно быть обязательно.

5.6.3.1. Описание параметров тела запроса представлено в Таблица 42.

Таблица 42

№	Имя атрибута	Описание поля	Тип	Обяз.	Примечание
1.	AccessGroupId	GUID группы доступа, по которой необходимо предоставлять доступ	GUID	+	Группа доступа добавляется к уже ранее назначенным для объекта. Аннулирование назначенных объекту групп доступа производится по отдельному запросу (см. п. 5.7)
2.	Loading	Признак необходимости загрузки контроллеров, участвующих в предоставлении доступа	bool	-	Если параметр отсутствует в запросе, то его значение будет определено как false. Подробнее о параметре см. п. 5.4.3.2

5.6.3.2. Пример тела запроса в формате JSON:

```
{
  "AccessGroupId": "6F9619FF-8B86-D011-B42D-00CF4FC964DD",
  "Loading": true
}
```

5.6.3.3. В случае если запрос будет содержать `AccessGroupId`, который отсутствует в БД СКУД, будет сформирован ответ с кодом 422 и телом ответа в формате JSON с ошибочным `AccessGroupId`. Пример тела ответа в формате JSON:

```
{
  "FailedAccessGroupId ":"6F9619FF-8B86-D011-B42D-00CF4FC964DD"
}
```

5.6.4. Пример URL с заполненными параметрами для выполнения метода:

https://192.168.0.1:443/acs/car/6F9619FF-8B86-D011-B42D-00CF4FC964FF/add_accessgroup

5.6.5. В результате назначения автомобилю прав доступа на объект будет сформирован ответ с кодом 201 и телом ответа в формате JSON.

5.6.5.1. Описание структуры ответа представлено в Таблица 43.

Таблица 43

№	Имя атрибута	Описание поля	Тип	Обяз.	Примечание
1.	CarId	Уникальный идентификатор автомобиля	GUID	+	
2.	Loading	Признак необходимости загрузки контроллеров, участвующих в предоставлении доступа	bool	+	Если параметр отсутствовал в запросе, то в ответе его значение будет false
3.	AccessGroups	Массив из GUID групп доступа, по которым определён доступ	AccessGroupId	+	.

5.6.5.2. Пример тела ответа с кодом 201 в формате JSON:

```
{
  "CarId": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
  "Loading": true,
  "AccessGroups":[
```

```
"6F9619FF-8B86-D011-B42D-00CF4FC964DD", "6F9619FF-8BB6-D011-B42D-00CF4FC964FF"
]
}
```

5.7. Аннулирование прав доступа на объект у автомобиля

5.7.1. Для аннулирования (отмены назначения) всех назначенных групп доступа (прав доступа) у автомобиля в БД СКУД необходимо отправить запрос методом:

PUT https://{IP:port}/acs/car/{CarId}/cancel_accessgroups

где {IP:port} – IP-адрес и порт компьютера VideoNet, являющегося основным сервером БД СКУД;

где {CarId} – Уникальный идентификатор автомобиля.

5.7.1.1. В случае если запрос будет содержать PersonId, который отсутствует в БД СКУД, будет сформирован ответ с кодом 404.

5.7.2. Параметры строки запроса отсутствуют.

5.7.3. Тело запроса в формате JSON отсутствует.

5.7.4. Пример URL с заполненными параметрами для выполнения метода:

https://192.168.0.1:443/acs/car/6F9619FF-8B86-D011-B42D-00CF4FC964FF/cancel_accessgroups

5.7.5. В результате успешного аннулирования прав доступа на объект у автомобиля будет сформирован ответ с кодом 201 и телом ответа в формате JSON.

5.7.5.1. Описание структуры ответа представлено в Таблица 44.

Таблица 44

№	Имя атрибута	Описание поля	Тип	Обяз.	Примечание
1.	CarId	Уникальный идентификатор автомобиля, у которого аннулирован доступ	GUID	+	

5.7.5.2. Пример тела ответа с кодом 201 в формате JSON:

```
{
```

"CarId": "6F9619FF-8B86-D011-B42D-00CF4FC964FF"

}

5.8. Выдача ключей человеку

5.8.1. Для выдачи (назначения) ключей человеку в БД СКУД необходимо отправить запрос методом:

PUT https://{IP:port}/acs/person/{PersonId}/add_key

где {IP:port} – IP-адрес и порт компьютера VideoNet, являющегося основным сервером БД СКУД;

где {PersonId} – Уникальный идентификатор человека.

