Identity Blitz

Сервер аутентификации Blitz Identity Provider

Версия 2.9.0

РУКОВОДСТВО АДМИНИСТРАТОРА

Москва, 2017

СОДЕРЖАНИЕ

ВВЕДЕНИЕ

Сервер аутентификации Blitz Identity Provider эффективно решает задачу защиты пользовательских учетных записей — снимает эту головную боль с разработчиков прикладных приложений. Blitz Identity Provider предоставляет готовые, гибко настраиваемые под заказчика и реализованные с учетом лучших практик функции защиты учетных записей:

1) обеспечение единого сквозного входа пользователя в приложения (Single Sign On);
2) двухфакторная аутентификацию с использованием различных методов;
3) конфигурируемый пользовательский интерфейс страниц входа, регистрации, восстановления доступа, управления учетной записью;
4) вход с использованием сторонних поставщиков идентификации: вход с помощью аккаунтов социальных сетей (Social Login), идентификация с использованием государственной Единой системы идентификации и аутентификации (ЕСИА, госуслуги), федеративный вход пользователей с использованием установок Blitz Identity Provider у организаций-партнеров или в филиалах организации;
5) авторизация входа в приложения в соответствии с настроенными правилами доступа различных групп пользователей к различным приложениям с требуемым уровнем аутентификации;
6) протоколирование событий безопасности, связанных с использованием учётных записей пользователей;
7) защита RESTful веб-сервисов, обеспечение авторизации при вызове сервисов различными клиентами с делегированными пользователями разрешениями на осуществление доступа от их имени.

Blitz Identity Provider подходит как для решения задач обеспечения доступа пользователей Интернет к интернет-сервисам компании, так и для задач обеспечения контроля доступа сотрудников к ресурсам своей организации, будь то развернутые внутри организации приложения или арендованные облачные сервисы.

Blitz Identity Provider можно использовать в качестве интеграционной платформы для подключения различных приложений организации и ее филиалов к существующим LDAP и серверам Microsoft Active Directory. Если в организации используется домен, то Blitz Identity Provider обеспечит сквозной доступ пользователя к различным приложениям организации таким образом, что пользователь будет проходить аутентификацию лишь единожды, при входе в сетевой домен со своего ПК.

1. УСТАНОВКА ПРОДУКТА

В этой главе рассматриваются шаги, выполнение которых необходимо до момента первого запуска сервера аутентификации Blitz Identity Provider.

ПО Blitz Identity Provider имеет две редакции – Standard Edition и Enterprise Edition.

- Редакция Standard Edition проста в установке, задействует в работе только один сервер, хорошо подходит для использования небольшими организациями.
- Редакция Enterprise Edition предоставляет возможность развертывания на нескольких серверах в кластере, предоставляет расширенный набор функциональности, позволяющей гибко настроить систему под индивидуальные требования организации.

1.1. Подготовка к установке

1.1.1. Подготовка к установке редакции Standard

Для развертывания редакции Standard необходимо использовать отдельный сервер (допустимо использовать виртуальный сервер) со следующими характеристиками:

- Минимальные характеристики: 1 процессорное ядро, не менее 2 Гб ОЗУ и 15 Гб пространства на жестком диске.
- Рекомендуемые характеристики: 2 процессорных ядра, 4 Гб ОЗУ, 30 Гб пространства на жестком диске.

Для работы приложения требуется следующее ПО, установленное на сервере:

- ОС Windows (7/8/10/Server) или ОС Linux (CentOS версии 7 или выше / RHEL версии 7 или выше / Debian версии 8 или выше / Ubuntu версии 16.04 или выше);
- Менеджер памяти Memcached версии 1.4.15 (и выше)[1] – только для Linux, в случае использования Windows отдельно устанавливать Memcached не требуется.

В качестве источника данных пользователей можно воспользоваться встроенным хранилищем, либо использовать внешнее хранилище (например, Microsoft Active Directory).

1.1.2. Подготовка к установке редакции Enterprise

Для работы Enterprise-редакции требуется один или несколько серверов. Рекомендуемая схема развертывания приведена на рисунке 1.

[1] http://memcached.org/downloads

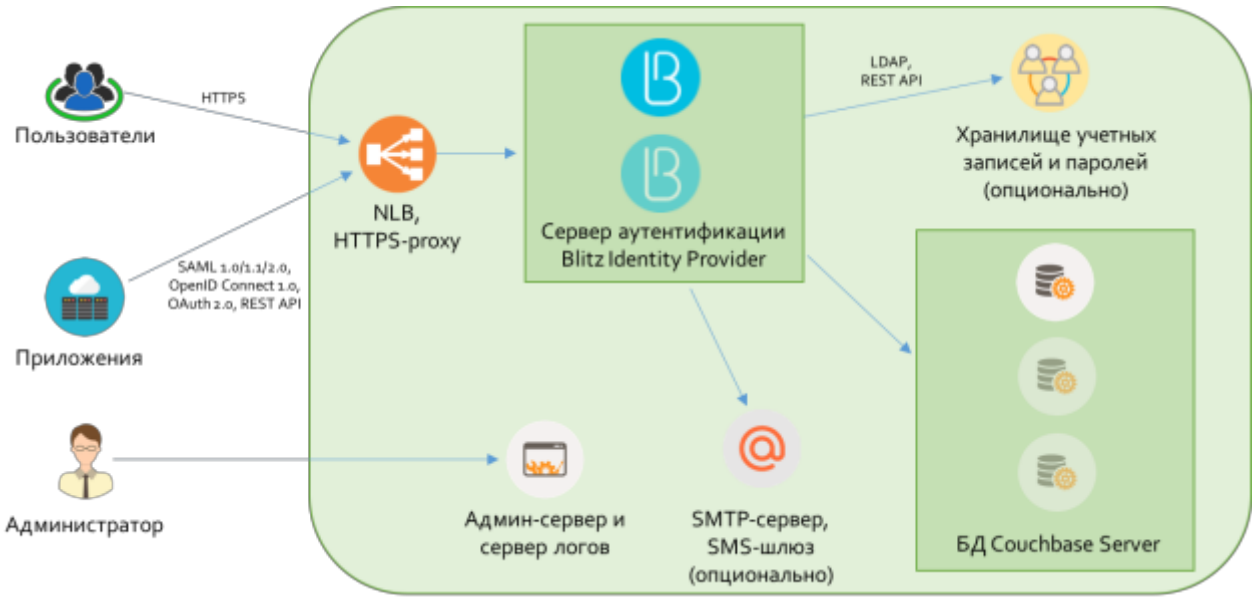

Рисунок 1 – Рекомендуемая схема развертывания Blitz Identity Provider Enterprise Edition

Требуемые конфигурации серверов для развертывания Blitz Identity Provider:

1. Балансировщик нагрузки (HTTPS-proxy) – можно использовать существующий веб-сервер для балансировки нагрузки и снятия SSL-шифрования с входящего трафика. Если принято решение использовать под Blitz Identity Provider отдельный балансировщик, то рекомендуется следующая конфигурация: 1 ядро, 2 Гб ОЗУ, 30 Гб пространства на жестком диске.
2. Сервера аутентификации Blitz Identity Provider – для обеспечения отказоустойчивости рекомендуется использовать не менее 2 серверов. Рекомендуется следующая конфигурация: 2 ядра, 8 Гб ОЗУ, 30 Гб пространства на жестком диске.
3. Сервера БД Couchbase Server – для обеспечения отказоустойчивости рекомендуется использовать 3 сервера[2]. Рекомендуется следующая конфигурация[3]: 2 ядра, 8 Гб ОЗУ, 50 Гб пространства на жестком диске.
4. Хранилища учетных записей и паролей – можно использовать одно или несколько хранилищ, таких как LDAP-сервера, Microsoft Active Directory, либо любую другую систему хранения учетных записей и паролей (в этом случае потребуется разработка REST-сервисов для интеграции Blitz Identity Provider с хранилищем учетных записей). Если планируется создание нового хранилища учетных записей, то в качестве него рекомендуется использовать LDAP-сервер, например, 389 Directory Server. Рекомендуемая конфигурация LDAP-сервера – 2 ядра, 8 Гб ОЗУ, 50 Гб пространства на жестком диске.

[2] https://developer.couchbase.com/documentation/server/4.6/install/deployment-considerations-lt-3nodes.html
[3] https://developer.couchbase.com/documentation/server/4.6/install/pre-install.html

5. Сервер администрирования и сервер логов – рекомендуется выделить отдельный сервер, на котором будет запущено веб-приложение «Консоль управления Blitz Identity Provider», и куда будет настроен сбор логов с различных серверов кластера Blitz Identity Provider. Рекомендуемая конфигурация админ-сервера – 1 ядро, 2 Гб ОЗУ, 30 Гб пространства на жестком диске.

На предназначенные для установки Blitz Identity Provider сервера требуется установить системное ПО:

- ОС Linux (CentOS версии 7 или выше / RHEL версии 7 или выше / Debian версии 8 или выше / Ubuntu версии 16.04 или выше);
- Java Oracle JDK 8u60 (или выше)[4];
- Менеджер памяти Memcached версии 1.4.15 (или выше);
- СУБД Couchbase Server.

1.2. Установка

1.2.1. Установка редакции Standard

Для установки под Windows необходимо запустить установщик и следовать инструкциям. В процессе установки, в частности, потребуется указать:

- логин и пароль доступа к консоли управления;
- домен, на котором будет работать Blitz Identity Provider.

Для установки под Linux перед установкой Blitz Identity Provider необходимо установить менеджер памяти Memcached версии 1.4.15 (или выше). Для этого выполнить на сервере команду (на примере CentOS / Red Hat Enterprise Linux):

```
yum install memcached
```

Затем для установки Blitz Identity Provider выполнить следующую команду (на примере установки под CentOS / Red Hat Enterprise Linux rpm-дистрибутива):

```
rpm -Uvh blitz-idp-Standard-X.X.X-X.noarch.rpm
```

Пример результатов успешной установки:

```
************************************************************
Your BlitzIdP configured on domain: bip-dapp02
You can change BlitzIdP domain using script: /usr/share/blitz-idp/scripts/change_domain.sh

Your BlitzIdP Console available at address - http://bip-dapp02:9000/blitz/console
Administration user credentials of BlitzIdP Console:
username - admin
password - 4f4b6AfCbb
Your can change user credentials at file - /usr/share/blitz-idp-Standard/conf/credentials
************************************************************
```

В данном сообщении содержится:

- ссылка на консоль управления;
- логин и пароль доступа к консоли;

[4] http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html

- ссылка на файл с учетными данными администратора;
- ссылка на скрипт, позволяющий изменить домен, на котором работает Blitz Identity Provider. При установке по умолчанию имя домена будет совпадать с именем сервера, на котором был запущен установщик.

1.2.2. Установка редакции Enterprise

Для установки необходимо выполнить следующую последовательность шагов (в качестве примера рассматривается установка под CentOS):

1. Установить и настроить Oracle JDK 1.8. на серверах, предназначенных для установки ПО сервера Blitz Identity Provider и административной консоли Blitz Identity Provider:

   1.1. Загрузить дистрибутив Oracle JDK 1.8. в виде архива tar по следующей ссылке:
   http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html

   1.2. Скопировать загруженный дистрибутив на сервера (например, в директорию /tmp).

   1.3. Создать директорию под установку Oracle JDK 1.8.

   ```
   mkdir -p /opt/oracle/jdk/
   ```

   1.4. Распаковать в созданную директорию дистрибутив Oracle JDK 1.8.

   ```
   tar xf /tmp/jdk-8u101-linux-x64.tar.gz -C /opt/oracle/jdk/
   ```

   1.5. Загрузить дистрибутив Java Cryptography Extension (JCE) Unlimited Strength Jurisdiction Policy Files 8 по следующей ссылке:
   http://www.oracle.com/technetwork/java/javase/downloads/jce8-download-2133166.html

   1.6. Скопировать загруженный дистрибутив на сервера (например, в директорию /tmp).

   1.7. Распаковать архив и скопировать содержимое в директорию с установленным Oracle JDK 1.8.

   ```
   cd /tmp
   unzip jce_policy-8.zip
   cp UnlimitedJCEPolicyJDK8/*.jar /opt/oracle/jdk/jdk1.8.0_101/jre/lib/security/
   ```

2. Установить менеджер памяти memcached. Сервис memcached должен быть установлен на серверах, предназначенных для установки ПО сервера Blitz Identity Provider и административной консоли Blitz Identity Provider. Версия memcached должна быть 1.4.15 или выше.

   2.1. Для установки memcached выполнить команду:

   ```
   yum -y install memcached
   ```

   2.2. После завершения установки добавить сервис memcached в автозапуск и запустить сервис:

   ```
   systemctl enable memcached
   systemctl start memcached
   ```

   2.3. Сервис memcached запускается на порту 11211. Нужно убедиться, что этот порт открыт на межсетевых экранах и может быть использован для соединения между серверами с Blitz Identity Provider.

3. Установка и настройка сервиса Couchbase Server.

3.1. Установить Couchbase Server на каждый из выделенных под установку СУБД серверов согласно инструкции:
https://developer.couchbase.com/documentation/server/4.6/install/install-linux.html.
Дистрибутив Couchbase Server можно загрузить с сайта производителя по ссылке:
https://www.couchbase.com/downloads.
Примечание: В DEV/TEST-средах допустимо Couchbase Server устанавливать на существующие сервера с Blitz Identity Provider, но в этом случае нужно учесть, что в Couchbase Server используется своя встроенная Memcached-служба, и во избежание конфликта необходимо скорректировать используемые Memcached порты в Blitz Identity Provider / Couchbase Server.

3.2. После завершения установки добавить сервис Couchbase Server в автозапуск и запустить сервис:

```
systemctl enable couchbase-server
systemctl start couchbase-server
```

3.3. Проверить работоспособность сервиса, выполнив команду:

```
systemctl status couchbase-server
```

3.4. Инициализировать на каждом сервере кластер Couchbase Server согласно инструкции (на первом сервере инициализируется кластер, остальные сервера включаются в кластер):
https://developer.couchbase.com/documentation/server/4.6/install/init-setup.html
Все настройки можно задать как предложено по умолчанию, только нужно для каждого сервера в hostname задать полное имя сервера. В качестве имени сервера не рекомендуется использовать его IP-адрес.

3.5. На любом одном из серверов кластера Couchbase Server выполнить скрипты по подготовке Couchbase Server к использованию Blitz Identity Provider:
Скрипты находятся в директории scripts в комплекте с дистрибутивом Blitz Identity Provider Enterprise Edition.
Скрипты нужно скопировать на любой сервер кластера Couchbase Server.
Далее перейти в директорию и выполнить скрипт создания buckets для хранения информации Blitz Identity Provider:

```
./cb_buckets_init.sh
```

В процессе выполнения скрипта понадобится ввести:

- Имя URL сервера Couchbase Server – ввести строку вида http://<hostname>:8091, где в качестве hostname указать имя хоста сервера, с которого выполняется скрипт.
- Логин учетной записи администратора Couchbase Server – задается при инициализации кластера при выполнении предыдущего пункта инструкции.

– Пароль учетной записи администратора Couchbase Server – задается при инициализации кластера при выполнении предыдущего пункта инструкции.

Выполнить скрипт создания индексов для выполнения поисковых запросов Blitz Identity Provider в БД:

```
./cb_indexes_init.sh
```

В процессе выполнения скрипта понадобится ввести через пробел список hostname всех серверов Couchbase Server из кластера.

4. Установить Blitz Identity Provider Console. Ее установку можно провести на любой сервер с ПО сервера Blitz Identity Provider, либо на выделенный административный сервер (рекомендуется). На сервере предварительно должны быть установлены Oracle JDK 1.8 и memcached (см. п.1 и п.2). Для установки Blitz Identity Provider Console необходимо:

4.1. На предназначенный для установки сервер скопировать файл blitz-console-2.9.0.bin (например, в директорию /tmp) из дистрибутива Blitz Identity Provider Enterprise Edition.

4.2. Запустить установщик:

```
cd /tmp
./blitz-console-2.9.0.bin
```

В ответ на запросы установщика задать:

– значение JAVA_HOME – задать директорию, в которую на сервере установлен Oracle JDK 1.8 (см. п.1, например, /opt/oracle/jdk/);

– внешнее имя домена, на котором будет функционировать Blitz Identity Provider;

– пароль к хранилищу ключей Blitz Identity Provider. Хранилище ключей будет сгенерировано в процессе установки, и доступ к хранилищу будет закрыт заданным паролем.

Дождаться окончания установки Blitz Identity Provider Console. Установка будет произведена в директорию /usr/share/identityblitz.

4.3. Отредактировать файл настроек /usr/share/identityblitz/blitz-config/blitz.conf:

Задать в блоке internal-store список развернутых ранее нод БД Couchbase Server.

```
"internal-store" : {
   "nodes" : [
      "[NODENAME-1]:8087",
      …
      "[NODENAME-n]:8087"
   ]
}
```

где [NODENAME-x] – имя каждого сервера БД Couchbase Server в виде FQDN имени сервера (например, nodename = node1.blitz.loc).

4.4. Настроить синхронизацию файлов конфигурации с сервера, на котором установлена Blitz Identity Provider Console, на сервера, на которых будет устанавливаться ПО сервера Blitz Identity Provider. Для этого на сервере с Blitz Identity Provider Console настроить incrontab со следующими параметрами:

```
/usr/share/identityblitz/blitz-config/ IN_MODIFY,IN_ATTRIB,IN_CREATE,IN_DELETE,IN_CLOSE_WRITE
/usr/share/identityblitz/scripts/config_sync.sh conf
/usr/share/identityblitz/blitz-config/flows IN_MODIFY,IN_ATTRIB,IN_CREATE,IN_DELETE,IN_CLOSE_WRITE
/usr/share/identityblitz/scripts/config_sync.sh flows
/usr/share/identityblitz/blitz-config/messages IN_MODIFY,IN_ATTRIB,IN_CREATE,IN_DELETE,IN_CLOSE_WRITE
/usr/share/identityblitz/scripts/config_sync.sh conf
/usr/share/identityblitz/blitz-config/saml IN_MODIFY,IN_ATTRIB,IN_CREATE,IN_DELETE,IN_CLOSE_WRITE
/usr/share/identityblitz/scripts/config_sync.sh conf
/usr/share/identityblitz/blitz-config/saml/conf IN_CREATE,IN_DELETE,IN_CLOSE_WRITE
/usr/share/identityblitz/scripts/config_sync.sh conf
/usr/share/identityblitz/blitz-config/saml/credentials IN_CREATE,IN_DELETE,IN_CLOSE_WRITE
/usr/share/identityblitz/scripts/config_sync.sh conf
/usr/share/identityblitz/blitz-config/saml/metadata IN_MODIFY,IN_ATTRIB,IN_CREATE,IN_DELETE,IN_CLOSE_WRITE
/usr/share/identityblitz/scripts/config_sync.sh conf
/usr/share/identityblitz/blitz-config/simple IN_MODIFY,IN_ATTRIB,IN_CREATE,IN_DELETE,IN_CLOSE_WRITE
/usr/share/identityblitz/scripts/config_sync.sh conf
/usr/share/identityblitz/blitz-config/assets IN_MODIFY,IN_ATTRIB,IN_CREATE,IN_DELETE,IN_CLOSE_WRITE
/usr/share/identityblitz/scripts/config_sync.sh assets
```

4.5. Запустить Blitz Identity Provider Console и добавить ее в автозапуск:

```
systemctl enable blitz-console
systemctl start blitz-console
```

5. Установить ПО Blitz Identity Provider на серверах, выделенных под его установку (рекомендуется выделить минимум 2 сервера). На серверах предварительно должны быть установлены Oracle JDK 1.8 и memcached (см. п.1 и п.2). Для установки Blitz Identity Provider необходимо:

5.1. На каждый предназначенный для установки сервер скопировать файлы blitz-idp-2.9.0.bin, blitz-registration-2.9.0.bin, blitz-recovery-2.9.0.bin (например, в директорию /tmp) из дистрибутива Blitz Identity Provider Enterprise Edition.

5.2. Запустить установщик сервиса аутентификации Blitz Identity Provider:

```
cd /tmp
./blitz-idp-2.9.0.bin
```

В ответ на запросы установщика задать:

- значение JAVA_HOME – задать директорию, в которую на сервере установлен Oracle JDK 1.8 (см. п.1, например, /opt/oracle/jdk/).

