

# Протокол JSON для настольного считывателя Desktop

Ревизия 1.03

## Оглавление

Описание устройства .....	4
Драйвер USB ←→ COM.....	4
Сервис для конвертации JSON запросов .....	4
Параметры подключения .....	4
Информационные команды.....	4
Информация об устройстве.....	4
Режим работы считывателя (разрешенные типы идентификаторов) .....	5
Получение текущего режима работы считывателя.....	5
Установка нового режима работы считывателя.....	5
Получение информации об оставшемся количестве мобильных идентификаторов .....	5
Получение информации о свойствах считывателя или считывателей, обслуживаемых сервисом .....	6
Получение информации об включенном режиме шифрования мобильных идентификаторов.....	6
Чтение кодов идентификаторов .....	7
Описание форматов вывода .....	7
Получения свойства идентификатора .....	11
Процедура чтения существующего мобильного идентификатора с помощью Desktop .....	12
Выпуск мобильных идентификаторов.....	13
Локальная выдача идентификаторов.....	13
Выдача идентификаторов по e-mail .....	13
Описание форматов вывода .....	16
Отправка по e-mail .....	16
Дополнительная защита систем доступа (персонификация идентификаторов).....	17
Определение, включен ли режим шифрования идентификаторов .....	17
Определение, включен ли режим шифрования мобильных идентификаторов.....	17
Установка режима шифрования мобильных идентификаторов.....	17
Включение режима шифрования и установка пароля шифрования .....	17
Выключение режима шифрования и очистка пароля.....	18
Установка режима шифрования для карт Mifare® .....	18
Установка режима шифрования Mifare® Classic.....	18
Установка режима шифрования с персонифицированным номером идентификатора Mifare® Classic.....	18
Отключение режима шифрования Mifare® Classic .....	19
Установка режима шифрования Mifare® Plus.....	19
Установка режима шифрования с персонифицированным номером идентификатора Mifare® Plus.....	19
Отключение режима шифрования Mifare Plus .....	20
Выпуск зашифрованной карты Mifare® Classic.....	20
Выпуск зашифрованной карты Mifare® Plus.....	21
Стирание зашифрованной карты .....	21
Настройка отправки электронной почты сервисом JSON при выдаче мобильных идентификаторов по e-mail ..	22

Нерабочие запросы.....	23
Получение и установка выходного формата кода идентификатора (*не реализовано).....	23
Получение и установка индикации считывателя (*не реализовано).....	23
Получение токена авторизации при удаленном доступе (*заморожено).....	23
Удаленная загрузка и обновление микропрограммы (*не реализовано).....	23
Получение и установка Facility кода для старых карт FSK (*не реализовано).....	23
Установка пароля шифрования для BLE (*устарело, заморожено).....	23

## Описание устройства

Настольный считыватель ASK / FSK / Mifare и поддержкой BLE с USB интерфейсом и открытым текстовым протоколом предназначен для максимально простой интеграции в любую систему СКУД.

Может использоваться для выдачи мобильных идентификаторов из выделенного списка мобильных идентификаторов.

### Характеристики

RF ID ASK, FSK, ASK+FSK (125 кГц)
Mifare (ISO 14443A - 13,56 МГц)
Чтение мобильных идентификаторов
Выдача мобильных идентификаторов из выделенного списка мобильных идентификаторов. Переход в режим выдачи по мастер-карте.
Обмен данными со смартфонами по Bluetooth Low Energy (Bluetooth Smart, BLE)
Используется зашифрованный 256-битным ключом и стойкий к саботажу протокол для обмена данными со смартфонами
Программирование Desktop по BLE со смартфона с Android с помощью ПО PW Config по BLE - устанавливается размерность (битность) идентификаторов, загружаются лицензии на мобильные идентификаторы.

### Драйвер USB ↔ COM

Для эмуляции COM порта в системе используется SyLabs CP210x USB to UART мост.

Драйвера можно скачать по следующей ссылке:

<http://www.silabs.com/products/development-tools/software/usb-to-uart-bridge-vcp-drivers>

### Сервис для конвертации JSON запросов

Для работы с настольным считывателем в протоколе JSON предназначено программное обеспечение Acs.DesktopService.exe. Текущая поддерживаемая платформа – Windows, необходимо наличие .NET4.5

Для поддержки работы в режиме эмуляции клавиатуры используется Acs.DesktopService.Admin.exe

### Параметры подключения

POST запросы на <http://127.0.0.1:40011>

## Информационные команды

### Информация об устройстве

С помощью данной команды можно получить информацию об устройстве - его серийном номере и версии микропрограммы

#### Запрос:

```
http://localhost:40011/GetVersion
```

#### Тело запроса:

```
{  
}
```

#### Ответ:

```
{  
  "Model": "DESKTOP READER",  
  "Serial": "00:32:00:17:00:20",  
  "Version": "09.35"  
}
```

## Режим работы считывателя (разрешенные типы идентификаторов)

С помощью данной команды можно включить или отключить работу (чтение и запись) с определенными типами идентификаторов.

Получение текущего режима работы считывателя

### Запрос:

```
http://localhost:40011/GetReaderType
```

### Тело запроса:

```
{  
}
```

### Ответ:

```
{  
  "ASKEEnabled": true,  
  "BLEEnabled": true,  
  "FSKEEnabled": true,  
  "INNEEnabled": false,  
  "MifareEnabled": true,  
  "MifarePlusEnabled": false  
}
```

Установка нового режима работы считывателя

### Запрос:

```
http://localhost:40011/SetReaderType
```

### Тело запроса:

```
{  
  "ASKEEnabled": true,  
  "BLEEnabled": true,  
  "FSKEEnabled": true,  
  "INNEEnabled": false,  
  "MifareEnabled": true,  
  "MifarePlusEnabled": true  
}
```

### Ответ:

```
{  
  "ASKEEnabled": true,  
  "BLEEnabled": true,  
  "FSKEEnabled": true,  
  "INNEEnabled": false,  
  "MifareEnabled": true,  
  "MifarePlusEnabled": true  
}
```

## Получение информации об оставшемся количестве мобильных идентификаторов

Устройство используется для выдачи мобильных идентификаторов из выделенного списка. Базовая поставка включает 5 бесплатных идентификаторов.