5.8.1.1. В случае если запрос будет содержать PersonId, который отсутствует в БД СКУД, будет сформирован ответ с кодом 404.

5.8.2. Параметры строки запроса отсутствуют.

5.8.3. Тело запроса в формате JSON должно быть обязательно.

5.8.3.1. Описание параметров тела запроса указано в Таблица 45.

Таблица 45

№	Имя атрибута	Описание поля	Тип	Обяз.	Примечание
1.	KeyId	Уникальный идентификатор ключа, назначаемого человеку	GUID	+	Один ключ может быть закреплён только за одним объектом. Для закрепления за человеком нескольких ключей необходимо повторить запрос с очередным/другим свободным KeyId

№	Имя атрибута	Описание поля	Тип	Обяз.	Примечание
2.	Loading	Признак необходимости загрузки контроллеров, участвующих в предоставлении доступа	bool	-	Если параметр отсутствует в запросе, то его значение будет определено как false. Подробнее о параметре см. п. 5.4.3.2

5.8.3.2. Пример тела запроса в формате JSON:

```
{
  "KeyId": "F24D3D59-0BF8-49B5-A156-FC03939AD0F9",
  "Loading": true
}
```

5.8.3.3. В случае если запрос будет содержать KeyId, который уже назначен другому объекту в БД СКУД, будет сформирован ответ с кодом 422 и телом ответа в формате JSON с параметрами структуры KeyResult (см. п. 4.10.4.2) с KeyId и уникальным идентификатором этого объекта. Пример тела ответа в формате JSON:

```
{
  "KeyId": "6AFA3248-2A3B-4DB5-9552-F58947666605",
  "KeyCode": "5/34245",
  "KeyType": 2,
  "Object": "5615E6F9-79C3-4623-9B18-21C194FEC2CC",
  "ObjectType": 1
}
```

5.8.3.4. В случае если запрос будет содержать KeyId, который отсутствует в БД СКУД, будет сформирован ответ с кодом 422 и телом ответа в формате JSON с ошибочным KeyId. Пример тела ответа в формате JSON:

```
{
  "FailedKeyId": "F24D3D59-0BF8-49B5-A156-FC03939AD0F9"
}
```

```
}
```

5.8.4. Пример URL с заполненными параметрами для выполнения метода:

https://192.168.0.1:443/acs/person/6F9619FF-8B86-D011-B42D-00CF4FC964FF/add_key

5.8.5. В результате успешной выдачи (назначения) ключей человеку будет сформирован ответ с кодом 201 и телом ответа в формате JSON.

5.8.5.1. Описание структуры ответа представлено в Таблица 46.

Таблица 46

№	Имя атрибута	Описание поля	Тип	Обяз.	Примечание
1.	PersonId	Уникальный идентификатор человека	GUID	+	
2.	Loading	Признак необходимости загрузки контроллеров, участвующих в предоставлении доступа	bool	+	Если параметр отсутствовал в запросе, то в ответе его значение будет false
3.	KeyId	Уникальный идентификатор ключа	GUID	+	

5.8.5.2. Пример тела ответа с кодом 201 в формате JSON.

```
{
```

"PersonId": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",

"Loading": true,

"KeyId": "F24D3D59-0BF8-49B5-A156-FC03939AD0F9"

```
}
```

5.9. Аннулирование выданных ключей у человека

5.9.1. Для аннулирования (отмены назначения) всех назначенных ключей у человека в БД СКУД необходимо отправить запрос методом:

PUT https://{IP:port}/acs/person/{PersonId}/cancel_keys

где {IP:port} – IP-адрес и порт компьютера VideoNet, являющегося основным сервером БД СКУД;

где {PersonId} – Уникальный идентификатор человека.

5.9.1.1. В случае если запрос будет содержать `PersonId`, который отсутствует в БД СКУД, будет сформирован ответ с кодом 404.