Дождаться окончания установки. Установка будет произведена в директорию /usr/share/identityblitz.

5.3. Запустить сервис аутентификации и добавить его в автозапуск:

```
systemctl enable blitz-idp
systemctl start blitz-idp
```

5.4. Запустить установщик сервиса регистрации пользователей Blitz Identity Provider (его установку можно пропустить, если не планируется использовать функцию самостоятельной регистрации пользователей):

```
cd /tmp
./blitz-registration-2.9.0.bin
```

В ответ на запросы установщика задать:

– значение JAVA_HOME – задать директорию, в которую на сервере установлен Oracle JDK 1.8 (см. п.1, например, /opt/oracle/jdk/).

Дождаться окончания установки. Установка будет произведена в директорию /usr/share/identityblitz.

5.5. Запустить сервис регистрации пользователей и добавить его в автозапуск:

```
systemctl enable blitz-reg
systemctl start blitz-reg
```

5.6. Запустить установщик сервиса восстановления паролей (его установку можно пропустить, если не планируется использовать функцию самостоятельного восстановления пользователями забытых паролей):

```
cd /tmp
./blitz-recovery-2.9.0.bin
```

В ответ на запросы установщика задать:

– значение JAVA_HOME – задать директорию, в которую на сервере установлен Oracle JDK 1.8 (см. п.1, например, /opt/oracle/jdk/).

Дождаться окончания установки. Установка будет произведена в директорию /usr/share/identityblitz.

5.7. Запустить сервис восстановления паролей и добавить его в автозапуск:

```
systemctl enable blitz-rec
systemctl start blitz-rec
```

6. Настроить на входном балансировщике терминирование SSL и балансировку запросов. Ниже приведен пример настройки для nginx:

```
upstream blitz-idp {
    server [BLITZ-IDP-NODE-01]:9000;
    server [BLITZ-IDP-NODE-02]:9000;
}
upstream blitz-reg {
    server [BLITZ-IDP-NODE-01]:9002;
    server [BLITZ-IDP-NODE-02]:9002;
}
upstream blitz-rec {
    server [BLITZ-IDP-NODE-01]:9003;
    server [BLITZ-IDP-NODE-02]:9003;
}
upstream blitz-console {
    server [BLITZ-IDP-CONSOLE-NODE-01]:9001;
}
server {
    server_name     demo.blitz-idp.ru;
    listen          443;
    ssl             on;
    ssl_certificate         /etc/nginx/ssl/server.key;
    ssl_certificate_key     /etc/nginx/ssl/server.crt;
    ssl_protocols               TLSv1.2 TLSv1.1 TLSv1;
    ssl_session_cache           shared:SSL:20m;
    ssl_session_timeout         10m;
    ssl_ciphers
'EECDH+ECDSA+AESGCM:AES128+EECDH:AES128+EDH:!RC4:!aNULL:!eNULL:!LOW:!3DES:!MD5:!EXP:!PSK:!SRP:!D
SS:!CAMELLIA:!ADH';
    ssl_prefer_server_ciphers       on;
    ssl_dhparam                 /etc/nginx/ssl/dhparam.pem;
    proxy_next_upstream error timeout invalid_header http_500 http_502 http_503 http_504;
    proxy_set_header        Accept-Encoding   "";
    proxy_set_header        Host            $host;
    proxy_set_header        X-Real-IP       $remote_addr;
    proxy_set_header        X-Forwarded-For $proxy_add_x_forwarded_for;
    proxy_set_header        X-Forwarded-Proto $scheme;
    add_header              Front-End-Https   on;
```

```
    location /blitz/console {
      proxy_pass http://blitz-console/blitz/console;
    }
    location /blitz/reg {
      proxy_pass http://blitz-reg/blitz/reg;
    }
    location /blitz/recovery {
      proxy_pass http://blitz-rec/blitz/recovery;
    }
    location /blitz {
      proxy_pass http://blitz-idp/blitz;
    }
}
```

где:

- [BLITZ-IDP-NODE-01], [BLITZ-IDP-NODE-02] – имена (hostname) серверов с ПО Blitz Identity Provider (сервисы аутентификации, регистрации пользователей, восстановления паролей);
- [BLITZ-IDP-CONSOLE-NODE-01] – имя (hostname) сервера с Blitz Identity Provider Console;
- server_name – доменное имя, по которому Blitz Identity Provider доступен для пользователей по https (например, demo.blitz-idp.ru).

7. Установить и настроить LDAP-сервер, если не планируется использовать существующие в организации хранилища учетных записей, а также если планируется создать новое хранилище на основе LDAP-сервера. В DEV/TEST-средах можно обойтись без создания LDAP-сервера и использовать в качестве хранилища пользователей внутреннее хранилище Blitz Identity Provider (хранение учетных записей в БД Couchbase Server).

2. НАСТРОЙКА

2.1. Общие настройки

После установки Blitz Identity Provider основная настройка системы осуществляется в консоли управления, которая доступна по ссылке, обозначенной в результатах установки продукта (см. раздел 1.2 документа)[5].

Обычно ссылка имеет вид:

http(s)://<hostname>:9000/blitz/console

2.2. Подключение источника учетных записей

В качестве источника данных пользователей Blitz Identity Provider позволяет использовать:

- внешнее хранилище. В качестве такового может выступать:
  - LDAP-хранилище – это может быть любой сервер, поддерживающий протокол LDAP (389 Directory Server, Oracle Directory Server, OpenLDAP и другие), а также Microsoft Active Directory или Samba4;
  - иное хранилище, для обращения к которому Blitz Identity Provider необходимо разработать специальные REST-сервисы (см. раздел 2.2.2 документа). Данная возможность доступна только для Enterprise-редакции Blitz Identity Provider.
- внутреннее хранилище Blitz Identity Provider. При использовании такого хранилища все идентификационные данные пользователей хранятся в БД Blitz Identity Provider[6].

Для корректной работы Blitz Identity Provider требуется настройка хотя бы одного хранилища и конфигурирование атрибутов (см. п. 2.2.4). По умолчанию настроено внутреннее хранилище и добавлен ряд атрибутов.

Следует учесть, что данные отдельного пользователя могут храниться только в одном хранилище. Система допускает конфигурирование и подключение нескольких хранилищ, однако рекомендуется использовать одно основное хранилище для работы. Решение об использовании второго хранилища должно быть принято с учетом применяемой модели данных. Например, в подключенном LDAP-хранилище можно хранить данные сотрудников организации, а во внутреннем хранилище – данные специально зарегистрированных «внешних» пользователей (сотрудники партнерских организаций, фрилансеры и пр.).

[5] В случае установки под Windows создается специальный ярлык для запуска консоли управления.
[6] При использовании Standard-редакции это будет встроенная БД на основе MapDB, а при использовании Enterprise-редакции – установленная отдельно БД на основе Couchbase Server.

Выбор и настройка используемого хранилища осуществляется после первичной настройки Blitz Identity Provider в разделе *Хранилище*. По умолчанию настроено внутреннее хранилище. Для добавления внешнего хранилища следует нажать на кнопку «Добавить новое хранилище», после чего указать тип внешнего хранилища и настроить параметры взаимодействия с ним.

Допустимо удалить внутреннее хранилище, если его не планируется использовать. Для этого необходимо перейти в свойства соответствующего внешнего хранилища и нажать на кнопку «Удалить».

Использование нескольких хранилищ может решить задачу входа пользователей, хранящихся в разных LDAP-каталогах или в разных ветках одного каталога. Например, в результате объединения двух компаний можно подключить два каталога к Blitz Identity Provider и обеспечить вход пользователей, не прибегая к настройкам доверия, построению метакаталога и пр.

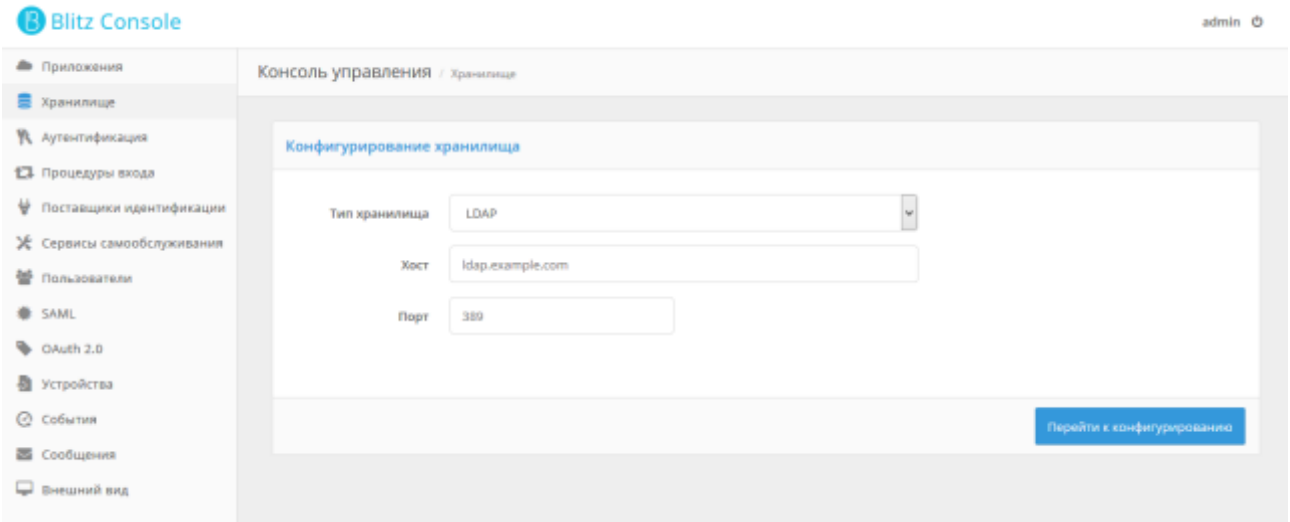

Рисунок 2 – Экран добавления хранилища учетных записей

2.2.1. Подключение внешнего LDAP-хранилища

Если в качестве источника идентификационных данных используется LDAP-хранилище, развернутое в организации, для его настройки необходимо воспользоваться разделом *Хранилище* консоли управления и выполнить следующие шаги:

- добавить новое хранилище, указать следующие данные:
  - тип добавляемого хранилища – выбрать LDAP;
  - адрес хранилища;
  - порт;
- сконфигурировать соединение с LDAP-хранилищем, прописав параметры соединения:
  - необходимость использования SSL-соединения;

- настройки пула соединений;

- указать логин и пароль пользователя, от имени которого будет осуществляться работа с LDAP-хранилищем (у этого пользователя должны быть права на чтение и на запись данных[7]), а также раздел каталога с учетными записями пользователей;
- указать настройки поиска – глубину поиска и максимальное число возвращаемых учетных записей (это влияет на число пользователей, отображаемых в разделе *Пользователи* консоли управления);
- указать параметры создания новых пользователей – DN родительского контейнера, внутри которого будут создаваться пользователи, и системные атрибуты, связанные со спецификой хранилища[8]. Данные параметры необходимы в том случае, если предполагается использовать Blitz Identity Provider для создания новых пользователей.

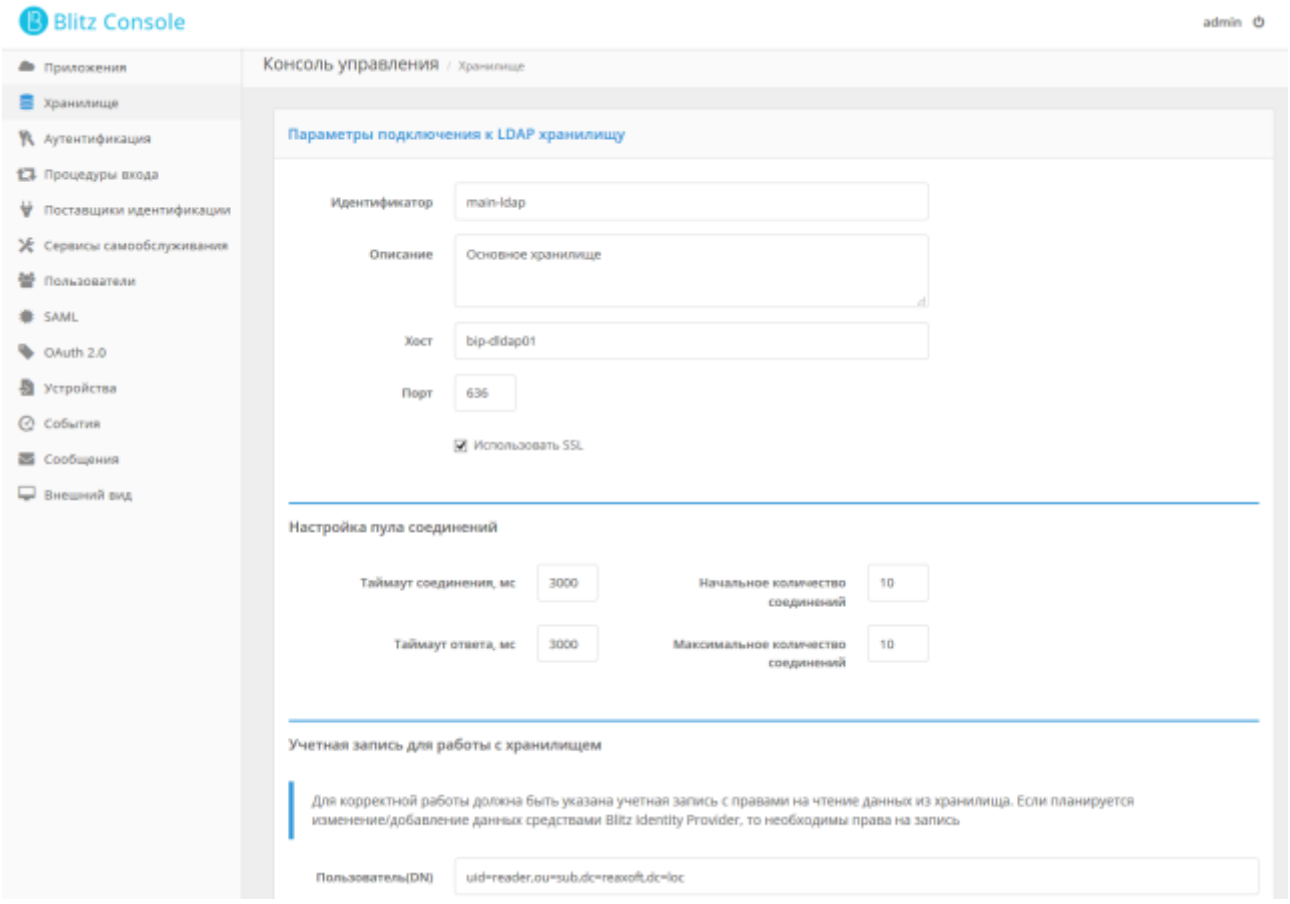

Рисунок 3 – Настройка подключения к LDAP-хранилищу данных (фрагмент)

[7] Допустимо указать пользователя только с правами на чтение, однако в этом случае не будут доступны некоторые функции, в частности, изменение данных пользователя из консоли управления и Личного кабинета, регистрация пользователя администратором, через сервисы самообслуживания и API.
[8] Например, objectclass, определяющий тип создаваемой учетной записи в LDAP. Для Microsoft Active Directory objectclass должен иметь формат Array of string и значение - top,person.

2.2.2. Подключение к хранилищу через REST-сервисы

Если в качестве источника идентификационных данных используется внешняя база данных (не LDAP-хранилище), то для подключения к ней требуется создание коннектора (не входит в поставку Blitz Identity Provider), который обеспечивает, с одной стороны, чтение (или изменение) необходимых данных из базы данных, а с другой – предоставляет данные в корректном формате в виде REST-сервисов для Blitz Identity Provider.

Для настройки взаимодействия с REST-сервисами необходимо выполнить следующие шаги:

- добавить новое хранилище, указав тип добавляемого хранилища - REST;
- указать URL следующих сервисов:
  - сервис поиска пользователей;
  - сервис проверки логина и пароля;
  - сервис смены пароля пользователем;
  - сервис добавления нового пользователя;
  - сервис изменения данных пользователя;
  - сервис удаления пользователя.

Скриншот страницы с настройками подключения к хранилищу с использованием REST-сервисов представлен на рис. 4.

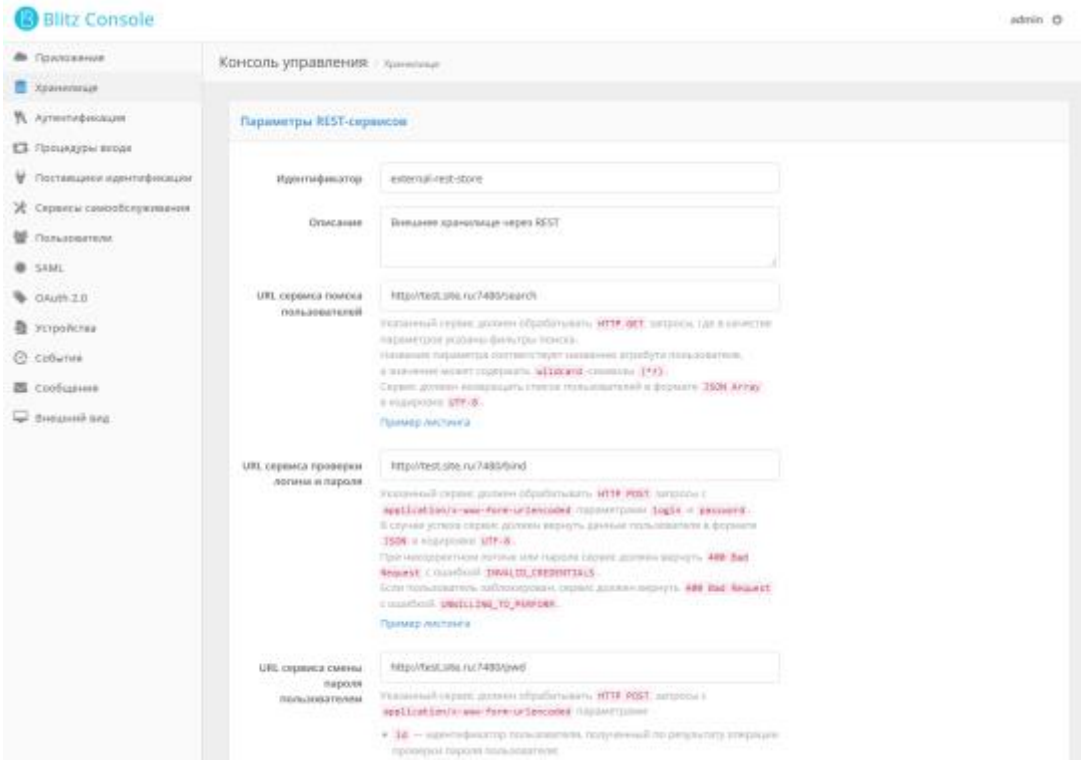

Рисунок 4 – Настройка подключения к хранилищу с использованием REST (фрагмент)

В следующих подразделах описаны требования к разработке REST-сервисов, предоставляющих необходимый Blitz Identity Provider доступ к хранилищу учетных записей.

После ввода URL-сервисов следует указать максимальное количество возвращаемых записей.

2.2.2.1. Сервис поиска пользователей

Сервис поиска пользователей должен обрабатывать запросы методом GET, где в качестве параметров указаны фильтры поиска. Название query-параметра должно соответствовать названию атрибута пользователя (данные атрибуты отмечаются как поисковые в консоли управления – см. п. 2.2.4 документа), а значение может содержать символы * и ? для поиска. Должно быть возможно указание нескольких атрибутов через амперсанд (&), в этом случае поиск должен осуществляться по всем атрибутам посредством логического ИЛИ.

Сервис должен возвращать список пользователей и их данные в формате json в кодировке UTF-8. Перечень возвращаемых данных (атрибут claims) определяется разработчиком сервиса – требуется возвращать те данные, которые далее предполагается использовать в системе.