С помощью данной команды можно посмотреть количество неиспользованных лицензий в списке мобильных идентификаторов.

### Запрос:

```
http://localhost:40011/GetQuantity
```

### Тело запроса:

```
{  
}
```

### Ответ:

```
{  
  "Quantity": 58  
}
```

Получение информации о свойствах считывателя или считывателей, обслуживаемых сервисом  
Поддерживается несколько типов считывателей, ANVIS – FingerPrintSupported, Biosmart – PalmVeinSupported,  
Desktop RFID карты – RFIDSUPPORTED, Desktop мобильные идентификаторы - IssuingMobileIdSupported

**Запрос:**

```
http://localhost:40011/GetQuantity
```

**Тело запроса:**

```
{  
}
```

**Ответ:**

```
{  
  "FingerPrintSupported": false,  
  "IssuingMobileIdSupported": true,  
  "PalmVeinSupported": false,  
  "RFIDSUPPORTED": true  
}
```

Получение информации об включенном режиме шифрования мобильных идентификаторов  
Проверка, используется ли дополнительный пароль шифрования при генерации и выпуске мобильных  
идентификаторов.

**Запрос:**

```
http://localhost:40011/GetBleEncryption
```

**Тело запроса:**

```
{  
}
```

**Ответ:**

```
{  
  " Encrypted": false  
}
```

## Чтение кодов идентификаторов

С помощью данной команды можно получить код идентификатора в заданных форматах.

### Запрос:

```
http://127.0.0.1:40011/GetIdentifiers
```

### Тело запроса по умолчанию, выводимый формат кода hex5:

```
{  
}
```

### Ответ:

```
{  
  "Identifiers": [ "String content", "String content", "String content" ]  
}
```

## Описание форматов вывода

### "Raw" – без преобразований, как получено от считывателя

### Запрос:

```
http://127.0.0.1:40011/GetIdentifiers
```

### Тело запроса:

```
{  
  "OutputAs": "Raw"  
}
```

### Получение данных:

```
{  
  "Identifiers": ["Em-Marine[565A] 123,16044"]  
}
```

### Например:

```
{  
  "Identifiers": [  
    "Mifare[E4009A14] 000,39444 1K (0004,08)",  
    "Mifare[04B62AE1BB0280] 182,10977 UL (0044,00)",  
    "Mifare[C2ECA99C] 236,43420 4K (0002,18)",  
    "Mifare[04210D21B21C80] 033,03361 DF (0344,20)",  
    "Em-Marine[565A] 123,16044",  
    "FSK[00120040] 08111",  
    "FSK[0010029A] 38193",  
    "MobileID[1122334455667788]"  
  ]  
}
```

### Где:

- Mifare – карточки Mifare® стандарта ISO14443A,
- Em-Marine – карточки с амплитудной модуляцией (ASK),
- FSK – карточки с частотной модуляцией
- MobileID – мобильный идентификатор, полученный от смартфона.

### "FacilityCode" – отображение в формате facility\_code. Код усекается до 3-х байт, старшая часть отбрасывается.

### Запрос:

```
http://127.0.0.1:40011/GetIdentifiers
```

### Тело запроса:

```
{  
  "OutputAs": "FacilityCode"  
}
```

### Получение данных:

```
{
  "Identifiers":["123,16044"]
}
```

**"Dec3" – отображение кода в десятичном формате. Код усекается до 3-х байт, старшая часть отбрасывается.**

**Запрос:**

<http://127.0.0.1:40011/GetIdentifiers>

**Тело запроса:**

```
{
  "OutputAs": "Dec3"
}
```

**Получение данных:**

```
{
  "Identifiers":["08076972"]
}
```

**"Dec3nz" – отображение кода в десятичном формате. Код усекается до 3-х байт, старшая часть отбрасывается, ведущие нули отбрасываются**

**Запрос:**

<http://127.0.0.1:40011/GetIdentifiers>

**Тело запроса:**

```
{
  "OutputAs": "Dec3"
}
```

**Получение данных:**

```
{
  "Identifiers":["8076972"]
}
```

**"Dec4" – отображение кода в десятичном формате. Код усекается до 4-х байт, старшая часть отбрасывается.**

**Запрос:**

<http://127.0.0.1:40011/GetIdentifiers>

**Тело запроса:**

```
{
  "OutputAs": "Dec4"
}
```

**Получение данных:**

```
{
  "Identifiers":["1208076972"]
}
```

**"Dec4nz" – отображение кода в десятичном формате. Код усекается до 4-х байт, старшая часть отбрасывается, ведущие нули отбрасываются**

**Запрос:**

<http://127.0.0.1:40011/GetIdentifiers>

**Тело запроса:**

```
{
  "OutputAs": "Dec4nz"
}
```

**Получение данных:**

```
{
  "Identifiers":["1208076972"]
}
```

**"Hex3" – отображение кода в шестнадцатеричном формате. Код усекается до 3-х байт, старшая часть отбрасывается.**

**Запрос:**

```
http://127.0.0.1:40011/GetIdentifiers
```

**Тело запроса:**

```
{
  "OutputAs": "Hex3"
}
```

**Получение данных:**

```
{
  "Identifiers": ["7B3EAC"]
}
```

**"Hex3r" – отображение кода в шестнадцатеричном формате. Код выводится в обратном порядке, усекается до 3-х байт, старшая часть отбрасывается.**