5.9.2. Параметры строки запроса отсутствуют.

5.9.3. Тело запроса в формате JSON отсутствует.

5.9.4. Пример URL с заполненными параметрами для выполнения метода:

https://192.168.0.1:443/acs/person/6F9619FF-8B86-D011-B42D-00CF4FC964FF/cancel_keys

5.9.5. В результате успешного аннулирования (отмены назначения) всех назначенных ключей у человека будет сформирован ответ с кодом 201 и телом ответа в формате JSON.

5.9.5.1. Описание структуры ответа представлено в Таблица 47.

Таблица 47

№	Имя атрибута	Описание поля	Тип	Обяз.	Примечание
1.	PersonId	Уникальный идентификатор человека, у которого аннулирован доступ	GUID	+	

5.9.5.2. Пример тела ответа с кодом 201 в формате JSON:

```
{
  "PersonId": "6F9619FF-8B86-D011-B42D-00CF4FC964FF"
}
```

5.10. Выдача ключей автомобилю

5.10.1. Для выдачи (назначения) ключей автомобилю в БД СКУД необходимо отправить запрос методом:

PUT https://{IP:port}/acs/car/{CarId}/add_key

где `{IP:port}` – IP-адрес и порт компьютера VideoNet, являющегося основным сервером БД СКУД;

где `{CarId}` – Уникальный идентификатор автомобиля.

5.10.1.1. В случае если запрос будет содержать `CarId`, который отсутствует в БД СКУД, будет сформирован ответ с кодом 404.

5.10.2. Параметры строки запроса отсутствуют.

5.10.3. Тело запроса в формате JSON должно быть обязательно.

5.10.3.1. Описание параметров тела запроса указано в Таблица 48.

Таблица 48

№	Имя атрибута	Описание поля	Тип	Обяз.	Примечание
1.	KeyId	Уникальный идентификатор ключа, назначаемого автомобилю	GUID	+	Один ключ может быть закреплён только за одним объектом. Для закрепления за автомобилем нескольких ключей необходимо повторить запрос с очередным/другим свободным KeyId.
2.	Loading	Признак необходимости прогрузки контроллеров, участвующих в предоставлении доступа	bool	-	Если параметр отсутствует в запросе, то его значение будет определено как false. Подробнее о параметре см. п. 5.4.3.2.

5.10.3.2. Пример тела запроса в формате JSON:

```
{
  "KeyId": "6AFA3248-2A3B-4DB5-9552-F58947666605",
  "Loading": true
}
```

5.10.3.3. В случае если запрос будет содержать KeyId, который уже назначен другому объекту в БД СКУД, будет сформирован ответ с кодом 422 и телом ответа в формате JSON с параметрами структуры KeyResult (см. п. 4.10.4.2) с KeyId и уникальным идентификатором этого объекта. Пример тела ответа в формате JSON:

```
{
```

```

    "KeyId": "6AFA3248-2A3B-4DB5-9552-F58947666605",
    "KeyCode": "5/34245",
    "KeyType": 2,
    "Object": "5615E6F9-79C3-4623-9B18-21C194FEC2CC",
    "ObjectType": 1
  }

```

5.10.3.4. В случае если запрос будет содержать KeyId, который отсутствует в БД СКУД, будет сформирован ответ с кодом 422 и телом ответа в формате JSON с ошибочным KeyId. Пример тела ответа в формате JSON:

```

{
  "FailedKeyId": "F24D3D59-0BF8-49B5-A156-FC03939AD0F9"
}

```

5.10.4. Пример URL с заполненными параметрами для выполнения метода:

https://192.168.0.1:443/acs/car/6F9619FF-8B86-D011-B42D-00CF4FC96400/add_key

5.10.5. В результате успешной выдачи (назначения) ключей автомобилю будет сформирован ответ с кодом 201 и телом ответа в формате JSON.

5.10.5.1. Описание структуры ответа представлено в Таблица 49.