Пример запроса:

```
GET /users/search?login=ivanov* HTTP/1.1
Host: idstore.identityblitz.com
Content-Type: application/json
Cache-Control: no-cache
```

Пример ответа:

```
HTTP/1.1 200 OK
Date: Mon, 18 Jul 2016 12:28:53 GMT
Content-Type: application/json; charset=utf-8

[
  {
    "id": "ivanov",
    "subjectId": "ivanov",
    "notes": [],
    "claims": [
      "name": "Ivan",
      "email": "ivanov@test.org",
      "organization": "TestOrg"
    ]
  },
  {
    "id": "ivanova",
    "subjectId": "ivanova",
    "notes": [],
    "claims": [
      "name": "Elena",
      "email": "ivanova@test.org",
      "organization": "TestOrg"
    ]
  }
]
```

2.2.2.2. Сервис проверки логина и пароля

Сервис проверки логина и пароля должен обрабатывать запросы методом POST, в теле которых указаны следующие параметры (в формате application/x-www-form-urlencoded):

- login – логин пользователя;
- password – пароль.

В случае успеха сервис должен вернуть данные пользователя в формате json в кодировке UTF-8.

При некорректном логине или пароле сервис должен вернуть 400 Bad Request с ошибкой INVALID_CREDENTIALS.

Если пользователь заблокирован, сервис должен вернуть 400 Bad Request с ошибкой UNWILLING_TO_PERFORM.

Пример запроса:

```
POST /users/bind HTTP/1.1
Host: idstore.identityblitz.com
Content-Type: application/x-www-form-urlencoded
Cache-Control: no-cache

login=ivanov&password=12345678
```

Пример ответа:

```
HTTP/1.1 200 OK
Date: Mon, 18 Jul 2016 12:38:53 GMT
Content-Type: application/json; charset=utf-8

{
   "id": "ivanov",
   "subjectId": "ivanov",
   "notes":["PASSWORD_EXPIRED"],
   "claims": [
      "name": "Ivan",
      "email": "ivanov@test.org",
      "organization": "TestOrg"
   ]
}
```

2.2.2.3. Сервис смены пароля пользователем

Сервис смены пароля пользователем должен обрабатывать запросы методом POST, в теле которых указаны следующие параметры (в формате application/x-www-form-urlencoded):

- id – идентификатор пользователя, полученный по результату операции проверки пароля пользователя;
- old_password – старый пароль;
- new_password – новый пароль.

В случае успеха сервис должен вернуть данные пользователя в формате json в кодировке UTF-8.

Если новый пароль не удовлетворяет политикам безопасности, сервис должен вернуть 400 Bad Request с ошибкой CONSTRAINT_VIOLATION.

Остальные возвращаемые ошибки должны быть аналогичны операции по проверке логина и пароля.

Пример запроса:

```
POST /users/changePassword HTTP/1.1
Host: idstore.identityblitz.com
Content-Type: application/x-www-form-urlencoded
Cache-Control: no-cache

id=ivanov&old_password=12345678&new_password=0987654321
```

Пример ответа:

```
HTTP/1.1 400 Bad Request
Date: Mon, 18 Jul 2016 12:43:23 GMT
Content-Type: text/plain; charset=utf-8

CONSTRAINT_VIOLATION
```

2.2.2.4. Сервис добавления нового пользователя

Сервис добавления нового пользователя должен обрабатывать запросы методом PUT, в теле которых указаны следующие параметры (в формате application/json):

- password – пароль пользователя (опционально);
- claims – атрибуты пользователя.

В случае успеха сервис должен вернуть данные пользователя в формате JSON в кодировке UTF-8.

Если пароль не удовлетворяет политикам безопасности, сервис должен вернуть 400 Bad Request с ошибкой CONSTRAINT_VIOLATION.

Если такой пользователь уже существует, сервис должен вернуть 400 Bad Request с ошибкой USER_ALREADY_EXISTS.

Пример запроса:

```
PUT /users HTTP/1.1
Host: idstore.identityblitz.com
Content-Type: application/json
Cache-Control: no-cache

{
   "password":"********",
   "claims": [
         "name": "Ivan",
         "email": "ivanov@test.org",
         "organization": "TestOrg"
   ]
}
```

Пример ответа:

```
HTTP/1.1 200 OK
Date: Mon, 18 Jul 2016 12:28:53 GMT
Content-Type: application/json; charset=utf-8
{
   "id": "ivanov",
   "subjectId": "ivanov",
   "claims": [
      "name": "Ivan",
      "email": "ivanov@test.org",
      "organization": "TestOrg"
   ]
}
```

2.2.2.5. Сервис изменения данных пользователя

Сервис изменения данных пользователя должен в качестве одного из query-параметров принимать идентификатор пользователя. При указании URL этого сервиса в консоли необходимо использовать строку подстановки для идентификатора пользователя – ${id}.

Сервис должен обрабатывать запросы методом POST, в теле которых указаны следующие параметры (в формате application/json):

- password – новое значение пароля пользователя (если пароль не передан, то он не должен измениться);
- replaced – новые значения атрибутов пользователя, которые нужно заменить или добавить;
- deleted – список названий удаляемых атрибутов.

В случае успеха сервис должен вернуть данные пользователя в формате JSON в кодировке UTF-8.

Если новый пароль не удовлетворяет политикам безопасности, сервис должен вернуть 400 Bad Request с ошибкой CONSTRAINT_VIOLATION.

Если такой пользователь не существует, сервис должен вернуть 400 Bad Request с ошибкой USER_NOT_FOUND.

Пример запроса:

```
POST /users/ivanov HTTP/1.1
Host: idstore.identityblitz.com
Content-Type: application/json
Cache-Control: no-cache

{
   "replaced": {
      "email": "ivanov@domain.org"
   },
   "deleted": ["organization"],
   "password": "########"
}
```

Пример ответа:

```
HTTP/1.1 200 OK
Date: Mon, 18 Jul 2016 12:38:53 GMT
Content-Type: application/json; charset=utf-8

{
   "id": "ivanov",
   "subjectId": "ivanov",
   "claims": [
      "name": "Ivan",
      "email": "ivanov@domain.org"
   ]
}
```

2.2.2.6. Сервис удаления пользователя

Сервис удаления учетной записи пользователя должен в качестве одного из query-параметров принимать идентификатор пользователя. При указании URL этого сервиса в консоли необходимо использовать строку подстановки для идентификатора пользователя – ${id}.

Сервис должен обрабатывать запросы методом DELETE.

В случае успеха сервис должен вернуть данные пользователя в формате JSON в кодировке UTF-8.

Если пользователь не существует, сервис должен вернуть 400 Bad Request с ошибкой USER_NOT_FOUND.

Пример запроса:

```
DELETE /users/ivanov HTTP/1.1
Host: idstore.identityblitz.com
Content-Type: application/json
Cache-Control: no-cache
```

Пример ответа:

```
HTTP/1.1 200 OK
Date: Mon, 18 Jul 2016 12:28:53 GMT
Content-Type: application/json; charset=utf-8

{
   "id": "ivanov",
   "subjectId": "ivanov",
   "claims": [
      "name": "Ivan",
      "email": "ivanov@domain.org"
   ]
}
```

2.2.3. Настройка внутреннего хранилища

Если в качестве источника идентификационных данных предполагается использовать внутреннюю базу данных, то для ее конфигурирования необходимо выполнить следующие шаги:

- добавить новое хранилище, указав тип добавляемого хранилища – BUILT-IN;
- указать идентификатор хранилища;
- дать описание хранилища;
- указать максимальное число возвращаемых учетных записей при поиске.

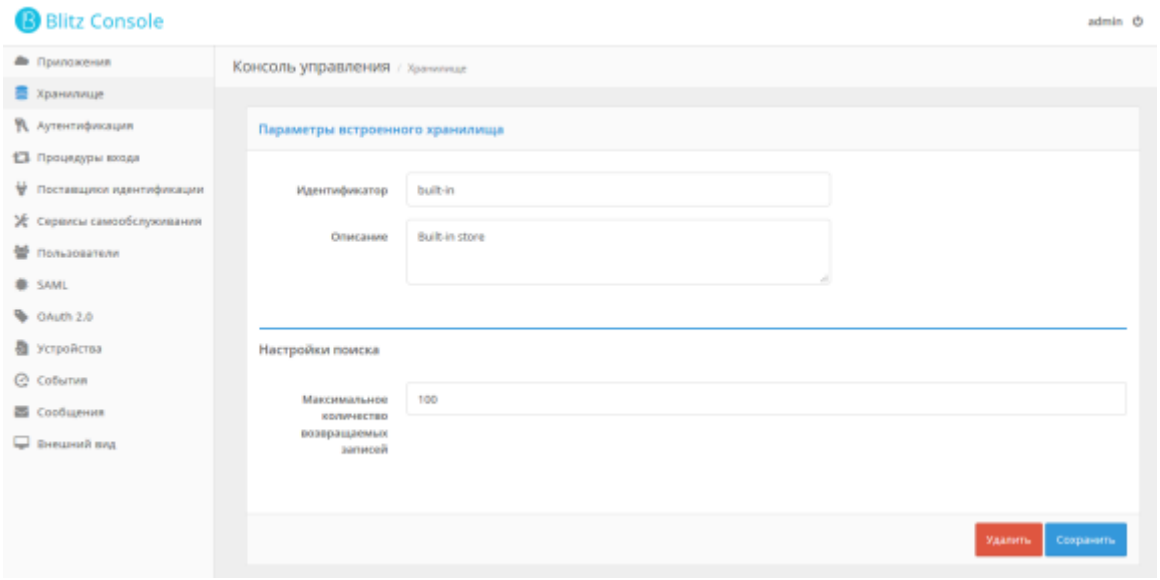

Рисунок 5 – Настройка внутреннего хранилища

2.2.4. Конфигурирование доступных атрибутов

Для корректной работы Blitz Identity Provider требуется указать, какие атрибуты будут использоваться в системе. Атрибуты используются в следующих целях:

- для проведения идентификации и аутентификации пользователей;
- для передачи подключенным к Blitz Identity Provider приложениям.

В качестве источника атрибутов пользователя могут выступать:

- основные хранилища, описанные в п. 2.2.1–2.2.3;
- хранилище дополнительных атрибутов – раздел внутренней базы данных Blitz Identity Provider. Рекомендуется использовать дополнительные атрибуты, если необходимо вести какой-то специфичный атрибут пользователя, и нет желания расширять схему используемого в качестве основного хранилища LDAP-каталога;
- вычисляемые атрибуты – в простейшем случае вычисляемому атрибуту может быть присвоено константное значение. Также можно сформировать значение вычисляемого атрибута на основе других атрибутов (например, получить домен пользователя из адреса электронной почты или «склеить» фамилию и имя пользователя).

На рис. 6 схематично представлен пример настроенных хранилищ и сконфигурированных атрибутов:

- В организации имеется LDAP-хранилище с данными сотрудников: идентификатор (uid), электронная почта (mail), мобильный телефон (mobile), адрес (address), имя (name), фамилия (surname).
- Организация предоставляет доступ к своим приложениям фрилансерам. Их учетные записи решено хранить не в основном LDAP-хранилище, а во внутреннем хранилище

Blitz Identity Provider. Набор атрибутов для фрилансеров используется такой же, как и у сотрудников: идентификатор (uid), электронная почта (mail), мобильный телефон (mobile), адрес (address), имя (name), фамилия (surname).

- Некоторые подключенные к Blitz Identity Provider приложения требовали передаче им особых атрибутов о сотрудниках/фрилансерах. Организация решила не расширять схему LDAP для хранения новых атрибутов, а использовать для этих целей хранилище дополнительных атрибутов Blitz Identity Provider. Пример таких атрибутов: роль пользователя в приложении (app_role), псевдоним (nickname), личный адрес электронной почты (personal_mail).

Внутреннее хранилище основных атрибутов всегда имеет ту же структуру, что и подключенное внешнее хранилище.

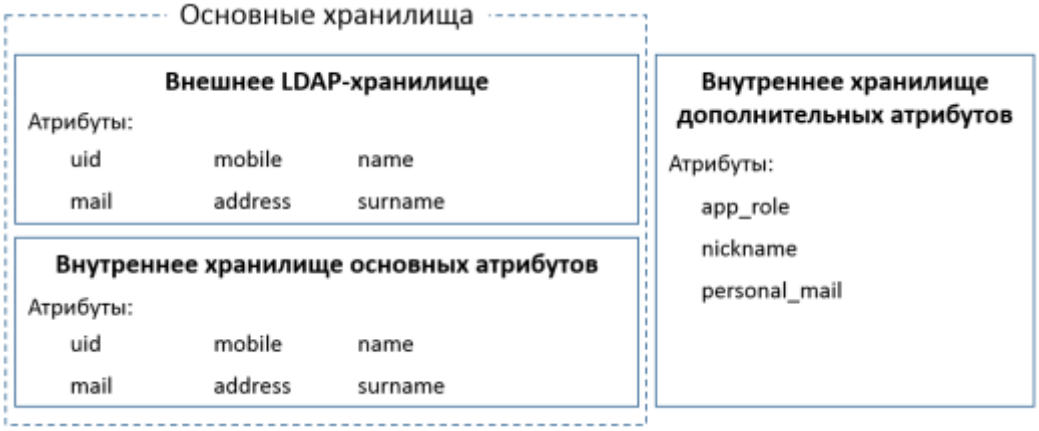

Рисунок 6 – Пример настроенных хранилищ и сконфигурированных атрибутов

Для конфигурирования перечня атрибутов, которые будут доступны Blitz Identity Provider, необходимо в разделе *Хранилище* выполнить следующие шаги:

- добавить новый атрибут, нажав на ссылку «+Добавить атрибут»;
- указать название атрибута, которое будет использоваться в Blitz Identity Provider;
- указать тип значения данных – формат данных;
- указать источник значения атрибута – это может быть основное хранилище, хранилище дополнительных атрибутов или выражение;
- указать правило формирования атрибутов:
  - при получении значения атрибута из хранилища основных/дополнительных атрибутов следует указать название этого атрибута в хранилище;
  - при получении значения атрибута с помощью некоторого правила следует прописать соответствующее правилу выражение. Если значение должно быть вычислено на основе ранее определенного атрибута, то для этого следует использовать строку подстановки вида ${attribute_name}. Доступно

использование параметров строк подстановки[9]. Например, для создания атрибута «домен» из электронной почты пользователя (mail) можно использовать выражение: ${mail##*@}.

- указать при необходимости регулярные выражения (правила чтения атрибута из хранилища) и строки подстановки (правила вывода атрибута). Например, чтобы очистить номер телефона (атрибут telephoneNumber в LDAP-хранилище) от скобок и пробелов можно использовать регулярное выражение ^(\+?)([78]?) ?\(?([0-9]{3})\)? ?([0-9]{7,11})$ и строку подстановки ${1-}${2-}${3-}${4-}.
- определить параметры атрибута:
  - является ли он базовым идентификатором (колонка «Ид.);
  - возможно ли производить по нему поиск (колонка «Поиск»)[10];
  - может ли атрибут использоваться в качестве утверждения (колонка «Утв.), т.е. можно ли будет данный атрибут передавать подключенным к Blitz Identity Provider приложениям по протоколам SAML и OAuth 2.0.

Не рекомендуется в будущем менять базовый идентификатор, т.к. все специфические пользовательские настройки привязываются именно к нему. Например, при изменении базового идентификатора могут быть потеряны связи между пользователями и их средствами усиленной аутентификации.

[9] См.: http://tldp.org/LDP/abs/html/parameter-substitution.html

[10] Если это атрибут из внешнего хранилища, то в целях производительности рекомендуется создать по нему поисковый индекс. Для внутреннего хранилища индекс будет создан автоматически при установке пометки «Поиск».

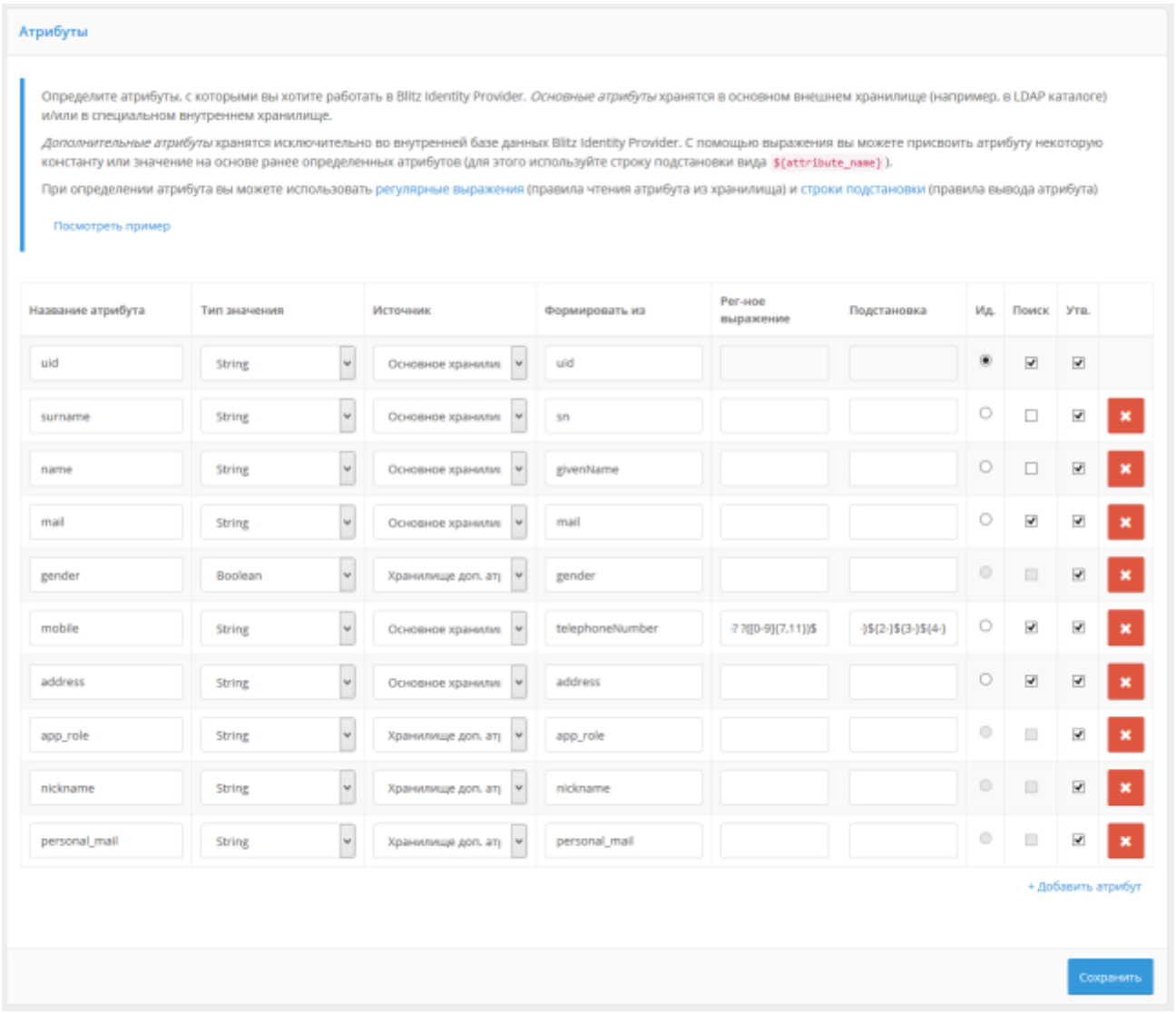

Рисунок 7 – Конфигурирование атрибутов

2.3. Настройка способов аутентификации

2.3.1. Общие сведения

Способы аутентификации настраиваются в разделе *Аутентификация* консоли управления (рис. 8). Все доступные методы аутентификации отнесены либо к первому, либо ко второму фактору (второй фактор используется для «усиления» первого фактора, например, пользователю в дополнение к паролю требуется ввести специальный код, сгенерированный его мобильным приложением). Чтобы включить метод аутентификации, его нужно сначала настроить.