**Запрос:**

```
http://127.0.0.1:40011/GetIdentifiers
```

**Тело запроса:**

```
{
  "OutputAs": "Hex3"
}
```

**Получение данных:**

```
{
  "Identifiers": ["AC3E7B"]
}
```

**"Hex4r" – отображение кода в шестнадцатеричном формате. Код выводится в обратном порядке, если код длиннее - усекается до 4-х байт (старшая часть отбрасывается) если короче – дополняется нулями (старшая часть)**

**Запрос:**

```
http://127.0.0.1:40011/GetIdentifiers
```

**Тело запроса:**

```
{
  "OutputAs": "Hex4r"
}
```

**Получение данных:**

```
{
  "Identifiers": ["AC5A3E7B"]
}
```

**Hex4" – отображение кода в шестнадцатеричном формате. Если код длиннее - усекается до 4-х байт (старшая часть отбрасывается) если короче – дополняется нулями (старшая часть)**

**Запрос:**

```
http://127.0.0.1:40011/GetIdentifiers
```

**Тело запроса:**

```
{
  "OutputAs": "Hex4"
}
```

**Получение данных:**

```
{  
  "Identifiers": ["5A7B3EAC"]  
}
```

**"Hex5" — отображение кода в шестнадцатеричном формате. Если код длиннее - усекается до 5-х байт, если короче – дополняется нулями (старшая часть)**

**Запрос:**

```
http://127.0.0.1:40011/GetIdentifiers
```

**Тело запроса:**

```
{  
  "OutputAs": "Hex5"  
}
```

**Получение данных (карточка Em-Marine):**

```
{  
  "Identifiers": ["565A7B3EAC"]  
}
```

**Получение данных (карточка FSK):**

```
{  
  "Identifiers": ["000010029A"]  
}
```

**"Hex5f3" — отображение кода в шестнадцатеричном формате. Код усекается до 3-х байт (старшая часть отбрасывается) и далее дополняется нулями до 5-ти байт (старшая часть)**

**Запрос:**

```
http://127.0.0.1:40011/GetIdentifiers
```

**Тело запроса:**

```
{  
  "OutputAs": "Hex5f3"  
}
```

**Получение данных (карточка Em-Marine):**

```
{  
  "Identifiers": ["00007B3EAC"]  
}
```

**"Hex6" – отображение кода в шестнадцатеричном формате. Если код длиннее - усекается до 6-ти байт (старшая часть отбрасывается) если короче – дополняется нулями (старшая часть)**

**Запрос:**

```
http://127.0.0.1:40011/GetIdentifiers
```

**Тело запроса:**

```
{  
  "OutputAs": "Hex6"  
}
```

**Получение данных (карточка Em-Marine):**

```
{  
  "Identifiers": ["00565A7B3EAC"]  
}
```

**Получение данных (мобильный идентификатор):**

```
{  
  "Identifiers": ["334455667788"]  
}
```

**"Hex7" – отображение кода в шестнадцатеричном формате. Если код длиннее - усекается до 7-ми байт (старшая часть отбрасывается) если короче – дополняется нулями (старшая часть)**

**Запрос:**

```
http://127.0.0.1:40011/GetIdentifiers
```

**Тело запроса:**

```
{
  "OutputAs": "Hex7"
}
```

**Получение данных (карточка Em-Marine):**

```
{
  "Identifiers": ["0000565A7B3EAC"]
}
```

**Получение данных (мобильный идентификатор):**

```
{
  "Identifiers": ["22334455667788"]
}
```

**"Hex8" – отображение кода в шестнадцатеричном формате. Если код длиннее - усекается до 8-ми байт (старшая часть отбрасывается) если короче – дополняется нулями (старшая часть)**

**Запрос:**

```
http://127.0.0.1:40011/GetIdentifiers
```

**Тело запроса:**

```
{
  "OutputAs": "Hex8"
}
```

**Получение данных (карточка Em-Marine):**

```
{
  "Identifiers": ["000000565A7B3EAC"]
}
```

**Получение данных (мобильный идентификатор):**

```
{
  "Identifiers": ["1122334455667788"]
}
```

Получения свойства идентификатора

**Запрос:**

```
http://127.0.0.1:40011/GetIdentifiers
```

**Тело запроса:**

```
{
  "Raw": "Mifare[4A011F33] 001,07987 DF (0004,20)"
}
```

**Получение данных:**

```
{
  "CardType": "Mifare",
  "Dec": "1241587507",
  "Dec3": "00073523",
  "Dec3nz": "73523",
  "Dec4": "1241587507",
  "Dec4nz": "1241587507",
  "FacilityCode": "001,07987",
  "Hex3": "011F33",
  "Hex3r": "331F01",
  "Hex4": "4A011F33",
}
```

```
"Hex4r": "331F014A",
"Hex5": "004A011F33",
"Hex5f3": "0000011F33",
"Hex6": "00004A011F33",
"Hex7": "0000004A011F33",
"Hex8": "000000004A011F33",
"MifareType": "DF",
"Raw": "Mifare[4A011F33] 001,07987 DF (0004,20)"
}
```

### Процедура чтения существующего мобильного идентификатора с помощью Desktop

1. Пользователь подносит смартфон к Desktop , и запускает или активирует мобильное приложение PW ID
2. Настольный считыватель получает код от смартфона
3. Код полученного мобильного идентификатора выводится в COM порт
4. Сервис JSON сохраняет полученный код в памяти и далее по http POST-запросу <http://127.0.0.1:40011/GetIdentifiers> передает в приложение.
5. Тип идентификатора и различные варианты кода можно получить с помощью запроса <http://127.0.0.1:40011/GetIdentifierInfo>

## Выпуск мобильных идентификаторов

Устройство используется для выдачи мобильных идентификаторов из выделенного списка. Базовая поставка включает 5 бесплатных идентификаторов.