Таблица 49

№	Имя атрибута	Описание поля	Тип	Обяз.	Примечание
1.	CarId	Уникальный идентификатор автомобиля	GUID	+	
2.	Loading	Признак необходимости загрузки контроллеров, участвующих в предоставлении доступа	bool	+	Если параметр отсутствовал в запросе, то в ответе его значение будет false
3.	KeyId	Уникальный идентификатор ключа	GUID	+	

5.10.5.2. Пример тела ответа с кодом 201 в формате JSON:

```

{
  "CarId": "6F9619FF-8B86-D011-B42D-00CF4FC96400",

```

```

    "Loading": true,
    "KeysData": [ "6AFA3248-2A3B-4DB5-9552-F58947666605" ]
}

```

5.11. Аннулирование выданных ключей у автомобиля

5.11.1. Для аннулирования (отмены назначения) всех назначенных ключей у автомобиля в БД СКУД необходимо отправить запрос методом:

PUT https://{IP:port}/acs/car/{CarId}/cancel_keys

где {IP:port} – IP-адрес и порт компьютера VideoNet, являющегося основным сервером БД СКУД;

где {CarId} – Уникальный идентификатор автомобиля.

5.11.2. Параметры строки запроса отсутствуют.

5.11.3. Тело запроса в формате JSON отсутствует.

5.11.3.1. В случае если запрос будет содержать PersonId, который отсутствует в БД СКУД, будет сформирован ответ с кодом 404.

5.11.4. Пример URL с заполненными параметрами для выполнения метода:

https://192.168.0.1:443/acs/car/6F9619FF-8B86-D011-B42D-00CF4FC964FF/cancel_keys

5.11.5. В результате успешного аннулирования (отмены назначения) всех назначенных ключей у автомобиля будет сформирован ответ с кодом 201 и телом ответа в формате JSON.

5.11.5.1. Описание структуры ответа представлено в Таблица 50.

Таблица 50

№	Имя атрибута	Описание поля	Тип	Обяз.	Примечание
1.	CarId	Уникальный идентификатор автомобиля, у которого аннулирован доступ	GUID	+	

5.11.5.2. Пример тела ответа с кодом 201 в формате JSON.

```

{
  "CarId": "6F9619FF-8B86-D011-B42D-00CF4FC964FF"
}

```

```
}
```

5.12. Блокировка доступа людям

5.12.1. Для блокировки доступа людям в БД СКУД необходимо отправить запрос методом:

PUT <https://{IP:port}/acs/personslist/block>

где {IP:port} – IP-адрес и порт компьютера VideoNet, являющегося основным сервером БД СКУД.

5.12.2. Параметры строки запроса отсутствуют.

5.12.3. Тело запроса в формате JSON должно быть обязательно.

5.12.3.1. Описание параметров тела запроса указано в Таблица 51.

Таблица 51

№	Имя атрибута	Описание поля	Тип	Обяз.	Примечание
1.	PersonData	Массив с уникальными идентификаторами людей, которых необходимо заблокировать	PersonId	+	

5.12.3.2. Пример тела запроса в формате JSON:

```
{
  "PersonData":["0AFF69FB-F5D9-401C-89C6-0304024A6DA0",
  "0D712037-694B-4BBE-8771-D435B02AF094 ",
  "1DC6C970-A4DC-42A3-8975-6373901D6C2B"]
}
```

5.12.4. Пример URL с заполненными параметрами для выполнения метода:

<https://192.168.0.1:443/acs/personlist/block>

5.12.5. В результате успешной блокировки доступа людям будет сформирован ответ с кодом 201 и телом ответа в формате JSON.

5.12.5.1. Описание структуры ответа представлено в Таблица 52.

Таблица 52

№	Имя атрибута	Описание поля	Тип	Обяз.	Примечание
1.	PersonData	Массив из GUID людей, которым заблокирован доступ	PersonId	-	Отсутствие массива в ответе означает, что нет людей, которым был заблокирован доступ.
2.	FailedPerson	Массив из GUID людей, которые были в запросе, но отсутствуют в БД СКУД	PersonId	-	Отсутствие массива в ответе означает, что в запросе не было людей, которые отсутствуют в БД СКУД

5.12.5.2. Пример тела ответа с кодом 201 в формате JSON.

```
{
  "PersonData":["0AFF69FB-F5D9-401C-89C6-0304024A6DA0",
  "0D712037-694B-4BBE-8771-D435B02AF094 "],
  "FailedPerson":["1DC6C970-A4DC-42A3-8975-6373901D6C2B"]
}
```

5.13. Разблокировка доступа людям

5.13.1. Для разблокировки доступа людям в БД СКУД необходимо отправить запрос методом:

PUT <https://{IP:port}/acs/personslist/unblock>

где {IP:port} – IP-адрес и порт компьютера VideoNet, являющегося основным сервером БД СКУД.