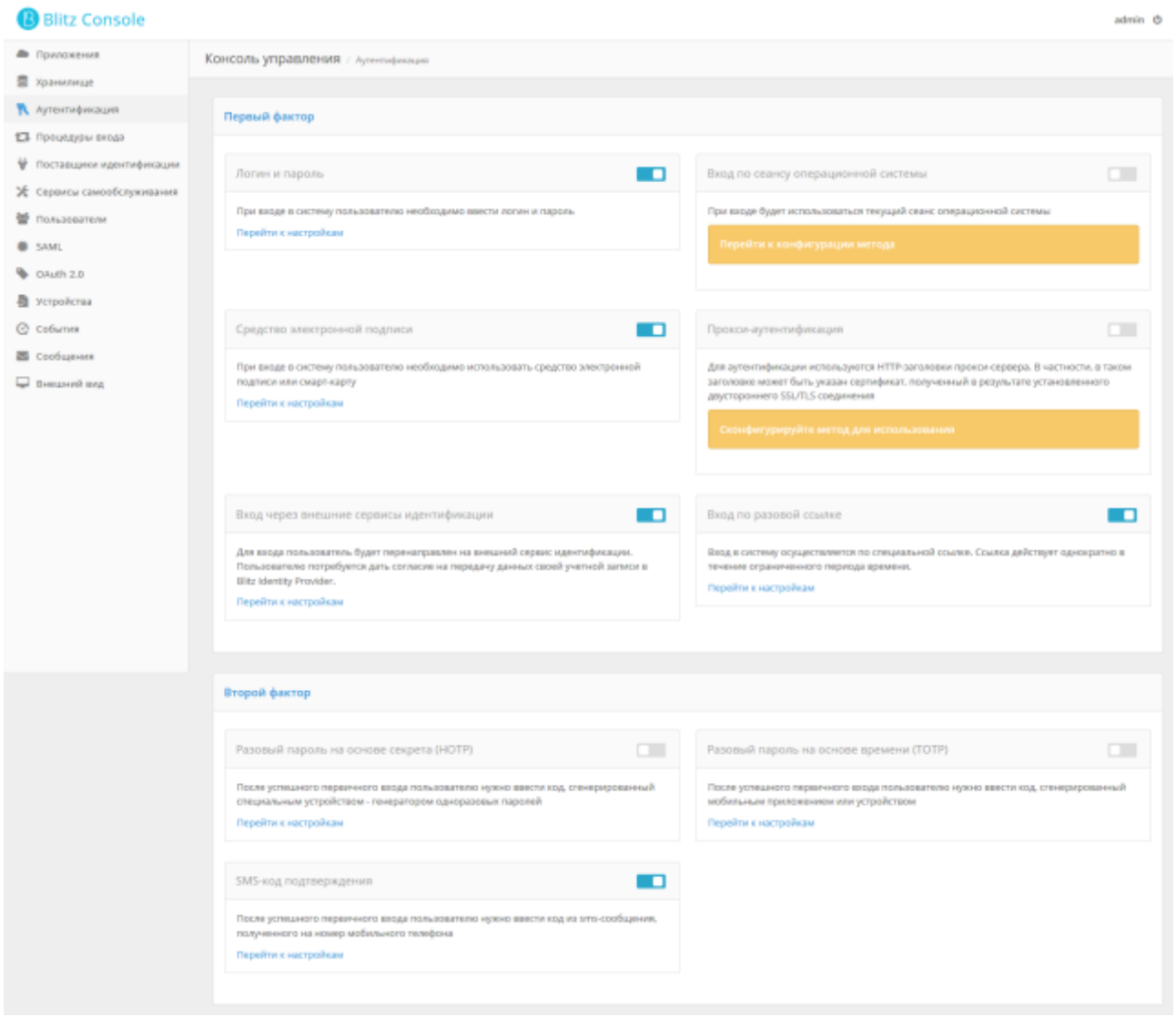

Рисунок 8 – Настройка способов аутентификации (фрагмент)

Помимо настройки каждого метода (настройки описаны далее в подразделах), в консоли можно управлять следующими настройками:

- перечень используемых методов аутентификации (для включения/отключения сконфигурированного метода следует установить переключатель в требуемое положение);
- требуемый уровень аутентификации по умолчанию – например, можно указать, что по умолчанию для всех пользователей будет требоваться второй фактор; напротив, если от пользователей требуется прохождение только первого фактора, то даже при наличии настроенных методов, относящихся ко второму фактору (например, TOTP), они не будут запрошены у пользователя.
- параметры продолжительности сессии (Рисунок 9).

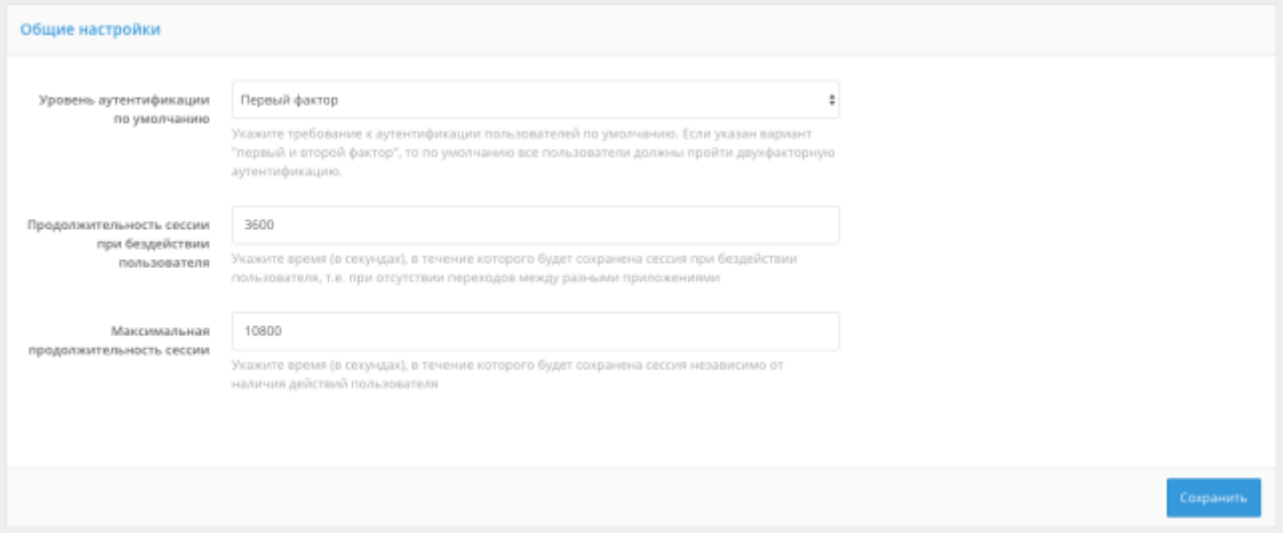

Рисунок 9 – Дополнительные настройки аутентификации

Уровень аутентификации для конкретного пользователя задается в разделе *Пользователи* – на карточке соответствующего пользователя.

2.3.2. Настройка входа по логину и паролю

Для использования входа по логину и паролю необходимо задать правила соответствия – каким образом определять, как введенный логин соотносится с пользователями в хранилище данных.

Для создания правила используется строка подстановки: ${login} – это строка, введенная пользователем в поле «логин». В результате, например, правило «mail=${login}» означает, что строка, введенная пользователем, будет сравниваться с атрибутом mail в хранилище данных (пример настройки см. на *рис*. 10);

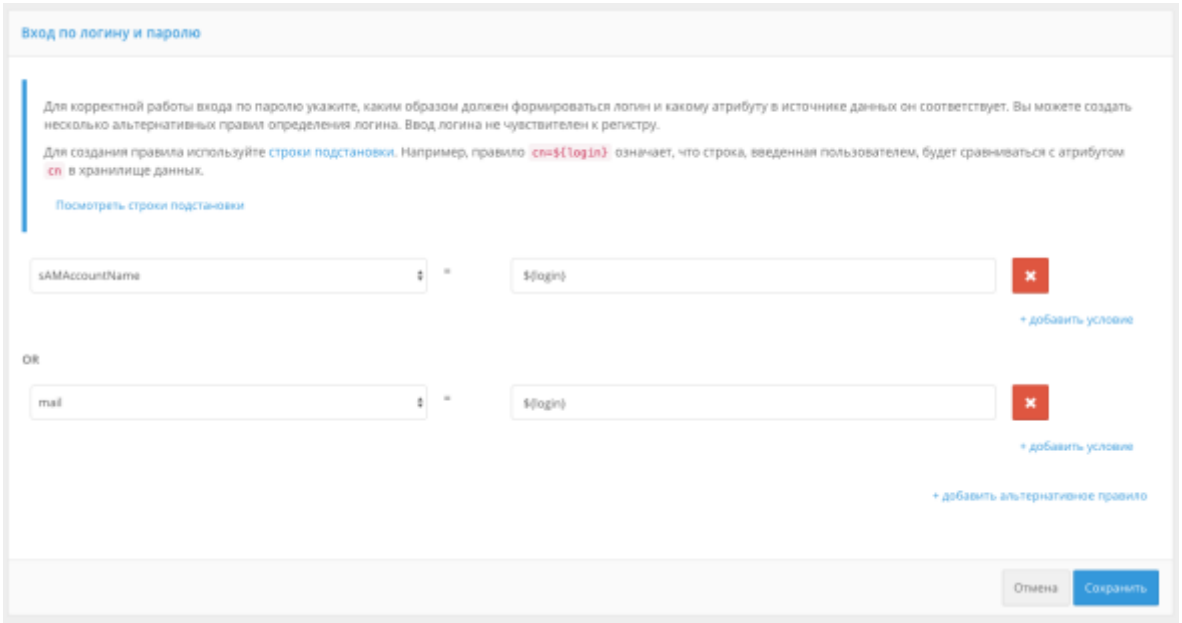

Рисунок 10 – Настройка входа по логину и паролю

Если заданы настройки входа по логину и паролю и отключены другие режимы входа,

стандартная страница входа Blitz Identity Provider будет выглядеть следующим образом (см. Рисунок 11).

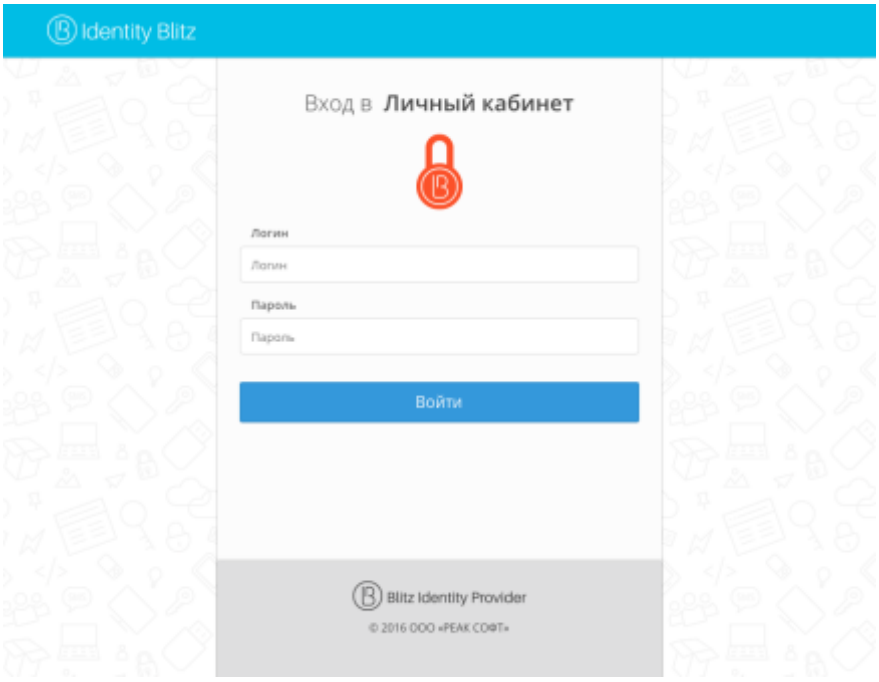

Рисунок 11 – Стандартный вид страницы входа с включенным режимом входа по логину и паролю

2.3.3. Настройка входа с помощью средства электронной подписи

2.3.3.1. Настройка входа

При использовании для аутентификации средства электронной подписи необходимо предварительно загрузить в Blitz Identity Provider сертификаты удостоверяющих центров (CA), подтверждающих подлинность ключей электронной подписи пользователей.

Также необходимо настроить правила соответствия для нахождения учетной записи пользователя в хранилище по его атрибутам из сертификата. Для определения правил используются строки подстановки по аналогии с обработкой логина. Например, правило «cn=${SUBJECT.CN}» означает, что атрибут SUBJECT.CN сертификата будет сравниваться с атрибутом cn в хранилище данных. Возможно указание нескольких условий одновременно, а также указание альтернативных правил.

При конфигурировании входа по электронной подписи можно указать, следует ли этот метод использовать в качестве первого и второго фактора. Если да, то пользователь, прошедший аутентификацию по электронной подписи, будет считаться прошедшим двухфакторную аутентификацию (пример настройки см. Рисунок 12).

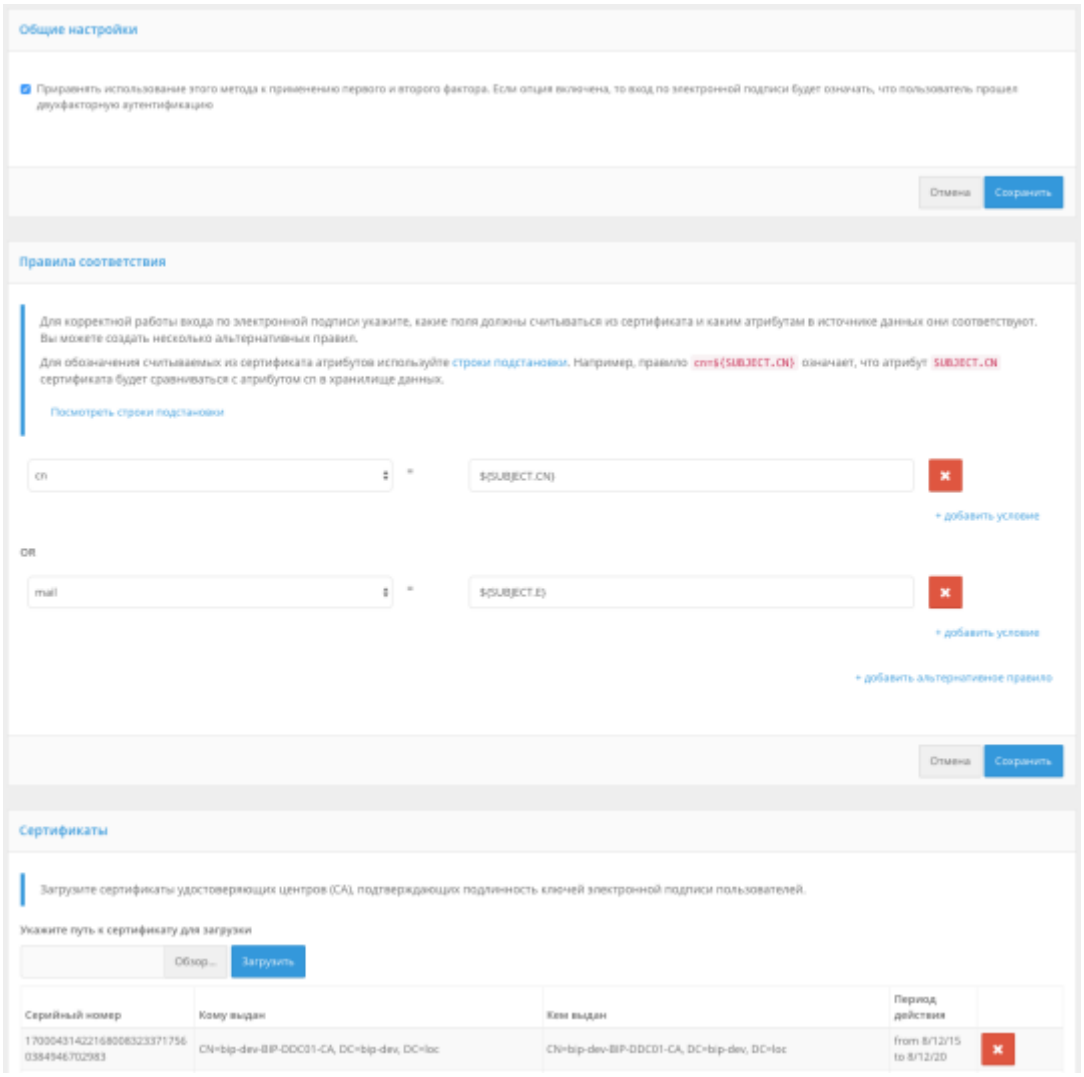

Рисунок 12 – Настройка входа по электронной подписи

Если заданы настройки входа и по логину и паролю, и с помощью средства электронной подписи, то стандартная страница входа Blitz Identity Provider будет выглядеть следующим образом (см. Рисунок 13).

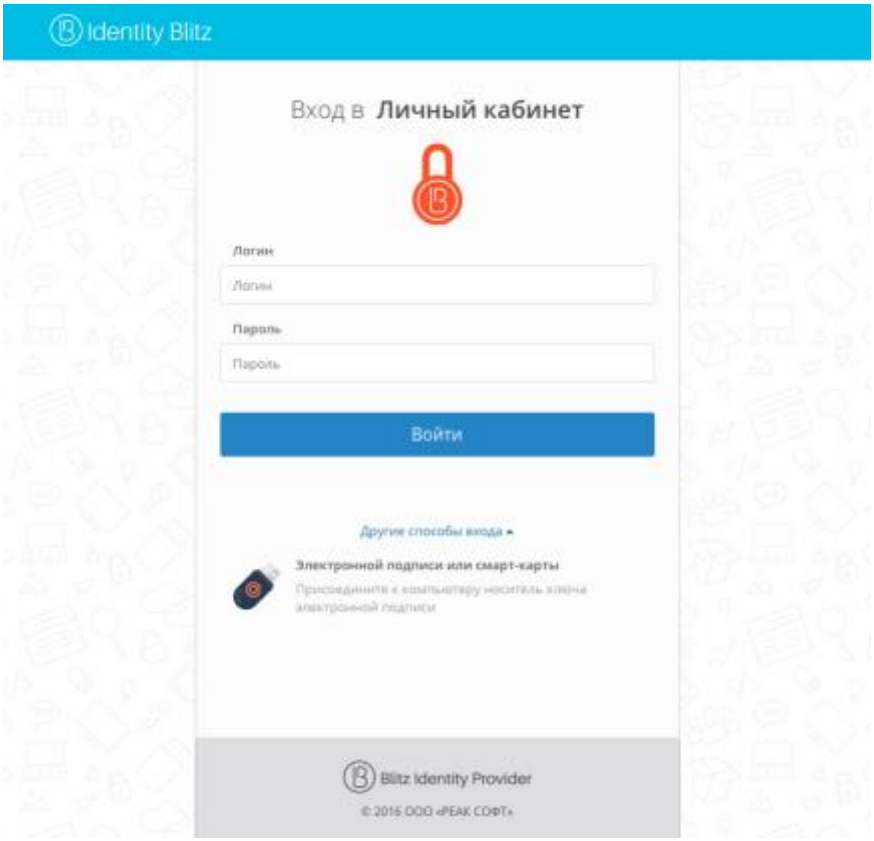

Рисунок 13 – Стандартный вид страницы входа с включенным режимом входа по электронной подписи и по логину и паролю

Если задать режим входа только по электронной подписи, то стандартная страница входа Blitz Identity Provider будет выглядеть следующим образом (см. Рисунок 14).

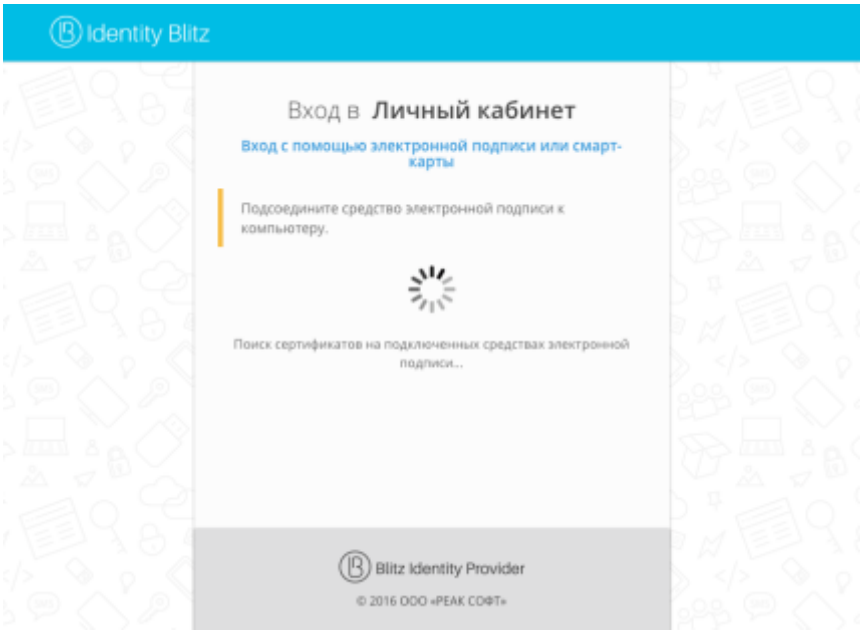

Рисунок 14 – Стандартный вид страницы входа с включенным режимом входа только по электронной подписи

2.3.3.2. Использование и обновление плагина

Для корректной работы входа по электронной подписи на компьютерах пользователей используется специальный плагин – Blitz Smart Card Plugin. При первом входе по электронной подписи пользователю будет предложено установить плагин (рис. 15). После загрузки файла и его запуска пользователю следует пройти все шаги установки плагина. При повторном входе с данного устройства не потребуется устанавливать плагин заново.