Код идентификатора будет получен также, как и при обычном чтении кодов, с помощью запроса <http://127.0.0.1:40011/GetIdentifiers>.

### Локальная выдача идентификаторов

#### Процедура локальной выдачи нового идентификатора с помощью PW Desktop

1. Администратор системы переводит настольный считыватель в режим выдачи идентификаторов – размещает на считывателе мастер-карту.
2. Пользователь подносит смартфон к PW Desktop, и в настройках приложения PW ID нажимает кнопку "Получить от Desktop"- приложение соединяется и получает мобильный идентификатор из выделенного списка мобильных идентификаторов.
3. Код полученного мобильного идентификатора выводится в COM порт
4. Сервис JSON сохраняет полученный код в памяти и далее по http POST-запросу <http://127.0.0.1:40011/GetIdentifiers> передает в приложение.

#### Запрос:

```
http://127.0.0.1:40011/GetIdentifiers
```

#### Тело запроса:

```
{
  "OutputAs": "Raw"
}
```

#### Получение данных:

```
{
  "Identifiers": ["NewMobileID[1122334455667788]"]
}
```

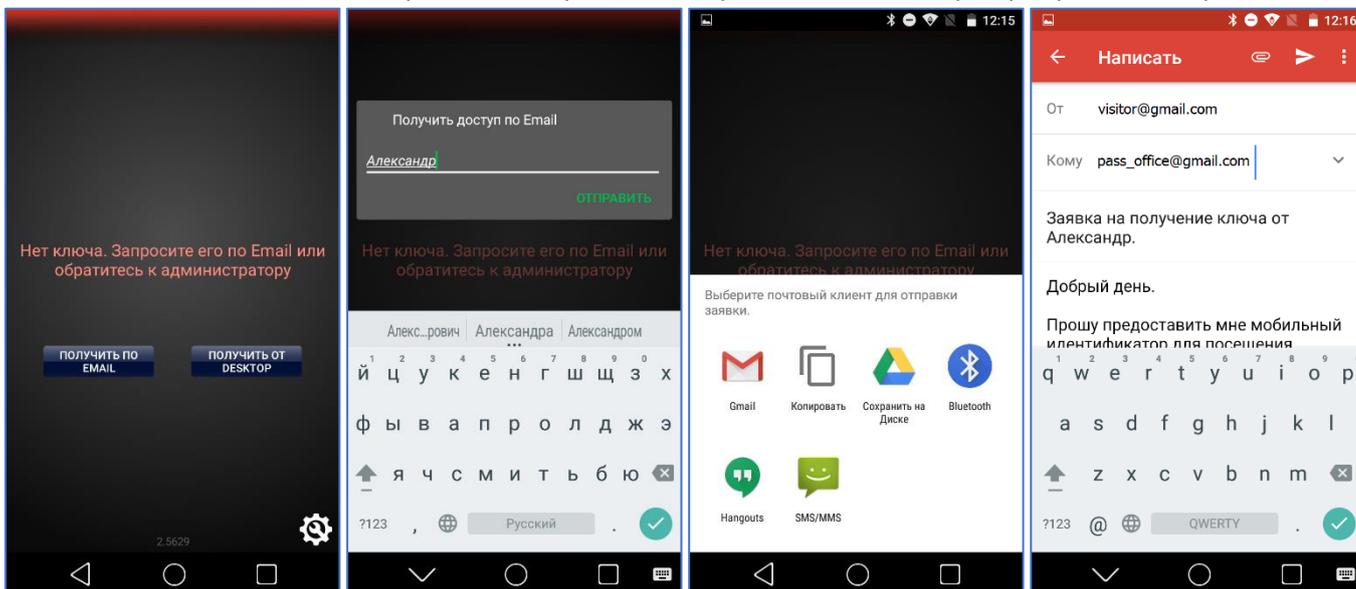
### Выдача идентификаторов по e-mail

С помощью сервиса JSON и PW Desktop возможна выдача мобильных идентификаторов по e-mail