5.13.2. Параметры строки запроса отсутствуют.

5.13.3. Тело запроса в формате JSON должно быть обязательно.

5.13.3.1. Описание параметров тела запроса указано в Таблица 53.

Таблица 53

№	Имя атрибута	Описание поля	Тип	Обяз.	Примечание
1.	Loading	Признак необходимости прогрузки контроллеров, участвующих в предоставлении доступа	bool	-	Если параметр отсутствует в запросе, то его значение будет определено как false. Подробнее о параметре см. п. 5.4.3.2
2.	PersonData	Массив с уникальными идентификаторами людей, которых необходимо разблокировать	PersonId	+	

5.13.3.2. Пример тела запроса в формате JSON

```
{
  "PersonData":["0AFF69FB-F5D9-401C-89C6-0304024A6DA0",
    "0D712037-694B-4BBE-8771-D435B02AF094",
    "1DC6C970-A4DC-42A3-8975-6373901D6C2B"]
}
```

5.13.4. Пример URL с заполненными параметрами для выполнения метода:

<https://192.168.0.1:443/acs/personlist/unblock>

5.13.5. В результате успешной разблокировки доступа людям будет сформирован ответ с кодом 201 и телом ответа в формате JSON.

5.13.5.1. Описание структуры ответа представлено в Таблица 54.

Таблица 54

№	Имя атрибута	Описание поля	Тип	Обяз.	Примечание
1.	PersonData	Массив из GUID людей, которым разблокирован доступ	PersonId	-	Отсутствие массива в ответе означает, что нет людей, которым был разблокирован доступ.

№	Имя атрибута	Описание поля	Тип	Обяз.	Примечание
2.	Loading	Признак необходимости загрузки контроллеров, участвующих в предоставлении доступа	bool	+	Соответствует значению параметра в запросе
3.	FailedPerson	Массив из GUID людей, которые были в запросе, но отсутствуют в БД СКУД	PersonId	-	Отсутствие массива в ответе означает, что в запросе не было людей, которые отсутствуют в БД СКУД

5.13.5.2. Пример тела ответа с кодом 201 в формате JSON:

```
{
  "PersonData":["0AFF69FB-F5D9-401C-89C6-0304024A6DA0",
    "0D712037-694B-4BBE-8771-D435B02AF094"],
  "Loading": false,
  "FailedPerson":["1DC6C970-A4DC-42A3-8975-6373901D6C2B"]
}
```


6. МЕТОДЫ УДАЛЕНИЯ ДАННЫХ ИЗ БД СКУД

6.1. Общие сведения

6.1.1. Внешняя система для удаления данных из БД СКУД должна отправить запрос методом DELETE.

6.1.2. Порядок авторизации указан в п. 3.1.2.

6.2. Удаление подразделения из БД СКУД

6.2.1. Для удаления группы (подразделения) из БД СКУД необходимо отправить запрос методом:

DELETE <https://{IP:port}/acs/group/{GroupId}>

где {IP:port} – IP-адрес и порт компьютера VideoNet, являющегося основным сервером БД СКУД;

где {GroupId} – Уникальный идентификатор группы (подразделения).

6.2.1.1. В случае если указанный GroupId будет отсутствовать в БД СКУД, будет сформирован ответ с кодом 404.

6.2.1.2. В случае если указанный GroupId будет относиться к группе, удаление которой невозможно, будет сформирован ответ с кодом 422 и телом ответа, содержащим значение некорректного GroupId. Пример тела ответа в формате JSON:

```
{  
  "GroupId": "6F9619FF-8B86-D011-B42D-00CF4FC964FF"  
}
```

6.2.2. Параметры строки запроса отсутствуют.