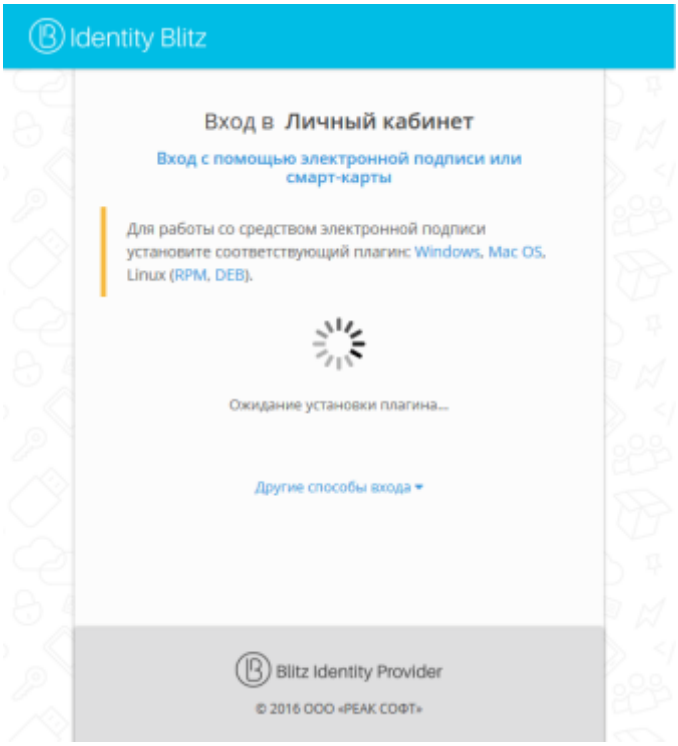

Рисунок 15 – Просьба установить плагин

Blitz Identity Provider поставляется вместе с версией плагина, позволяющей работать со средством электронной подписи в качестве метода аутентификации.

При необходимости обновить версию Blitz Smart Card Plugin следует заменить дистрибутивы плагина – они размещены в директории assets с установкой Blitz Identity Provider, в архиве assets.zip. Структура архива имеет следующий вид:

```
plugins/sc/deb/BlitzScPlugin.deb
plugins/sc/rpm/BlitzScPlugin.rpm
plugins/sc/win/BlitzScPlugin.msi
plugins/sc/mac/BlitzScPlugin.pkg
bdk_2.9.0-SNAPSHOT.jar
```

Иными словами, для обновления плагина необходимо распаковать архив assets.zip, заменить файлы с дистрибутивом плагина под все операционные системы и заархивировать обратно файлы в assets.zip.

2.3.4. Настройка входа через внешние сервисы идентификации

Возможен вход с использованием следующих внешних сервисов идентификации:

- поставщика идентификации социальной сети Facebook;
- поставщика идентификации социальной сети ВКонтакте;
- поставщика идентификации Google;
- Единой системы идентификации и аутентификации (ЕСИА) сайта gosuslugi.ru (только в Enterprise-редакции).
- Blitz Identity Provider, установленного в партнерской организации (только в Enterprise-редакции).

Подключения к внешним сервисам идентификации должны быть предварительно сконфигурированы в консоли управления в разделе «Поставщики идентификации» (см. п. 2.6 документа).

В разделе настроек «Вход через внешние сервисы идентификации» необходимо выбрать, какие из настроенных поставщиков идентификации должны использоваться при входе, а также режим разрешения входа с помощью этих учетных записей. Возможны следующие режимы (см. Рисунок 16):

- «любой пользователь может войти в систему» – для входа с использованием внешнего сервиса идентификации можно воспользоваться любой учетной записью выбранного поставщика идентификации;
- «только пользователи, зарегистрированные администратором» – для возможности входа с использованием внешнего сервиса идентификации учетная запись пользователя должна быть предварительно добавлена администратором через раздел *Пользователи* консоли управления.

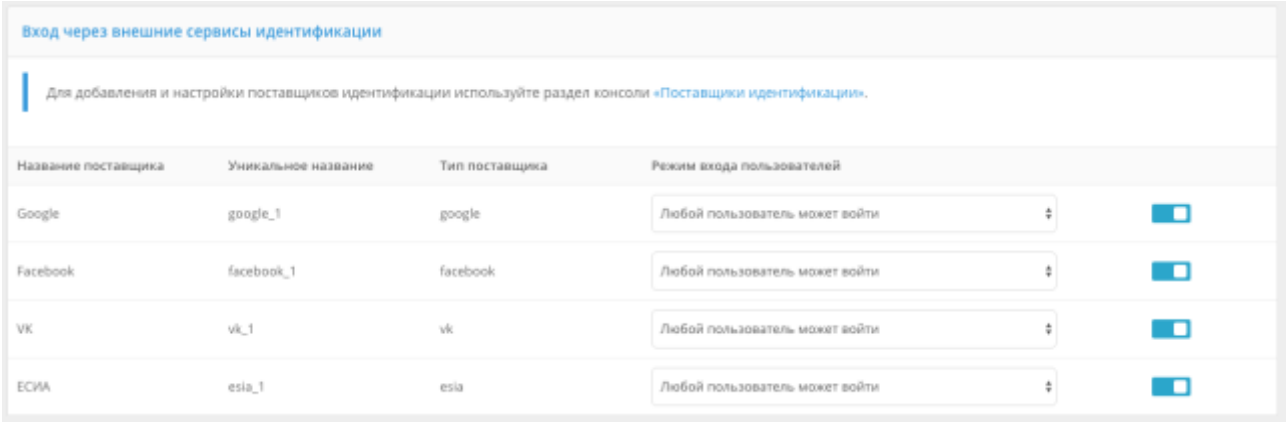

Рисунок 16 – Включение необходимых внешних сервисов идентификации

Стандартный внешний вид страницы входа Blitz Identity Provider при включенных внешних сервисах идентификации показан на рисунке 17.

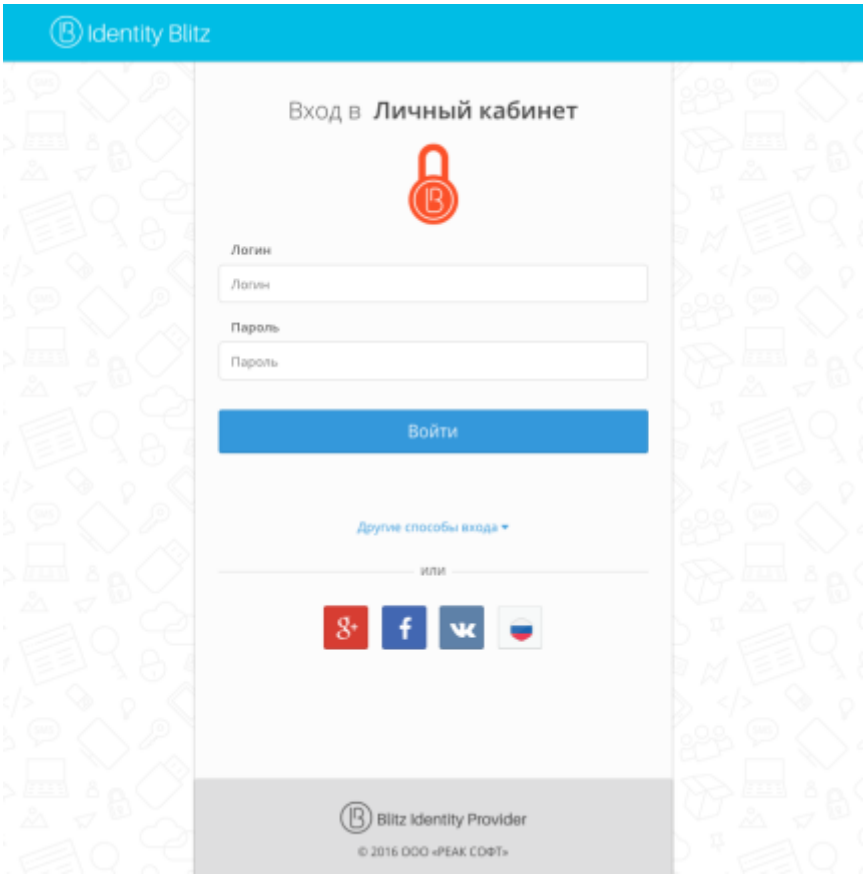

Рисунок 17 – Вид страницы входа с включенными режимами входа с использованием внешних сервисов идентификации

2.3.5. Настройка входа с помощью прокси-аутентификация

Способ входа с использованием прокси-аутентификации доступен только в Enterprise-редакции.

Прокси-аутентификация (аутентификация с помощью прокси-сервера) относится к первому фактору, т.е. она может заменить вход по логину и паролю. Идентификация в этом случае производится по данным, передаваемым в HTTP-заголовках.

При включенной прокси-аутентификации Blitz Identity Provider производит только идентификацию пользователя, тогда как аутентификацию (в результате проверки сертификата) осуществляет прокси-сервер. Включение данного метода аутентификации допустимо в тех случаях, когда все пользователи обращаются к Blitz Identity Provider через прокси-сервер.

Для корректной работы метода необходимо указать:

- требуемые HTTP-заголовки (перечень HTTP-заголовков, которые должны присутствовать для проведения аутентификации пользователя),
- HTTP-заголовок с сертификатом пользователя (заголовок, в котором передается сертификат пользователя, опциональный параметр),

- соответствие значений HTTP-заголовков и идентификационных данных пользователя, имеющихся в хранилище учетных записей.

Также возможна настройка маппинга атрибутов сертификата, передаваемого в HTTP-заголовке, и данных пользователя в хранилище.

Пример настроек входа с помощью прокси-аутентификации представлен на рис. 18.

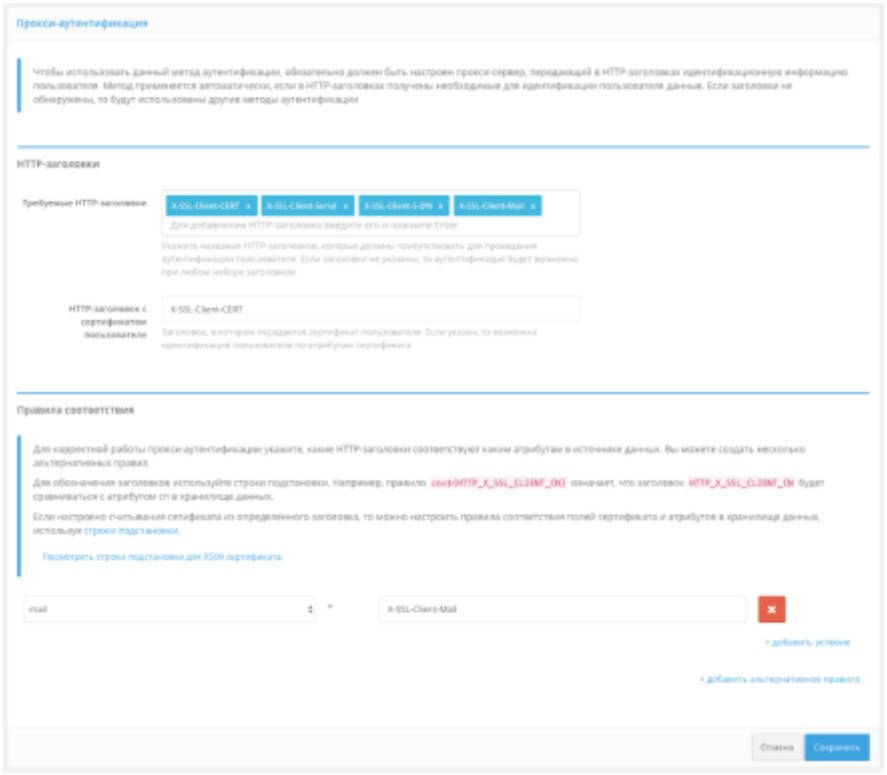

Рисунок 18 – Настройка входа с помощью прокси-аутентификации

2.3.6. Настройка входа с помощью сеанса операционной системы

Способ входа с использованием сеанса операционной системы доступен только в Enterprise-редакции. Этот метод входа позволяет пользователям не проходить дополнительно идентификацию и аутентификацию с помощью Blitz Identity Provider, если они ранее вошли со своего ПК в сеть организации и прошли идентификацию и аутентификацию средствами операционной системы (вошли в домен). Такие пользователи получат возможность сквозной идентификации при доступе ко всем приложениям, подключенным к Blitz Identity Provider.

Для использования возможности входа с помощью сеанса операционной системы в организации должен быть развернут Kerberos-сервер (отдельно или в составе контроллера домена организации) и выполнены следующие настройки (см. описания далее в подразделах):

1. Настройки контролера домена и Kerberos-сервера.
2. Настройки в консоли управления Blitz Identity Provider.
3. Настройки браузеров пользователей.

2.3.6.1. Настройки контролера домена и Kerberos-сервера

В контролере домена необходимо зарегистрировать учетную запись для сервера Blitz Identity Provider. Для созданной учетной записи нужно на странице «Account» в блоке «Account options» оснастки контроллера домена включить настройки «User cannot change password» и «Password never expires». Также в зависимости от используемого контроллером домена шифрования нужно включить одну или несколько опций «This account supports AES 256 bit encryption», «This account supports AES 128 bit encryption», «Use Kerberos DES encryption types for this account».

Далее необходимо создать Service Principal Name (SPN) для идентификации сервера Blitz Identity Provider сервером Kerberos. Это выполняется с помощью следующей команды:

```
ktpass -princ HTTP/idp.company.ru@DOMAIN.LOC -rndPass -mapuser DOMAIN\blitzidpsrv -out C:\temp\spnego_spn.keytab -mapOp set -crypto ALL -ptype KRB5_NT_PRINCIPAL
```

Параметры команды ktpass:

- значение параметра mapuser – имя созданной в домене учетной записи сервера Blitz Identity Provider, например, DOMAIN\blitzidpsrv;
- значение параметра princ – указывается имя SPN сервера с Blitz Identity Provider для идентификации в среде Kerberos. Это имя состоит из имени хоста сервера с Blitz Identity Provider, имени Kerberos Realm в верхнем регистре (обычно совпадает с именем домена) и используемого транспортного протокола (HTTP). Пример значения SPN – HTTP/idp.company.ru@DOMAIN.LOC.
- параметр rndPass без значения – показывает, что нужно сгенерировать случайный пароль. Если нужно указать пароль явно, то вместо –rndPass нужно использовать опцию –pass.
- параметр mapOp – если задан в значение add, то указывает, что новый SPN нужно добавить к существующим. Если задано значение set, то запись SPN должна перезаписать существующую.
- параметр out – задает путь к генерируемому keytab-файлу. Например, C:\temp\spnego_spn.keytab.

Сгенерированный keytab-файл необходимо сохранить. Он будет необходим для последующей настройки в консоли управления Blitz Identity Provider.

2.3.6.2. Настройки в консоли управления Blitz Identity Provider

Необходимо перейти в консоли управления в разделе *Аутентификация* к настройкам способа входа «Вход по сеансу операционной системы». В открывшемся окне (рис. 19) задать настройки Kerberos-сервера:

- имя Kerberos Realm – обычно совпадает с именем домена;

- имя Kerberos-сервера, являющегося KDC-серверов.

Рисунок 19 – Настройки Kerberos-сервера

Далее необходимо задать настройки SPN, зарегистрированного для сервера Blitz Identity Provider (Рисунок 20). Потребуется указать имя SPN и загрузить сгенерированный ранее для SPN keytab-файл.

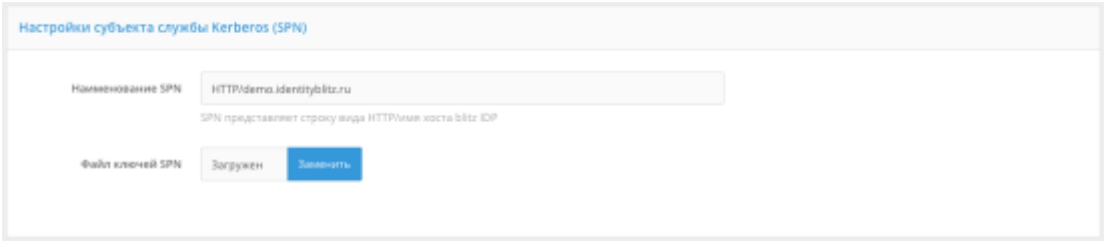

Рисунок 20 – Настройки SPN для Blitz Identity Provider

Далее необходимо определить параметры соответствия Kerberos-токена и учетной записи в Blitz Identity Provider (Рисунок 21). Например, можно задать соответствие, что получаемый из Kerberos-токена идентификатор пользователя (username) должен соответствовать атрибуту sAMAccountName, получаемому из Microsoft Active Directory / Samba4 для учетной записи.

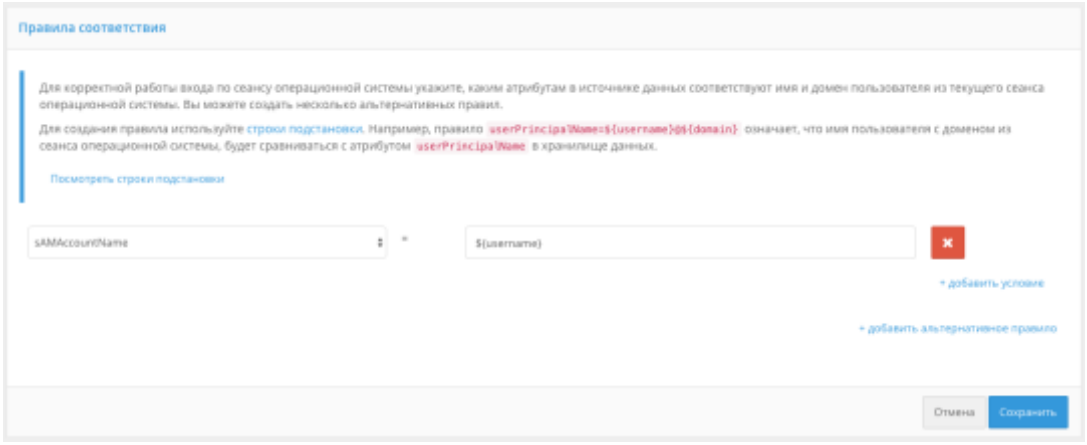

Рисунок 21 – Настройка соответствия Kerberos-идентификатора пользователя и его учетной записи в хранилище

Далее необходимо установить параметры задержек при использовании метода входа с использованием сеанса операционной системы (Рисунок 22).

Blitz Identity Provider предоставляет два возможных сценария использования входа по сеансу операционной системы:

Основной сценарий. Пользователи входят в операционную систему, и после этого должны сквозным образом входить во все приложения, подключенные к Blitz Identity Provider. Предоставлять пользователям возможность войти в приложения под другой учетной записью не требуется. В этом случае нужно установить «Время задержки перед запуском метода» в 0 секунд. При обращении к приложению сразу будет произведена попытка сквозного входа по сеансу операционной системы.

Дополнительный сценарий. Пользователи не всегда имеют возможность войти в домен операционной системы, либо пользователям в некоторых случаях необходима возможность войти в приложения под другой учетной записью чем та, что они использовали для входа в домен. В этом случае нужно установить «Время задержки перед запуском метода» в такое количество секунд, которое будет даваться пользователю для возможности отменить автоматический вход с использованием сеанса операционной системы.