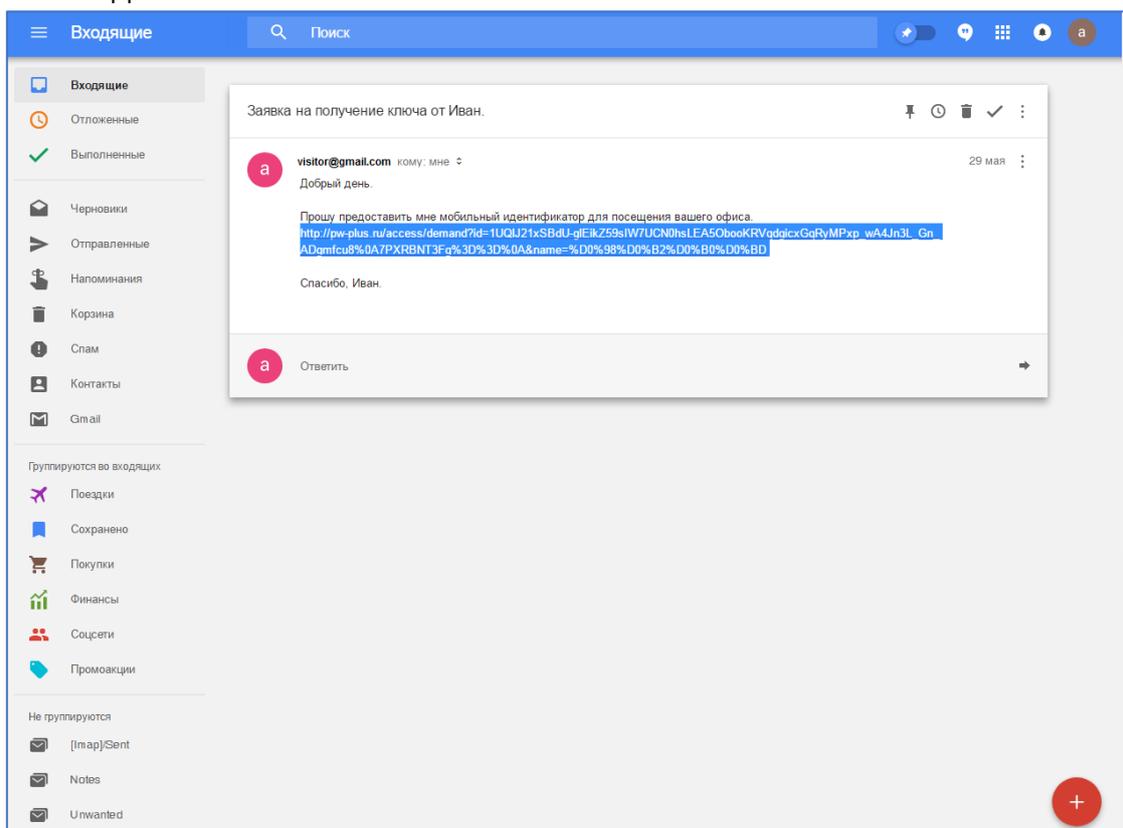
#### Процедура нового идентификатора по e-mail с помощью PW Desktop и PW Config

1. Если у посетителя еще нет приложения Proxway ID, он его устанавливает на своем смартфоне
2. В приложении Proxway ID пользователь нажимает кнопку "Получить по e-mail"(или в настройках приложения использует форму "Запрос сетевого ключа"), вводит свое имя и выбирает программу почты для отправки заявки.

Далее вводит адрес, используемый для приема заявок в бюро пропусков, и отправляет письмо



- Получив почту с заявкой, сотрудник бюро пропусков переводит настольный считыватель в режим выдачи идентификаторов – размещает на считывателе мастер-карту.
- Затем сотрудник бюро пропусков открывает письмо и копирует параметр "ID пользователя" в СКУД.



Для удобства операторов рекомендуем копировать полную ссылку и уже средствами ПО СКУД извлекать параметр "ID пользователя"

#### Выпуск мобильных идентификаторов для удаленных пользователей

Передача - POST запрос:

http://localhost:4001/issueRemoteIdentifier

Тело запроса:

```
{
  "UserCode": "VnztWvpazJFikIMKQb2KCKJEmWg_cN48UnRczPmjcyK6rLOA-sZLE_xp_wA4Jn3L_Gn_ADgmfctI E8eldlCONw=",
  "SendEmail": true,
  "Email": "visitor@gmail.com",
  "OutputAs": "Raw"
}
```

Url:	http://pw-plus.ru/access/de
Код пользователя:	VnztWvpazJFikIMKQb2KC
Email:	<input checked="" type="checkbox"/> visitor@gmail.com
Формат идентификатора:	<input checked="" type="checkbox"/> Raw
Отправить	

Ответ:

```
{
  "Identifier": "MobileID[5CF0CF54BE81EAFB] E237D9E381C792C2DC",
  "ActivationLink": "http://pw-plus.ru/access/accept?id=rempPN_tr-xluUprsiXQAw%3D%3D%0A"
}
```

- СКУД у JSON сервиса запрашивает лицензию для выдачи мобильного идентификатора
- JSON сервис получает мобильный идентификатор из выделенного списка PW Desktop, код полученного мобильного идентификатора выводится в COM порт. Сервис JSON сохраняет полученный код в памяти и далее передает в приложение СКУД вместе с сформированной ссылкой подтверждения.

### Запрос:

<http://localhost:40011/IssueRemoteIdentifier>

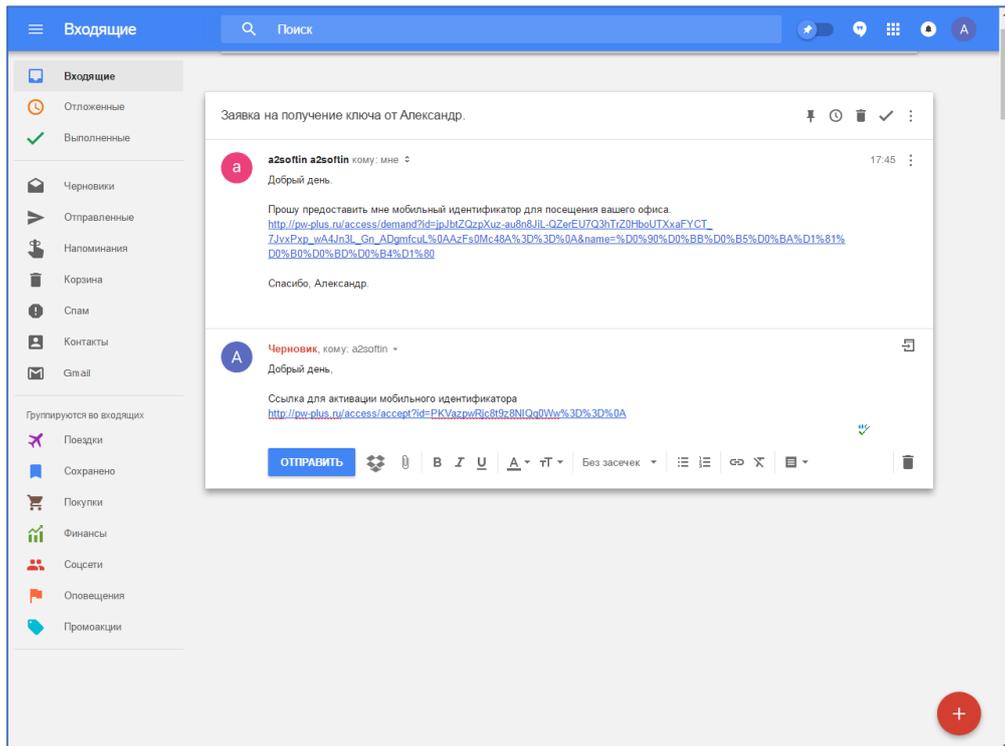
### Тело запроса:

```
{
  "UserCode": "rPMU130_rsi_Rjr5zjKt4nT-z0-l_fhxdR0plbGoImURrdGWWW6qEPxp_wA4Jn3L_Gn_ADgmfctw%0ALGkKpUmcLg%3D%3D%0A"
}
```