6.2.3. Тело запроса в формате JSON отсутствует.

6.2.4. Пример URL с заполненными параметрами для выполнения метода:

<https://192.168.0.1:443/acs/group/6F9619FF-8B86-D011-B42D-00CF4FC964FF>

6.2.5. В результате успешного удаления группы (подразделения) из БД СКУД будет сформирован ответ с кодом 201 и телом ответа в формате JSON.

6.2.5.1. Описание структуры ответа представлено в Таблица 55.

Таблица 55

№	Имя атрибута	Описание поля	Тип	Обяз.	Примечание
1.	GroupId	Уникальный идентификатор удаленной группы (подразделения)	GUID	+	Возвращается GUID удалённой группы (подразделения)

6.2.5.2. Пример тела ответа с кодом 201 в формате JSON:

```
{
  "GroupId": "6F9619FF-8B86-D011-B42D-00CF4FC964FF",
}
```

6.3. Удаление должности из БД СКУД

6.3.1. Для удаления должности из БД СКУД необходимо отправить запрос методом:

DELETE <https://{IP:port}/acs/position/{PositionId}>

где {IP:port} – IP-адрес и порт компьютера VideoNet, являющегося основным сервером БД СКУД;

где {PositionId} – Уникальный идентификатор должности.

6.3.1.1. В случае если указанный PositionId будет отсутствовать в БД СКУД, будет сформирован ответ с кодом 404.

6.3.2. Параметры строки запроса отсутствуют.

6.3.3. Тело запроса в формате JSON отсутствует.

6.3.4. Пример URL с заполненными параметрами для выполнения метода:

<https://192.168.0.1:443/acs/position/6F9619FF-8B86-D011-B42D-00CF4FC964FF>

6.3.5. В результате успешного удаления должности из БД СКУД будет сформирован ответ с кодом 201 и телом ответа в формате JSON.

6.3.5.1. Описание структуры ответа представлено в Таблица 56.

Таблица 56

№	Имя атрибута	Описание поля	Тип	Обяз.	Примечание
1.	PositionId	Уникальный идентификатор удаленной должности	GUID	+	Возвращается удаленный GUID

6.3.5.2. Пример тела ответа с кодом 201 в формате JSON:

```
{
  "PositionId": "6F9619FF-8B86-D011-B42D-00CF4FC964FF"
}
```

6.4. Удаление человека из БД СКУД

6.4.1. Для удаления человека из БД СКУД необходимо отправить запрос методом:

DELETE https://{IP:port}/acs/person/{PersonId}

где {IP:port} – IP-адрес и порт компьютера VideoNet, являющегося основным сервером БД СКУД;

где {PersonId} – уникальный идентификатор человека.

6.4.1.1. В случае если запрос будет содержать PersonId, который отсутствует в БД СКУД, будет сформирован ответ с кодом 404.

6.4.2. Параметры строки запроса отсутствуют.

6.4.3. Тело запроса в формате JSON отсутствует.

6.4.4. Пример URL с заполненными параметрами для выполнения метода:

<https://192.168.0.1:443/acs/person/6F9619FF-8B86-D011-B42D-00CF4FC964FF>

6.4.5. В результате успешного удаления человека из БД СКУД будет сформирован ответ с кодом 201 и телом ответа в формате JSON.

6.4.5.1. Описание структуры ответа представлено в Таблица 57.

Таблица 57

№	Имя атрибута	Описание поля	Тип	Обяз.	Примечание
1.	PersonId	Уникальный идентификатор удаленного человека	GUID	+	

6.4.5.2. Пример тела ответа с кодом 201 в формате JSON:

```
{
  "PersonId": "6F9619FF-8B86-D011-B42D-00CF4FC964FF"
}
```

6.5. Удаление автомобиля из БД СКУД

6.5.1. Для удаления автомобиля из БД СКУД необходимо отправить запрос методом:

DELETE <https://{IP:port}/acs/car/{CarId}>

где {IP:port} – IP-адрес и порт компьютера VideoNet, являющегося основным сервером БД СКУД

где {CarId} – уникальный идентификатор автомобиля.