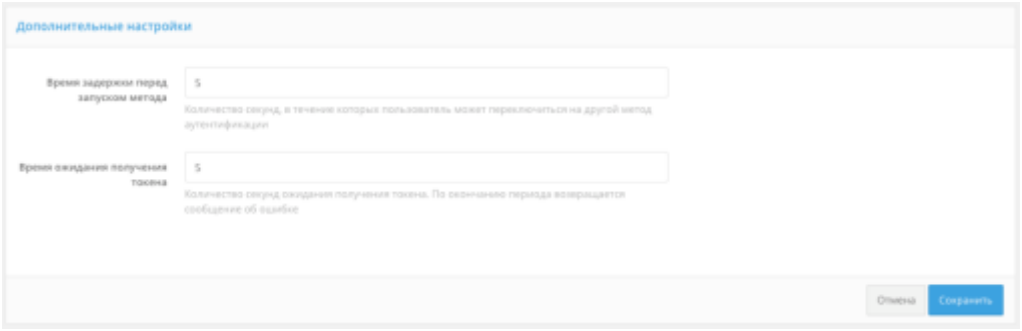

Рисунок 22 – Настройка входа с помощью прокси-аутентификации

Время ожидания получения токена нужно установить достаточным, чтобы Kerberos-сервер успевал предоставить ответ Blitz Identity Provider. Обычно достаточно установить 3-5 секунд.

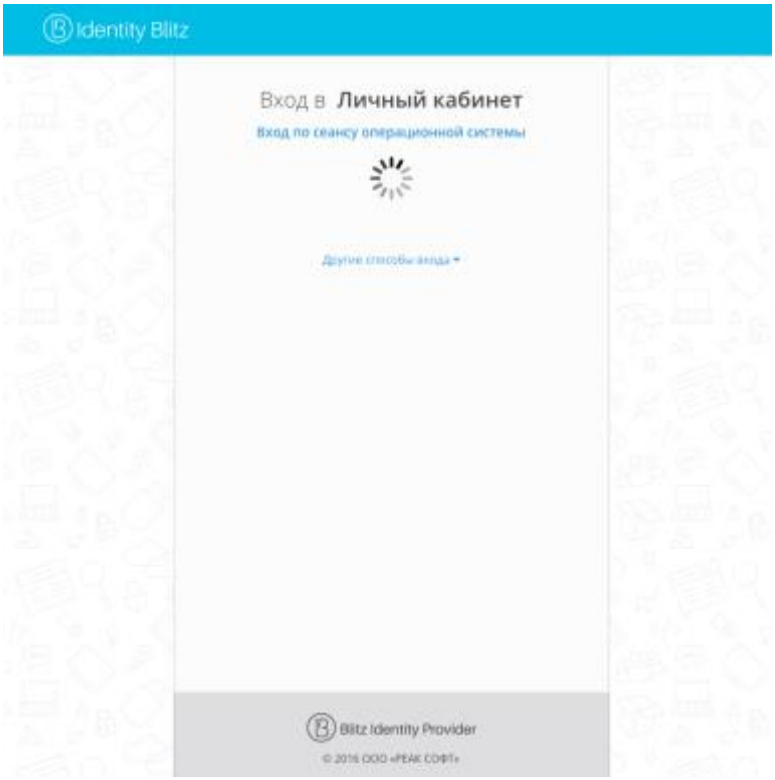

Рисунок 23 – Пример экрана входа при использовании режима входа через сеанс операционной системы

2.3.6.3. Настройки браузеров пользователей

В зависимости от используемого пользователем браузера может потребоваться его дополнительная настройка для поддержки Kerberos-идентификации.

Для Google Chrome в Windows и Apple Safari в macOS отдельная настройка не требуется.

Для Google Chrome в macOS и в Linux однократно нужно провести запуск Google Chrome специальным образом:

```
"/Applications/Google Chrome.app/Contents/MacOS/Google Chrome" --args --auth-server-whitelist="idp.domain.ru" --auth-negotiate-delegate-whitelist="idp.domain.ru"
```

Где в качестве idp.domain.ru нужно указать URL сайта Blitz Identity Provider.

Для Microsoft Internet Explorer нужно задать следующие настройки:

- в меню «Сервис → Свойства обозревателя → Безопасность → Местная интрасеть» нажать кнопку «Сайты». В открывшемся окне нажать кнопку «Дополнительно» и внести сайт с Blitz Identity Provider в список сайтов «Местная интрасеть» (см. Рисунок 24).
- в меню «Сервис → Свойства обозревателя → Безопасность → Местная интрасеть» нажать кнопку «Другой…». В открывшемся окне найти настройку «Проверка подлинности пользователя → Вход». Установить ее в значение «Автоматический вход в сеть только в зоне интрасети» (см. Рисунок 25).

- перезапустить браузер.

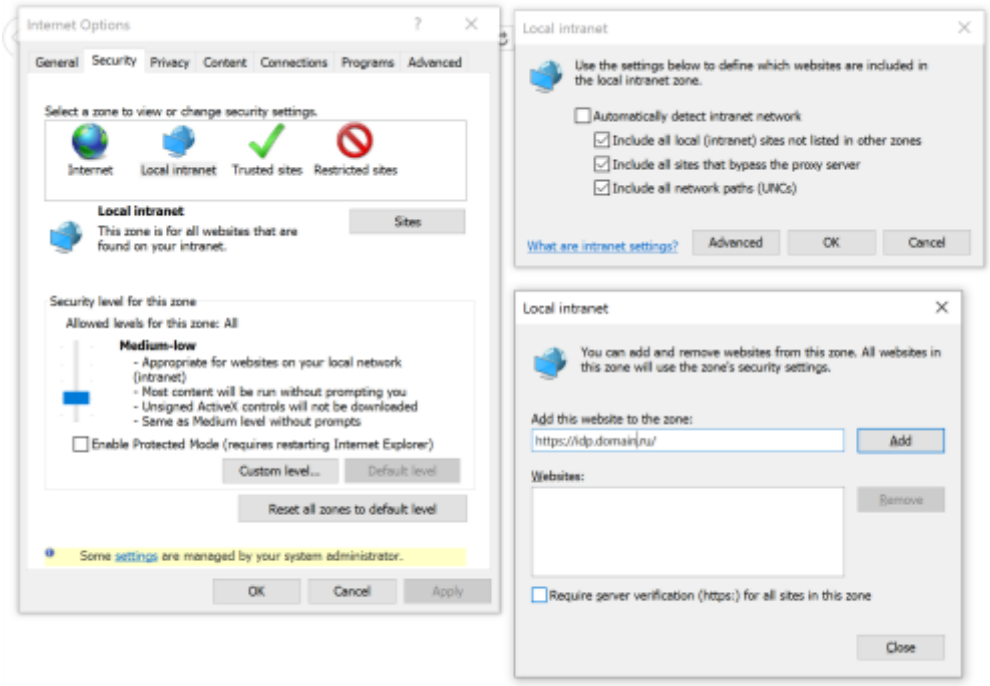

Рисунок 24 – Настройки Internet Explorer для Kerberos – включение Blitz Identity Provider в ресурсы Локальной вычислительной сети

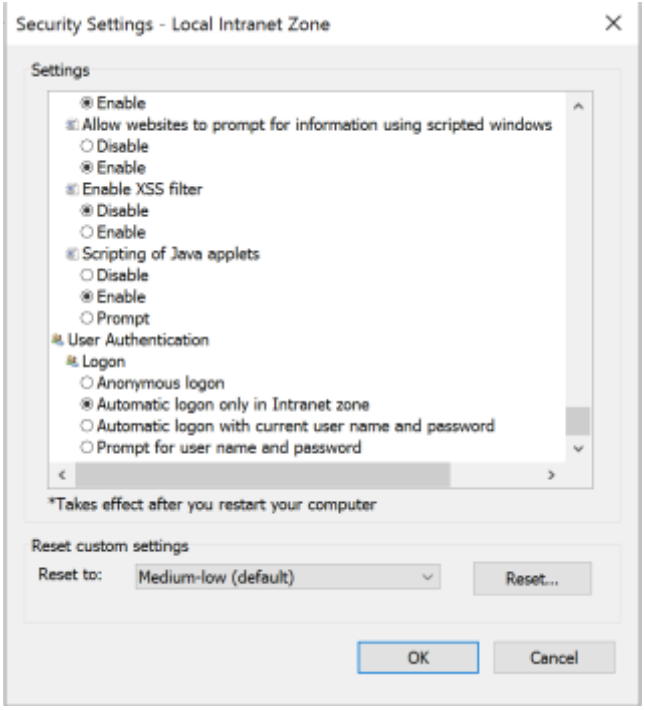

Рисунок 25 – Настройки Internet Explorer для Kerberos – включение встроенной идентификации

Для Mozilla Firefox нужно задать следующие настройки:

- в адресной строке браузера ввести about:config и нажать Enter. В следующем окне ввести «network.nego» в поле «Фильтры». Дважды нажать на найденной записи

«network.negotiate-auth.trusted-uris» и установить в ней значение URL сайта с Blitz Identity Provider, например, idp.domain.ru. Закрыть всплывающее окно кнопкой ОК.

- дважды нажать на найденной записи «network.negotiate-auth.delegation-uris» и установить в ней значение URL сайта с Blitz Identity Provider, например, idp.domain.ru. Закрыть всплывающее окно кнопкой ОК.
- перезапустить браузер.

2.3.7. Усиленная аутентификация с помощью разового пароля на основе состояния (HOTP)

Blitz Identity Provider позволяет настроить использование HOTP-устройств в качестве средств для проверки второго фактора аутентификации. В качестве HOTP-устройства можно использовать любой аппаратный ключ, совместимый со стандартом RFC4226 «HOTP: An HMAC-Based One-Time Password Algorithm»[11].

Для использования HOTP необходимо:

- настроить и включить этот метод аутентификации (см. Рисунок 26);
- загрузить в Blitz Identity Provider файл с описаниями HOTP-устройств. Файл с описаниями, как правило, предоставляет поставщик HOTP-устройств. Для загрузки файла с описанием используется раздел *Устройства* в консоли управления Blitz Identity Provider (см. раздел 2.9 документа);
- выдать пользователю HOTP-устройство и привязать его к учетной записи пользователя. Привязку можно выполнить двумя способами – либо администратор привязывает устройство по серийному номеру к учетной записи пользователя в консоли управления в разделе *Пользователи*, либо пользователь привязывает устройство к своей учетной записи самостоятельно с использованием веб-приложения «Профиль пользователя».

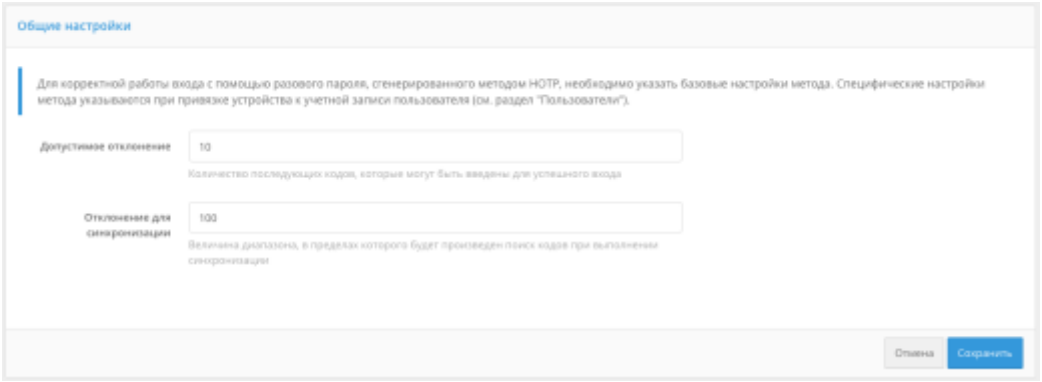

Рисунок 26 – Настройки HOTP-аутентификации

Для использования разовых паролей основе секрета (HOTP) необходимо задать

[11] https://tools.ietf.org/html/rfc4226

максимальное допустимое отклонение при проверке кода — количество последующих кодов (например, если пользователь случайно нажал кнопку генерирования нового пароля и не использовал его в процессе аутентификации), при котором аутентификация пройдет успешно. При этом при вводе пользователем правильного кода Blitz Identity Provider автоматически восстановит синхронизацию с устройством.

Если пользователь многократно будет нажимать на устройстве кнопку выработки кода и не будет использовать код для аутентификации, то устройство окажется рассинхронизированным с сервером Blitz Identity Provider. В этом случае при очередном входе пользователя в Blitz Identity Provider ему будет предложено провести процедуру сверки устройства путем ввода трех последовательно выработанных устройством кодов аутентификации. Далее в соответствии с заданной настройкой «Отклонение для синхронизации» Blitz Identity Provider проверит, встречается ли введенная пользователем последовательность кодов, и восстановит синхронизацию с устройством в случае успеха.

На рисунках приведен пример внешнего вида страницы входа Blitz Identity Provider при запросе ввода пользователем кода подтверждения, выработанного HOTP-устройством (см. Рисунок 27), и в процессе проведения процедуры сверки устройства (см. Рисунок 28).

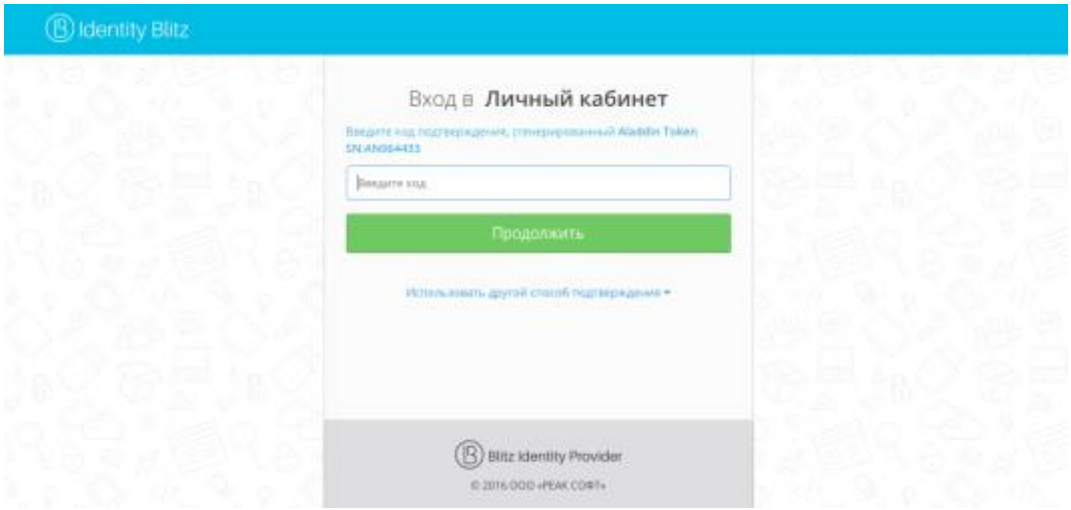

Рисунок 27 – Запрос кода подтверждения, выработанного HOTP-устройством

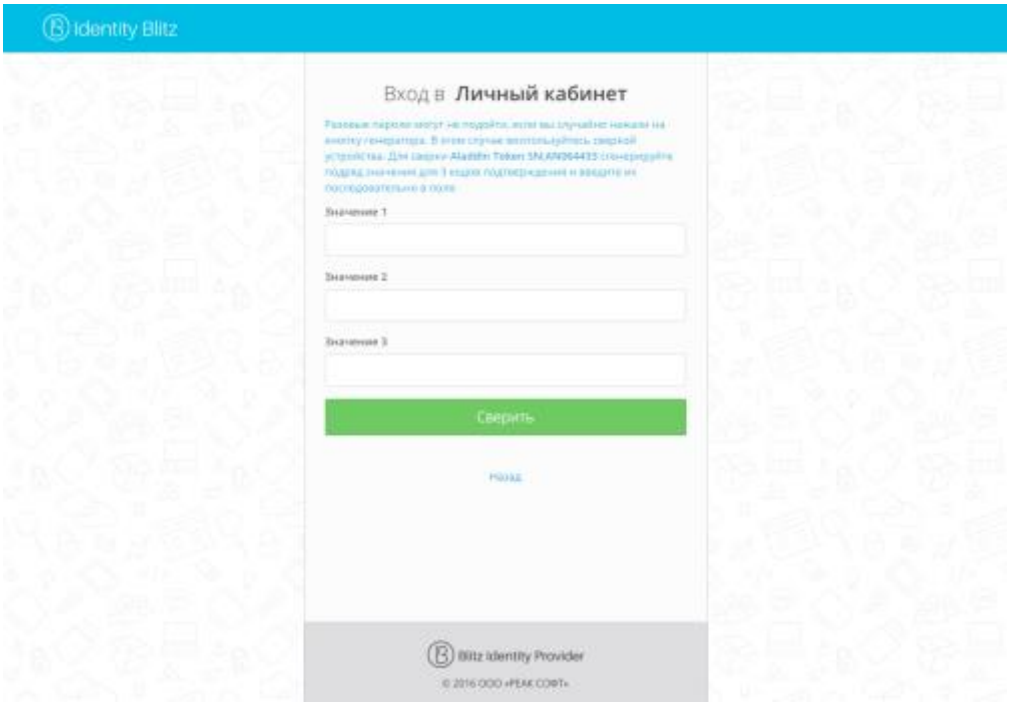

Рисунок 28 – Процедура сверки HOTP-генератора

2.3.8. Усиленная аутентификация с помощью разового пароля основе времени (TOTP)

Blitz Identity Provider позволяет настроить использование TOTP-устройств в качестве средств для проверки второго фактора аутентификации. В качестве TOTP-устройства можно использовать любые средства, совместимые со стандартом RFC6238 «TOTP: Time-Based One-Time Password Algorithm»[12]. В качестве таковых могут быть:

- аппаратные ключи-генераторы разовых паролей на основе времени;
- мобильные приложения[13].

На рисунке приведен пример внешнего вида страницы входа Blitz Identity Provider при запросе ввода пользователем кода подтверждения (см. Рисунок 29).

[12] https://tools.ietf.org/html/rfc6238
[13] Наиболее известные зарубежные приложения для выработки TOTP-кодов: Google Authenticator, Authy, FreeOTP Authenticator, Microsoft Authenticator. Наиболее известные российские приложения для выработки TOTP-кодов: Яндекс.Ключ, Bitrix OTP.

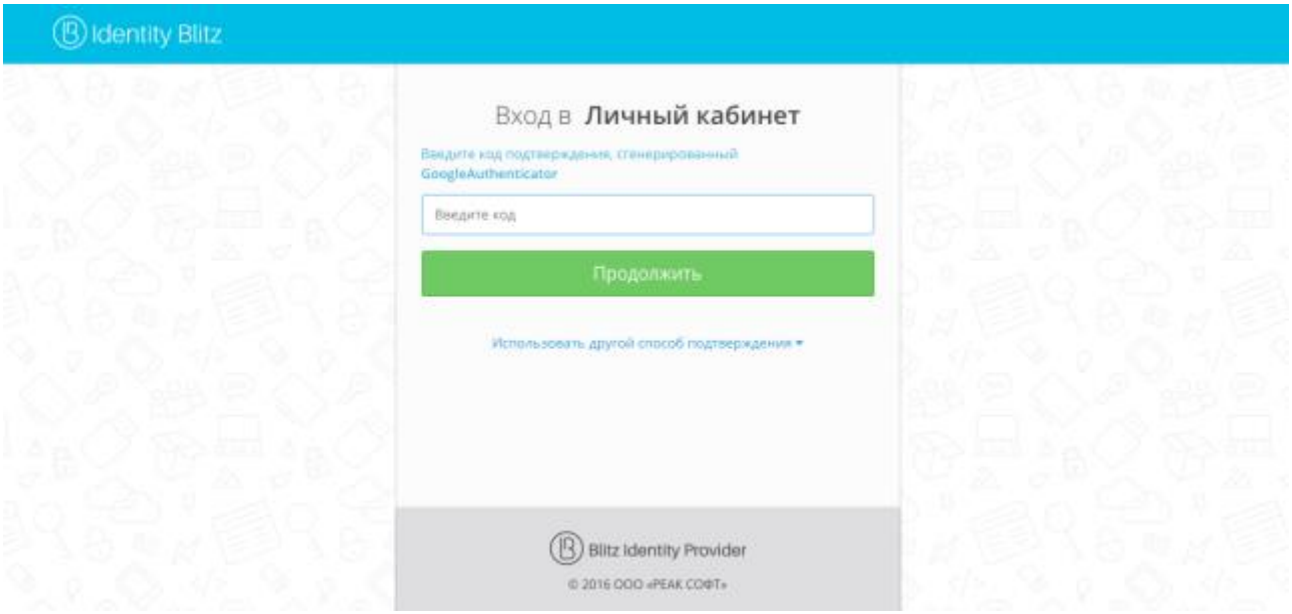

Рисунок 29 – Запрос кода подтверждения, выработанного TOTP-приложением

Независимо от используемого средства, общие настройки TOTP-аутентификации задаются в разделе *Аутентификация*, метод аутентификации – «Разовый пароль на основе времени (TOTP)». В настройках необходимо указать:

1. Допустимое отклонение при проверке кода (количество предыдущих / последующих кодов). По умолчанию оба значения равны 1: пользователь при входе может ввести как текущий пароль, так и следующий или предыдущий (т.е. сгенерированный в соседних временных интервалах). Такая необходимость может возникнуть, например, для компенсации возможной незначительной рассинхронизации серверного времени и времени на TOTP-устройстве пользователя (мобильном приложении или аппаратном ключе).
2. Настройка отображения генераторов разовых паролей, которая включает в себя атрибут с именем пользователя и название единой системы входа. Эти параметры будут отображаться в мобильном приложении после привязки учетной записи пользователя.
3. Ссылки на приложения-генераторы разовых паролей. Следует указать ссылки на приложения, которые рекомендуются использовать.