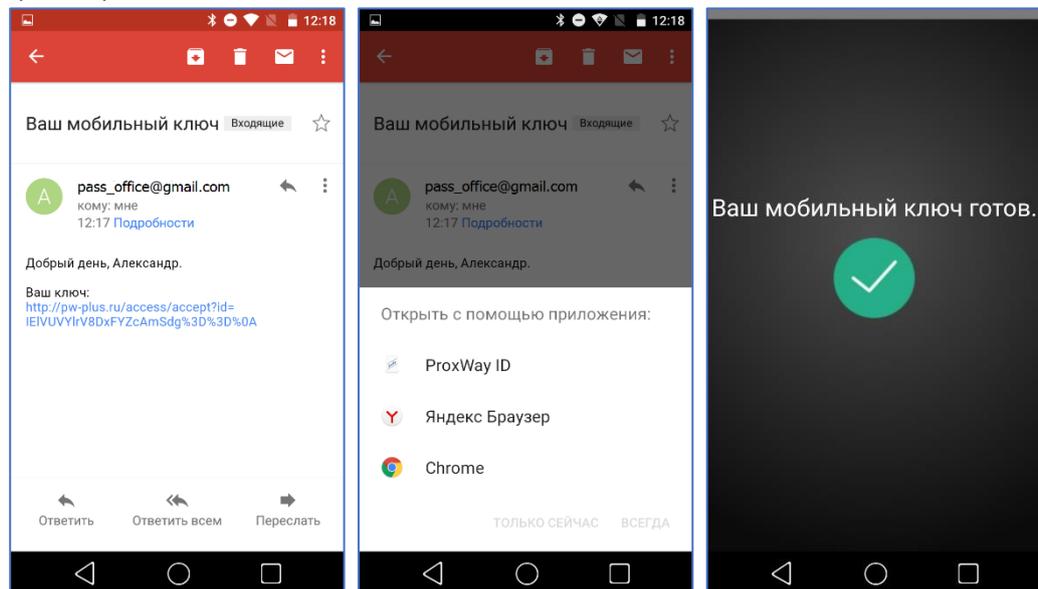
### Получение данных:

```
{
  "Identifier": "54BE81EAF4",
  "ActivationLink": "http://pw-plus.ru/access/accept?id=uPZnzueJCPuhpxHAKl8kIQ%3D%3D%0A"
}
```

7. Сотрудник бюро пропусков копирует полученную ссылку в e-mail и отправляет письмо на адрес посетителя.



8. Получив почту с заявкой, посетитель открывает её на смартфоне с помощью приложения Proxway ID. После нажатия на ссылку в письме будет запущено приложение и сохранен код идентификатора.



## Описание форматов вывода

Форматы вывода кода идентификатора те же, что для процедуры чтения кодов идентификаторов

### Запрос:

```
http://localhost:40011/IssueRemoteIdentifier
```

### Тело запроса:

```
{
  "UserCode": "rPMU130_rsi_Rjr5zjKt4nT-z0-l_fhxdR0plbGcImURrdGWW6qEPxp_wA4Jn3L_Gn_ADgmfctw%0ALGkKpUmcLg%3D%3D%0A",
  "OutputAs": "Raw"
}
```

### Получение данных:

```
{
  "Identifier": "NewMobileID[5CF0CF54BE81EAF4]",
  "ActivationLink": "http://pw-plus.ru/access/accept?id=uPZnzueJCPuhpxHAKl8kIQ%3D%3D%0A"
}
```

## Отправка по e-mail

Если настроена отправка электронной почты сервисом JSON, то при удаленной выдаче мобильных идентификаторов ссылка для их активации будет отправлена на указанный в запросе e-mail адрес.

### Запрос:

```
http://localhost:40011/IssueRemoteIdentifier
```

### Тело запроса:

```
{
  "UserCode": "VnztWVpazJFikIMKQb2KCKjEmWg_cN48UnRczPmjcyK5rLOA-sZLE_xp_wA4Jn3L_Gn_ADgmfctl_E8eldICONw== ",
  "SendEmail": true,
  "Email": "visitor@gmail.com",
  "OutputAs": Raw
}
```

### Получение данных:

```
{
  "Identifier": "MobileID[5CF0CF54BE81EAFB]_E237D9E381C792C2DC",
  "ActivationLink": "http://pw-plus.ru/access/accept?id=rembPN_rr-xluUprslXQAw%3D%3D%0A"
}
```

## Дополнительная защита систем доступа (персонализация идентификаторов)

Считыватели Proxway поддерживают режим, в котором они принимают код только тех мобильных идентификаторов, которые зашифрованы известными им ключом шифрования (паролем).