6.5.1.1. В случае если запрос будет содержать CarId, который отсутствует в БД СКУД, будет сформирован ответ с кодом 404.

6.5.2. Параметры строки запроса отсутствуют.

6.5.3. Тело запроса в формате JSON отсутствует.

6.5.4. Пример URL с заполненными параметрами для выполнения метода:

<https://192.168.0.1:443/acs/car/6F9619FF-8B86-D011-B42D-00CF4FC964FF>

6.5.5. В результате успешного удаления автомобиля из БД СКУД будет сформирован ответ с кодом 201 и телом ответа в формате JSON.

6.5.5.1. Описание структуры ответа представлено в Таблица 58.

Таблица 58

№	Имя атрибута	Описание поля	Тип	Обяз.	Примечание
1.	CarId	Уникальный идентификатор удаленного автомобиля	GUID	+	

6.5.5.2. Пример тела ответа с кодом 201 в формате JSON:

```
{  
  "CarId": "6F9619FF-8B86-D011-B42D-00CF4FC964FF"  
}
```

ПРИЛОЖЕНИЕ

ТЕСТОВОЕ ПРИЛОЖЕНИЕ

Тестовое приложение выполняет запрос на сервер VideoNet и позволяет оценить корректность выполнения настроек, указанных в разделе 2.

Исходный код на Python

Для корректной отправки запросов на сервер VideoNet на языке Python в запросе необходимо указать файлы «Имя сертификата.crt», «Имя сертификата-decrypted.key» и «Имя сертификата.pem» (см. 2.2.15.2).

Пример скрипта:

```
import requests

def main():
    url = 'https://test.videonet.ru/acs/accessgroupslist'
    r = requests.get(url,
                     auth=('admin', '111111'),
                     cert=('ssl/cert.crt', 'ssl/cert-decrypted.key'),
                     verify='ssl/cert.pem')

    print(r.status_code)

if __name__ == "__main__":
    main()
```

В результате выполнения скрипта с запросом при правильной настройке, выполненной согласно разделу 2, должен прийти код 200.

Исходный код на C#

Код написан на .NET Core 7.0 и должен размещаться в файле Program.cs.

Пример кода:

```
using System.Runtime.InteropServices;
using System.Security.Authentication;
using System.Security.Cryptography.X509Certificates;
using System.Text;

var username = "admin";
var password = "111111";
var baseAddress = "https://test.videonet.ru";
```

```
var crtFileName = "ssl/cert.crt";
var decryptedKeyFileName = "ssl/cert-decrypted.key";
var getQuery = "https://test.videonet.ru/acs/accessgroupslist";

X509Certificate2 CreateCertFromPemFile(string certPath, string keyPath)
{
    if (!RuntimeInformation.IsOSPlatform(OSPlatform.Windows))
        return X509Certificate2.CreateFromPemFile(certPath, keyPath);

    using var cert = X509Certificate2.CreateFromPemFile(certPath, keyPath);
    return new X509Certificate2(cert.Export(X509ContentType.Pkcs12));
}

var handler = new HttpClientHandler();
handler.ClientCertificateOptions = ClientCertificateOption.Manual;
handler.SslProtocols = SslProtocols.Tls12;
handler.ClientCertificates.Add(CreateCertFromPemFile(crtFileName, decryptedKeyFileName));

var client = new HttpClient(handler)
{
    BaseAddress = new Uri(baseAddress),
};

var encoded = Convert.ToBase64String(Encoding.GetEncoding("ISO-8859-1").GetBytes(username + ":" + password));

client.DefaultRequestHeaders.Add("Authorization", $"Basic {encoded}");

var result = await client.GetAsync(getQuery);

Console.WriteLine($"Result code: {(int)result.StatusCode} - {result.StatusCode}");
Console.WriteLine($"Result body: {result.Content.ReadAsStringAsync().Result}");

Console.WriteLine($"Press Enter for exit..");

Console.ReadLine();
```

При запуске скомпилированного тестового примера при правильной настройке, выполненной согласно разделу 2, должен прийти ответ на запрос с кодом 200.