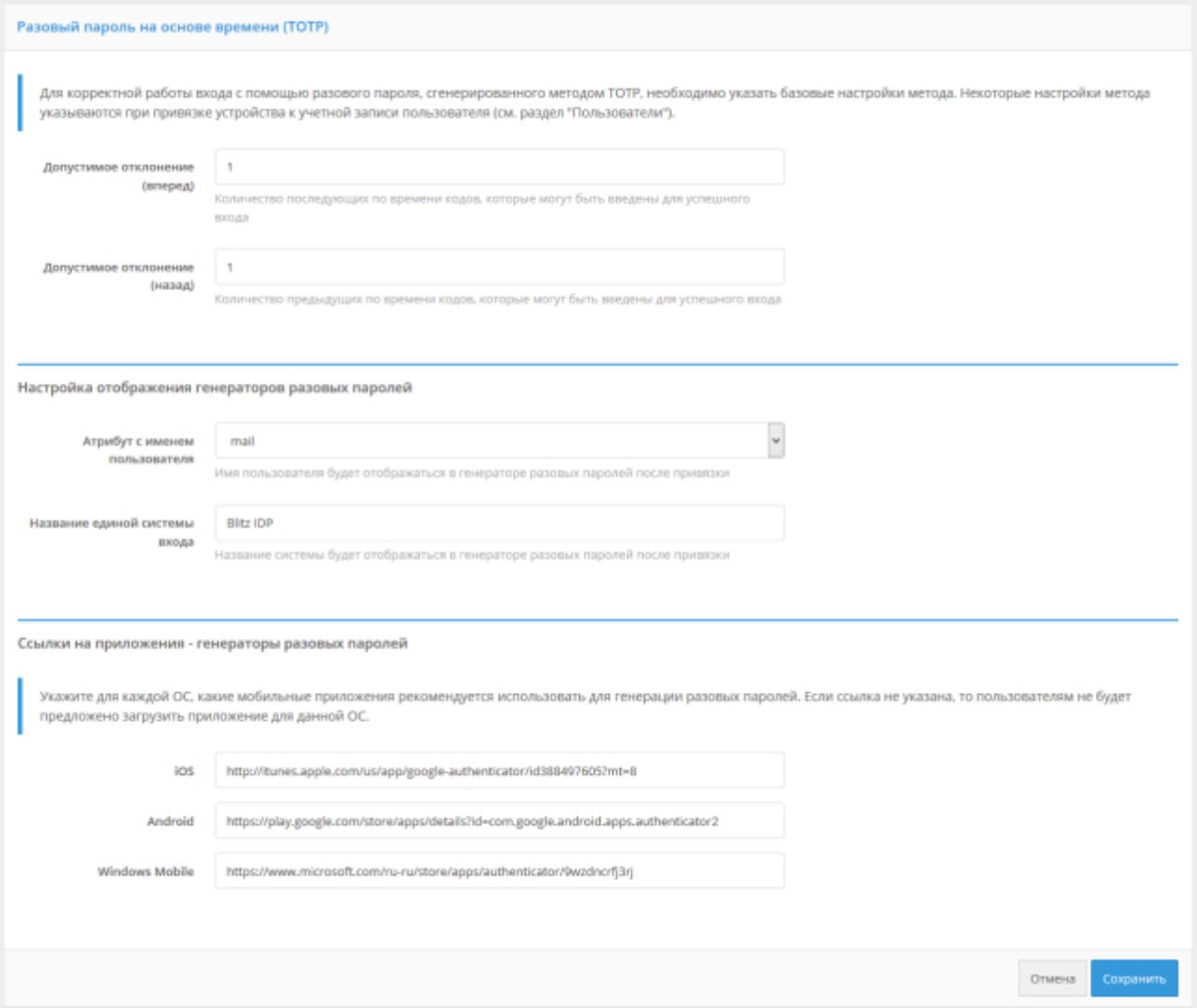

Рисунок 30 – Общие настройки TOTP-аутентификации

Привязка устройств через консоль управления отличается в зависимости от того, используются аппаратные ключи-генераторы разовых паролей или мобильные приложения.

2.3.8.1. Привязка аппаратных генераторов

Для привязки аппаратных ключей-генераторов разовых паролей необходимо перейти в раздел *Устройства*. Затем загрузить файл с описаниями TOTP-устройств, который, как правило, предоставляет поставщик TOTP-устройства. Подробнее этот процесс описан в разделе 2.9 документа.

После загрузки файла следует:

- перейти к учетной записи пользователя, которому необходимо привязать устройство (см. п. 2.8.4.2 документа);
- найти раздел «Генератор паролей на основе времени (TOTP)»;
- выбрать «Другой тип»;
- ввести серийный номер необходимого устройства и текущий разовый код

подтверждения.

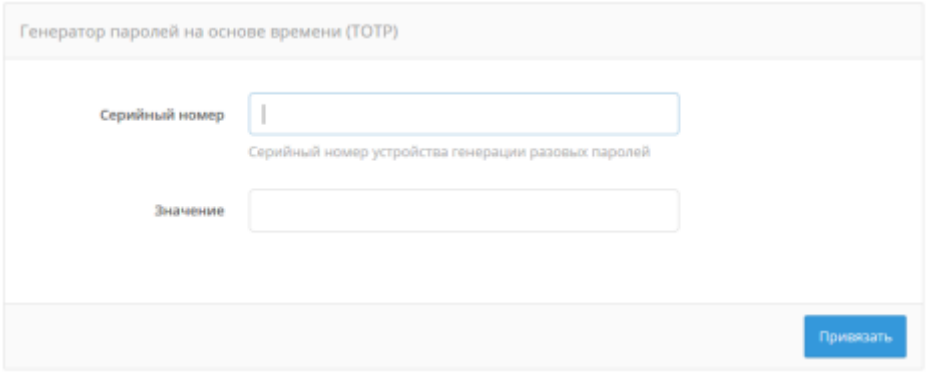

Рисунок 31 – Привязка аппаратного TOTP-генератора

2.3.8.2. Привязка мобильного приложения

Для привязки мобильного приложения следует:

- перейти к учетной записи пользователя, которому необходимо привязать мобильное приложение (см. п. 2.8.4.2 документа);
- найти раздел «Генератор паролей на основе времени (TOTP)»;
- выбрать «GoogleAuthenticator»;
- при необходимости отредактировать название мобильного приложения;
- с помощью мобильного приложения сфотографировать отображаемый QR-код или ввести в приложение строчку-секрет.

Также пользователь может самостоятельно привязать мобильное приложение, генерирующее TOTP-коды, в своем Личном кабинете.

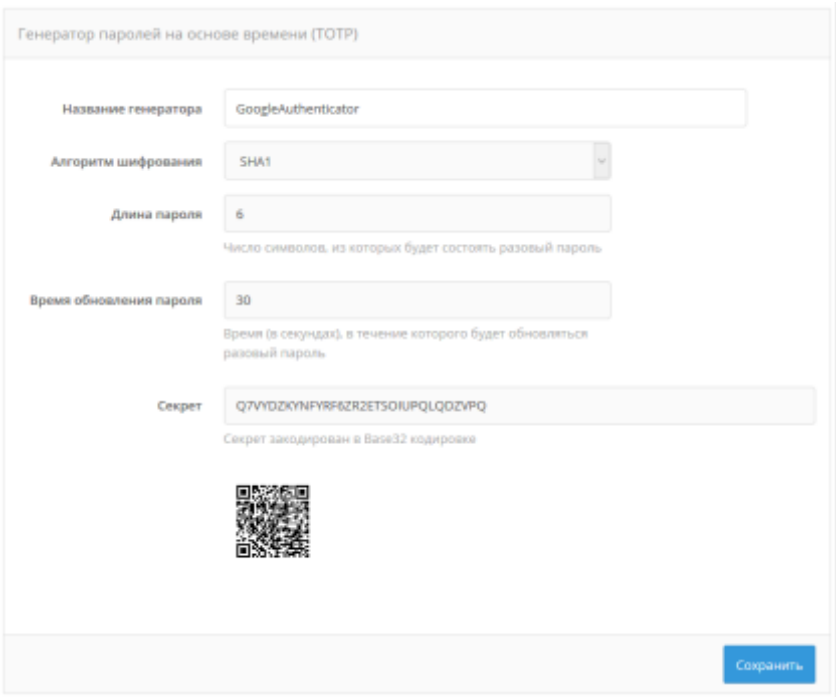

Рисунок 32 – Привязка мобильного приложения, генерирующего TOTP-коды

2.3.9. Усиленная аутентификация с помощью разовых паролей, отправляемых в виде sms-сообщений

Blitz Identity Provider позволяет настроить использование направляемых по SMS кодов подтверждения в качестве механизма проверки второго фактора аутентификации.

Для использования кодов подтверждения по SMS необходимо:

- настроить и включить этот метод аутентификации (см. Рисунок 33). Для корректной работы метода обязательно нужно выбрать атрибут, в котором сохранен номер мобильного телефона пользователя. При необходимости можно изменить длину кода подтверждения и время его действия;
- настроить подключение Blitz Identity Provider к SMS-шлюзу (см. раздел 2.11.1).

Следует помнить, что если у пользователи не задан номер мобильного телефона, то он на сможет использовать этот метод подтверждения входа.

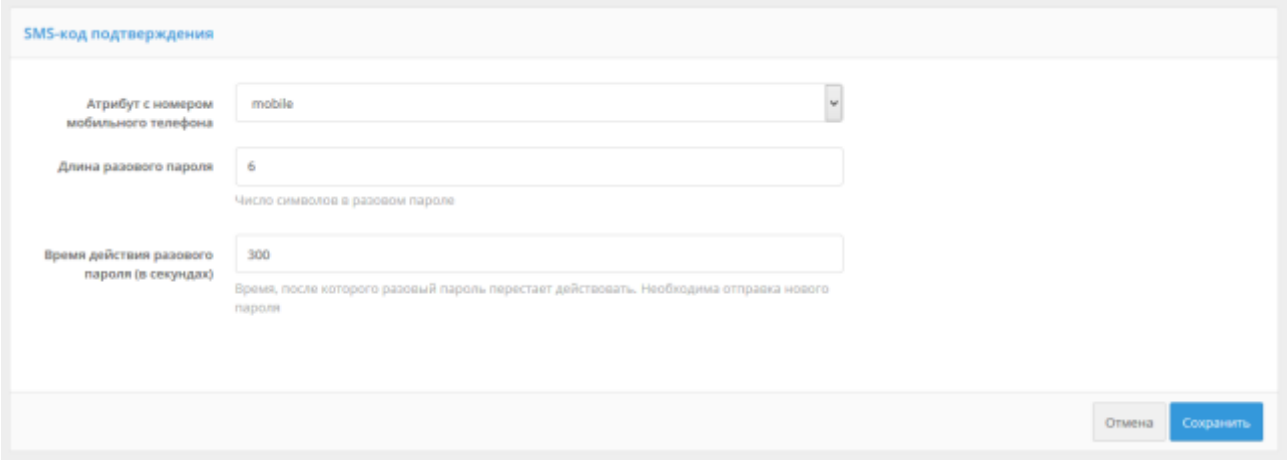

Рисунок 33 – Настройки SMS-кодов для усиленной аутентификации

На рисунках 34, 35 приведен пример внешнего вида страницы входа Blitz Identity Provider при запросе ввода пользователем кода подтверждения, отправленного по SMS.

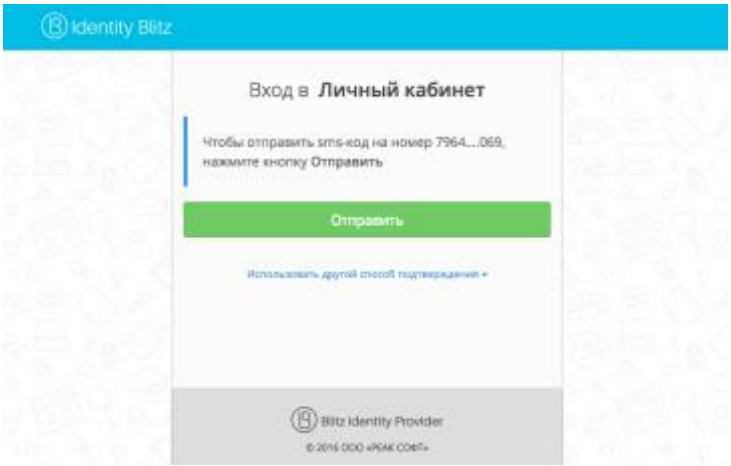

Рисунок 34 – Инициирование отправки SMS-кода

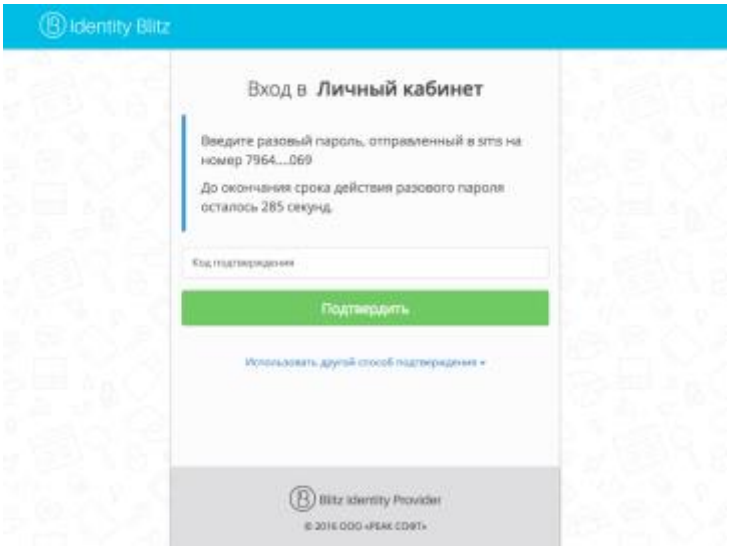

Рисунок 35 – Ожидание ввода пользователем SMS-кода

2.4. Регистрация приложений

Регистрация приложений в Blitz Identity Provider необходима для того, чтобы приложения могли использовать предоставляемые Blitz Identity Provider сервисы:

- запрашивать идентификацию и аутентификацию пользователей;
- вызывать REST-сервисы Blitz Identity Provider (доступно только в Enterprise-редакции).

Управление приложениями осуществляется в разделе *Приложения* консоли управления (см. Рисунок 36).

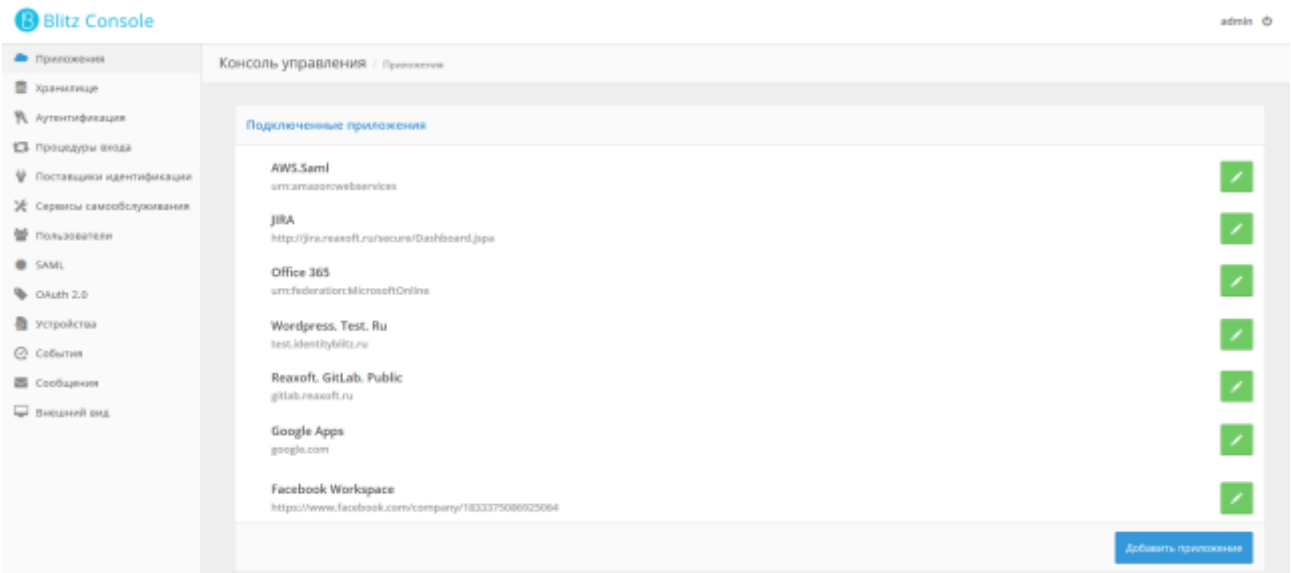

Рисунок 36 – Главный экран настройки приложений

2.4.1. Создание учетной записи нового приложения

Для подключения нового веб-приложения необходимо перейти в раздел *Приложения* консоли и выбрать пункт «Добавить приложение». Это действие запустит мастер подключения нового приложения, работа которого включает в себя следующие шаги:

Шаг 1. Базовые настройки. Требуется указать идентификатор подключаемого приложения (при подключении по протоколу SAML идентификатор соответствует entityID, при подключении по OAuth 2.0 – client_id, при задании идентификатора для OAuth 2.0 **недопустимо** использовать двоеточие), его название и домен, т.е. URL, по которому доступно данное приложение (рис. 37).

Название приложения используется в дальнейшем в Blitz Identity Provider при отображении на странице входа в случае инициирования приложением запроса на идентификацию пользователя.

Домен приложения используется при необходимости перенаправления пользователя в приложение из веб-страниц Blitz Identity Provider. Перенаправление осуществляется на указанный домен или на переданный в процессе взаимодействия с Blitz Identity Provider специализированный redirect_url, но при этом выполняется сверка, что redirect_url соответствует заданному в настройке приложения домену.