Чтобы создать закрытую систему достаточно выполнить программирование считывателей задав им секретный пароль, и Desktop будет выдавать уже закрытые уникальным ключом шифрования идентификаторы.

Персонализация поддерживается для идентификаторов Mobile ID и Mifare®. Также для Mifare® есть возможность хранить свой номер идентификатора в зашифрованных секторах данных.

Внимание!!! При работе с идентификатором Mifare® Classic шифруются все свободные сектора данных, чтобы исключить возможность взлома с помощью подбора. Если до этого на карточке были сектора с данными другой программы, то они пропускаются.

### Определение, включен ли режим шифрования идентификаторов

С помощью данной команды можно информацию об режиме шифрования, включенном в устройстве.

Таблица значений возможных режимов приведена ниже, по умолчанию – все значения "false".

#### Запрос:

```
http://localhost:40011/isEncrypted
```

#### Тело запроса:

```
{  
}
```

#### Ответ:

```
{  
  "EncryptedMifare": true,  
  "EncryptedMifarePlus": true,  
  "EncryptedMobile": false,  
  "MifareDiversificationEnabled": false,  
  "MifarePlusDiversificationEnabled": false,  
  "MifarePlusStartIdentifier": null,  
  "MifareStartIdentifier": "00AA00EE01"  
}
```

### Определение, включен ли режим шифрования мобильных идентификаторов

С помощью данной команды можно получить информацию включено ли шифрование для мобильных идентификаторов.

#### Запрос:

```
http://127.0.0.1:40011/GetBleEncryption
```

#### Тело запроса:

```
{  
}
```

#### Ответ:

```
{  
  "Encrypted": false  
}
```

### Установка режима шифрования мобильных идентификаторов

Включение режима шифрования и установка пароля шифрования

#### Запрос:

```
http://localhost:40011/SecureMobileIDs
```

#### Тело запроса:

```
{
  "MobilePassword": "123456AA"
}
```

**Ответ:**

```
{
  "EncryptedMobile": true
}
```

Внимание!!! Шифрованные и нешифрованные идентификаторы хранятся в различных областях памяти программы PW ID.

Выключение режима шифрования и очистка пароля

**Запрос:**

```
http://localhost:40011/SecureMobileIDs
```

**Тело запроса:**

```
{
  "MobilePassword": ""
}
```

**Ответ:**

```
{
  "EncryptedMobile": false
}
```

Выдача зашифрованных мобильных идентификаторов происходит таким же образом, как и незашифрованных. Возможен локальный или удаленный (по e-mail) выпуск Mobile ID.

Установка режима шифрования для карт Mifare®

Установка режима шифрования Mifare® Classic.

С помощью данной команды можно включить только шифрование карт Mifare® Classic.

**Запрос:**

```
http://localhost:40011/SecureMifareIDs
```

**Тело запроса:**

```
{
  "MifarePassword": "1122334455FF"
}
```

**Ответ:**

```
{
  "EncryptedMifare": true
}
```

Установка режима шифрования с персонализированным номером идентификатора Mifare® Classic

С помощью данной команды можно только включить шифрование для Mifare® Classic и задать стартовый номер для записи индивидуальных идентификаторов. После каждого выпуска карты номер будет увеличиваться на единицу.

**Запрос:**

```
http://localhost:40011/SecureMifareIDs
```

**Тело запроса:**

```
{
```

```
"MifarePassword": "1122334455FF",
"StartingCode": "1000EA1000"
}
```

**Ответ:**

```
{
  "EncryptedMifare": true
}
```

где 1122334455FF пароль - 6 байт (12 символов hex), а 1000EA1000 (5 байт, 10 символов hex) - стартовый номер для записи индивидуальных идентификаторов

Отключение режима шифрования Mifare® Classic

Будет отключен режим шифрования, очищен пароль и текущее значение индивидуального идентификатора

**Запрос:**

<http://localhost:40011/SecureMifareIDs>

**Тело запроса:**

```
{
  "MifarePassword": ""
}
```

**Ответ:**

```
{
  "EncryptedMifare": false
}
```

Установка режима шифрования Mifare® Plus.

С помощью данной команды можно включить только шифрование карт Mifare® Plus.

**Запрос:**

<http://localhost:40011/SecureMifarePlusIDs>

**Тело запроса:**

```
{
  "MifarePlusPassword":
  "11223344556677FF11223344556677FF11223344556677FF11223344556677FF",
}
```

**Ответ:**

```
{
  "EncryptedMifarePlus": true
}
```

Установка режима шифрования с персонифицированным номером идентификатора Mifare® Plus

С помощью данной команды можно только включить шифрование для Mifare® Plus и задать стартовый номер для записи индивидуальных идентификаторов. Стартовый номер совпадает с номером для Mifare® Classic. После каждого выпуска карты номер будет увеличиваться на единицу.

**Запрос:**

<http://localhost:40011/SecureMifarePlusIDs>

**Тело запроса:**

```
{
  "MifarePlusPassword":
  "11223344556677FF11223344556677FF11223344556677FF11223344556677FF",
  "StartingCode": "1000EA1000"
}
```

**Ответ:**

```
{
  "EncryptedMifarePlus": true
}
```

где 1122334455FF пароль - 6 байт (12 символов hex), а 1000EA1000 (5 байт, 10 символов hex) - стартовый номер для записи индивидуальных идентификаторов

Отключение режима шифрования Mifare Plus

Будет отключен режим шифрования, очищен пароль и текущее значение индивидуального идентификатора

**Запрос:**

```
http://localhost:40011/SecureMifarePlusIDs
```

**Тело запроса:**

```
{
  "MifarePlusPassword": ""
}
```

**Ответ:**

```
{
  "EncryptedMifarePlus": false
}
```

Выпуск зашифрованной карты Mifare® Classic

Будет выпущен новый идентификатор с параметрами, заданными в PW Config

**Запрос:**

```
http://localhost:40011/GetEncryptedMifareID
```

**Тело запроса (можно установить формат вывода):**

```
{
  "OutputAs": "Raw"
}
```

**Ответ:**

```
{
  "Mifare[1000EA1000] 234,04096 1K (0004,08) "
}
```

**Процедура выдачи нового идентификатора**

1. Администратор размещает на считывателе новую (чистую) карту Mifare®.
2. Администратор дает команду на выпуск новой карты – выполняется шифрование.
3. По завершению шифрования код полученного идентификатора считывается и выводится в СОМ порт (передается в СКУД и т.д.)

**Возможные ошибки при выпуске карты:**

Выпуск не удался, все 16 секторов недоступны для шифрования

```
{
  "code": 5,
  "message": "All mifare card blocks are inaccessible"
}
```

Карта была убрана со считывателя, несколько секторов не зашифрованы (потенциальная брешь для атаки, следует перевыпустить этот идентификатор)

```
{
  "code": 6,
  "message": "Mifare card blocks protection error"
}
```

## Выпуск зашифрованной карты Mifare® Plus

Будет выпущен новый идентификатор с параметрами, заданными в PW Config

### Запрос:

```
http://localhost:40011/GetEncryptedMifarePlusID
```

### Тело запроса (можно установить формат вывода):

```
{
  "OutputAs": "Raw"
}
```

### Ответ:

```
{
  "Mifare[10EA1000] 234,04096 DF (0004,08)"
}
```

### Процедура выдачи нового идентификатора

1. Администратор размещает на считывателе новую (чистую) карту Mifare® Plus.
2. Администратор дает команду на выпуск новой карты – выполняется шифрование.
3. По завершению шифрования код полученного идентификатора считывается и выводится в СОМ порт (передается в СКУД и т.д.)

### Возможные ошибки при выпуске карты:

Выпуск не удался, ни один из секторов недоступен для шифрования

```
{
  "code": 5,
  "message": "All mifare card blocks are inaccessible"
}
```

## Стирание зашифрованной карты

Выпущенный ранее идентификатор, с параметрами заданными с помощью PW Config, будет очищен

### Запрос:

```
http://localhost:40011/EraseMifareID
```

### Тело запроса (можно установить формат вывода):

```
{
  "OutputAs": "Raw"
}
```

### Ответ:

```
{
}
```

### Процедура выдачи нового идентификатора

1. Администратор размещает на считывателе карту Mifare®.
2. Администратор дает команду на стирание карты – выполняется стирание.

## Настройка отправки электронной почты сервисом JSON при выдаче мобильных идентификаторов по e-mail

Можно настроить работу сервиса JSON с почтовым SMTP сервером. Тогда при удаленной выдаче мобильных идентификаторов ссылка для их активации будет отправлена на указанный в запросе e-mail адрес.

Параметры почтового сервера сохраняются в реестре:

```
[HKEY_LOCAL_MACHINE\SOFTWARE\WOW6432Node\ELICS\ProxWayIP\2.00]
  "SmtpServerEnabled"
  "SMTPEncrypted"
  "SMTPServer"
  "SMTPPort"
  "SMTPUser"
  "SMTPPassword"
  "SMTPSendFrom"
```

### Запрос проверки параметров:

<http://localhost:40011/GetEmailServer>

### Тело запроса:

```
{
}
```

### Ответ:

```
{
  "SmtpServerEnabled": true,
  "SMTPEncrypted": false,
  "SMTPServer": "192.168.1.1",
  "SMTPPort": 25,
  "SMTPUser": "user",
  "SMTPPassword": "password",
  "SMTPSendFrom": "user@domail.com"
}
```

### Запрос установки параметров:

<http://localhost:40011/SetEmailServer>

### Тело запроса:

```
{
  "SmtpServerEnabled": true,
  "SMTPEncrypted": false,
  "SMTPServer": "192.168.1.1",
  "SMTPPort": 25,
  "SMTPUser": "user",
  "SMTPPassword": "password",
  "SMTPSendFrom": "user@domail.com"
}
```

### Ответ:

```
{
  "SmtpServerEnabled": true
}
```

## Нерабочие запросы

### Получение и установка выходного формата кода идентификатора (\*не реализовано)

Получение текущего формата и установка длины кода идентификатора (Wiegand26-Wiegand64) для его обрезания/дополнения.

#### Запрос:

```
http://localhost:40011/GetOutputInterface  
http://localhost:40011/SetOutputInterface
```

### Получение и установка индикации считывателя (\*не реализовано)

Получение текущих настроек индикации и их установка.

#### Запрос:

```
http://localhost:40011/GetIndication  
http://localhost:40011/SetIndication
```

### Получение токена авторизации при удаленном доступе (\*заморожено)

#### Запрос:

```
http://localhost:40011/Authorization
```

### Удаленная загрузка и обновление микропрограммы (\*не реализовано)

#### Запрос:

```
http://localhost:40011/UploadFirmware
```

### Получение и установка Facility кода для старых карт FSK (\*не реализовано)

Получение текущих настроек индикации и их установка.

#### Запрос:

```
http://localhost:40011/GetFacilityCode  
http://localhost:40011/SetFacilityCode
```

### Установка пароля шифрования для BLE (\*устарело, заморожено)

#### Запрос:

```
http://localhost:40011/SetBleEncryption
```

#### Тело запроса:

```
{  
  "Key": 12345600  
}
```

#### Ответ:

```
{  
}
```