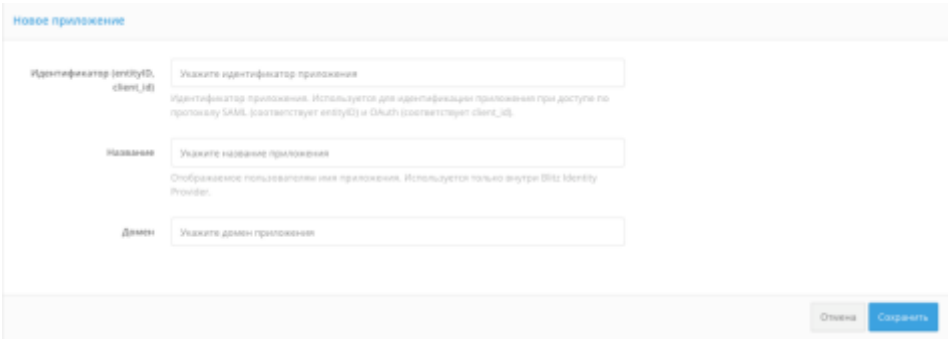

Рисунок 37 – Базовые настройки приложения

Шаг 2. Специфические настройки (см. Рисунок 38). После добавления приложения необходимо перейти к редактированию его специфических настроек, т.е. тех, которые зависят от требуемого протокола взаимодействия приложения с Blitz Identity Provider.

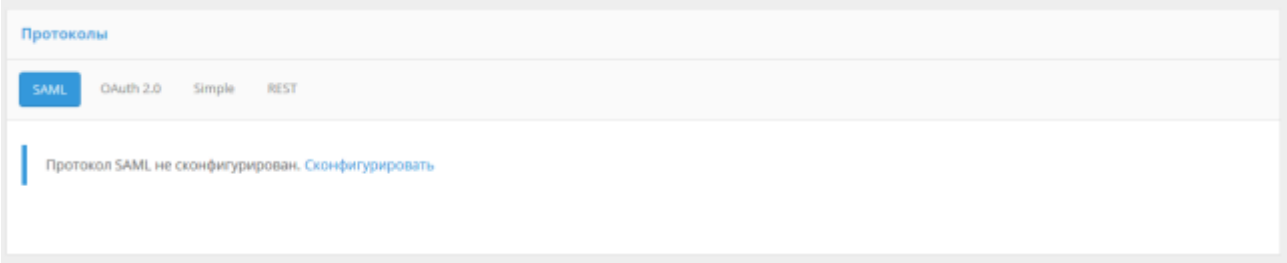

Рисунок 38 – Специфические настройки приложения

Поддерживаются следующие протоколы взаимодействия:

- SAML – для подключения приложений по SAML 1.0, 1.1, 2.0 для идентификации и аутентификации пользователей.
- OAuth 2.0 – для подключения приложений по OAuth 2.0, OpenID Connect 1.0 (OIDC) для идентификации и аутентификации пользователей.
- Simple – для подключения веб-приложений для осуществления идентификации и аутентификации с помощью подстановки в приложение логина и пароля с proxy-сервера, если приложение не поддерживает возможности подключения по SAML/OIDC. Доступно только в Enterprise-редакции.
- REST – для подключения приложений, использующих REST-сервисы Blitz Identity Provider по регистрации/изменению учетных записей, управлению устройствами аутентификации пользователей. Доступно только в Enterprise-редакции.

Если организация планирует разработку или доработку собственных приложений для подключения их к Blitz Identity Provider, то разработчикам необходимо ознакомиться с документом «Руководство по интеграции». В том же документе описана и настройка приложений для использования REST-сервисов Blitz Identity Provider.

Если организация планирует подключить к Blitz Identity Provider приложения, имеющие штатную поддержку подключения по SAML 1.0, SAML 1.1, SAML 2.0 или OIDC (OpenID Connect 1.0, OAuth 2.0), то можно обратиться в ООО «РЕАК СОФТ» за специализированной инструкцией по настройке этих приложений, либо воспользоваться описаниями в последующих подразделах, описывающих метод настройки на стороне Blitz Identity Provider подключения произвольного приложения с поддержкой SAML/OIDC.

Инструкции по применению протокола Simple для подключения к Blitz Identity Provider веб-приложений, в которых отсутствует поддержка SAML/OIDC, в данном руководстве не приводятся и могут быть высланы в виде отдельной инструкции под интеграцию конкретного приложения с Blitz Identity Provider. Для получения специализированной инструкции необходимо обратиться с запросом в ООО «РЕАК СОФТ».

2.4.2. Настройка SAML

При подключении приложения по SAML необходимо задать следующие настройки (см. Рисунок 39):

- загрузить SAML-метаданные подключаемого приложения;
- указать, нужно ли подписывать SAML-атрибуты (SAML Assertions) в ответах Blitz Identity Provider;
- указать, нужно ли шифровать SAML-атрибуты в ответах Blitz Identity Provider;
- указать, нужно ли шифровать SAML-идентификаторы (SAML NameIds) в ответах

Blitz Identity Provider;

- указать, нужно ли включать в ответ перечень утверждений с атрибутами пользователей;
- указать, какие SAML-атрибуты пользователя из Blitz Identity Provider передавать в приложение. SAML-атрибуты должны быть предварительно сконфигурированы в разделе *SAML* консоли управления (описано далее в этом разделе).

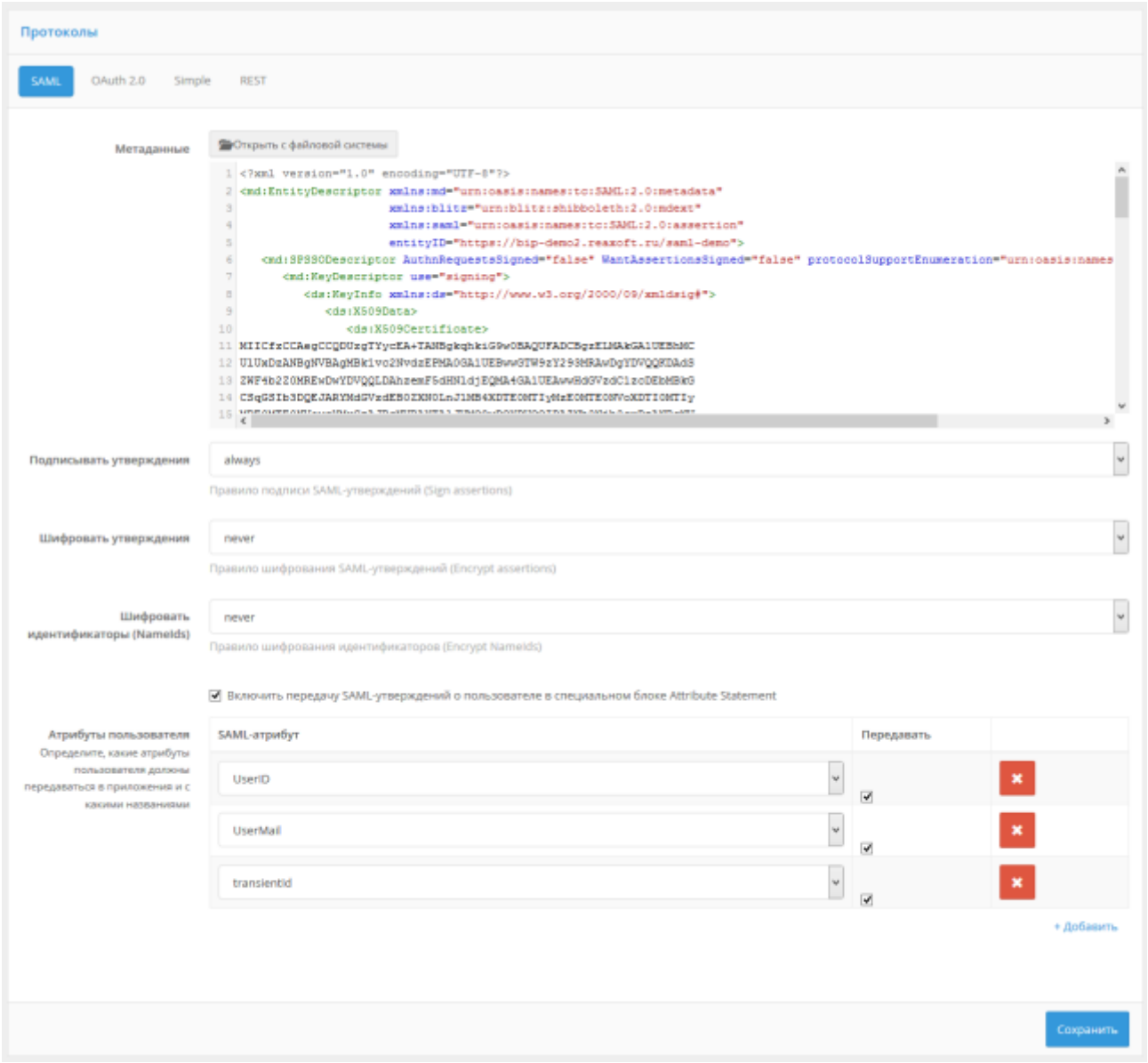

Рисунок 39 – Настройки протокола SAML для приложения

Для регистрации SAML-атрибутов пользователя в Blitz Identity Provider используется раздел *SAML* консоли управления (см. Рисунок 40).

Для добавления нового SAML-атрибута необходимо выполнить следующую последовательность шагов:

1. Нажать на ссылку «+ Добавить новый SAML-атрибут».
2. Ввести:

- название SAML-атрибута (именно оно будет отображаться при подключении SAML-приложений);
- источник атрибута (отображаются все атрибуты, определенные в разделе *Хранилище*);

3. Нажать «Добавить». Атрибут будет добавлен.
4. Определить кодировщики атрибутов. Для этого необходимо:
   - нажать на ссылку «Добавить кодировщик»;
   - выбрать тип кодировщика; следует обратить внимание, что тип кодировщика зависит от версии протокола, с которой работает поставщик услуг (подключенное приложение);
   - название SAML-атрибута, которое будет передано поставщику услуг (в рамках данного типа кодировщика);
   - короткое название, которое будет передано поставщику услуг (в рамках данного типа кодировщика);
   - формат имени.

При необходимости можно определить несколько кодировщиков выбранного SAML-атрибута (для этого каждый кодировщик должен относиться к разным типам кодировщиков).

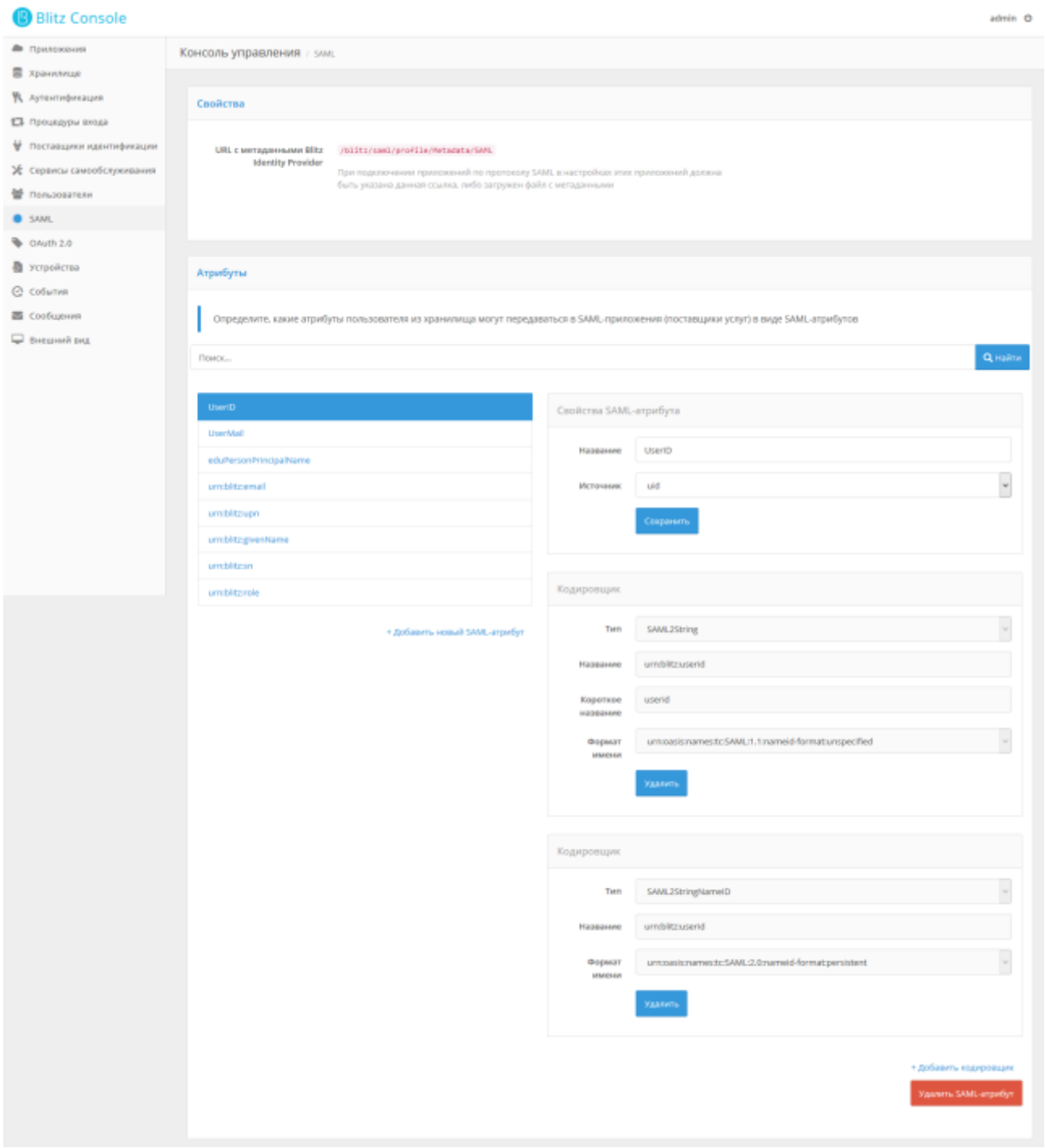

Рисунок 40 – Создание SAML-атрибутов

2.4.3. Настройка OAuth 2.0

При подключении приложения по OAuth 2.0 / OpenID Connect 1.0 необходимо задать следующие настройки (см. Рисунок 41):

- указать секретный ключ подключаемого приложения (client_secret), который должен использоваться подключенным приложением при обращении к Blitz Identity Provider (если не указан, то аутентификация приложения-клиента должна производиться иначе, например, с использованием proxy TLS);

- указать предопределенную ссылку возврата (redirect_uri) – URL, на который по умолчанию будет переадресован пользователь после прохождения авторизации (redirect_uri);
- указать допустимые префиксы ссылок возврата – префикс используется для проверки ссылок возврата (redirect_uri), переданных в запросах на идентификацию от приложений. Если в запросе на аутентификацию указана ссылка возврата и она не соответствует ни одному из указанных префиксов, то в аутентификации будет отказано;
- разрешения по умолчанию – разрешения (scope), которые будут по умолчанию выданы приложению после аутентификации. Если не указаны, то в запросе на аутентификацию всегда должны быть явно прописаны требуемые разрешения;
- отметить при необходимости опцию «Не требовать от пользователя согласие на предоставление доступа к данным о себе». Если она отмечена, то при первом входе пользователя в систему не будет отображена страница согласия на предоставление данных этой системе (рис. 42).

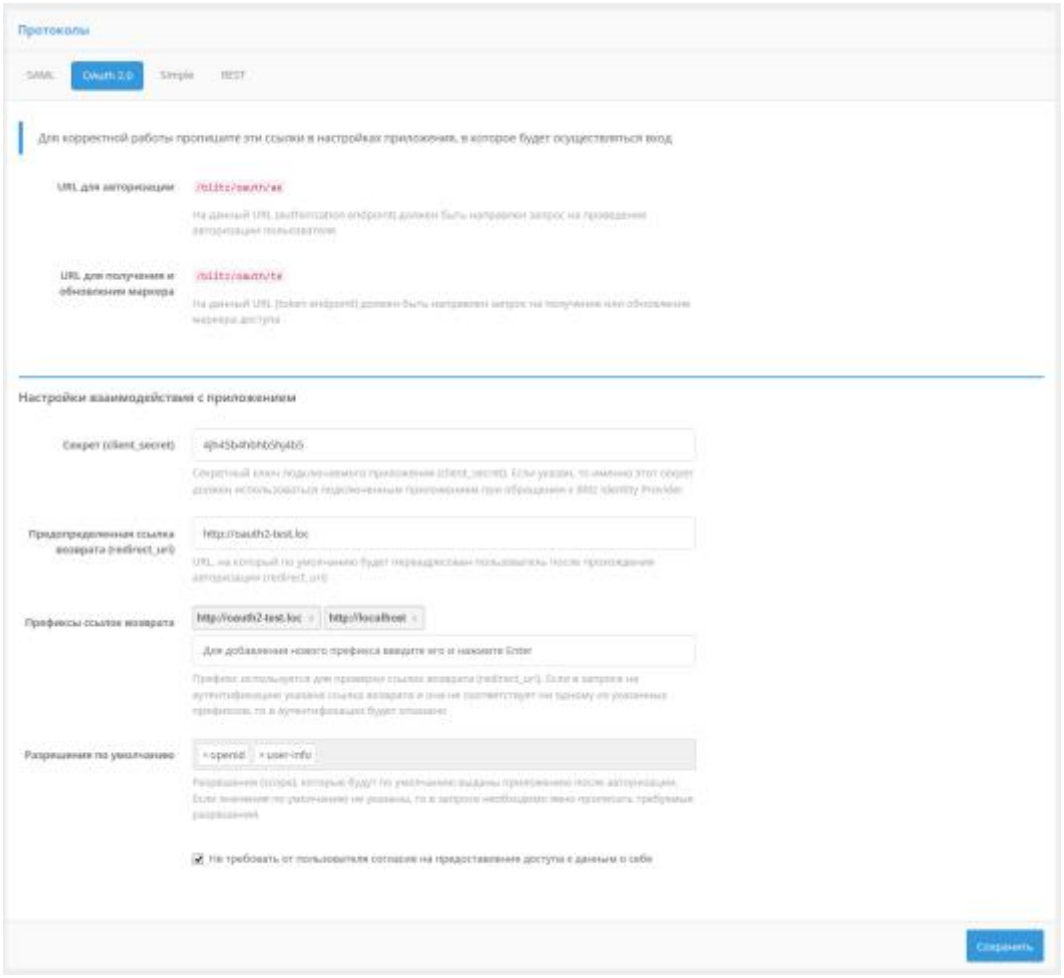

Рисунок 41 – Настройки протокола OAuth 2.0 / OIDC для приложения

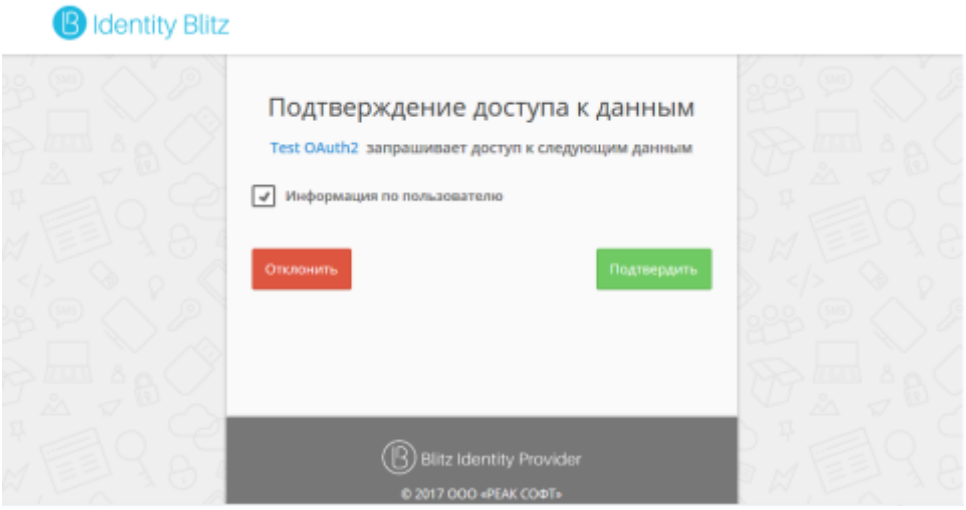

Рисунок 42 – Окно предоставление доступа к данным

Если необходимо проведение аутентификации пользователя по протоколу OIDC (OpenID Connect 1.0), то в качестве одного из разрешений (scope) необходимо указать *openid*. В этом случае в обмен на авторизационный код при вызове Token Endpoint будут выданы не только маркер доступа (access token) и маркер обновления (refresh token), но и идентификационный маркер (ID token).

Для задания общих настроек OAuth 2.0, а также для конфигурирования набора разрешений (scope) используется раздел *OAuth 2.0* консоли управления (см. Рисунок 43).

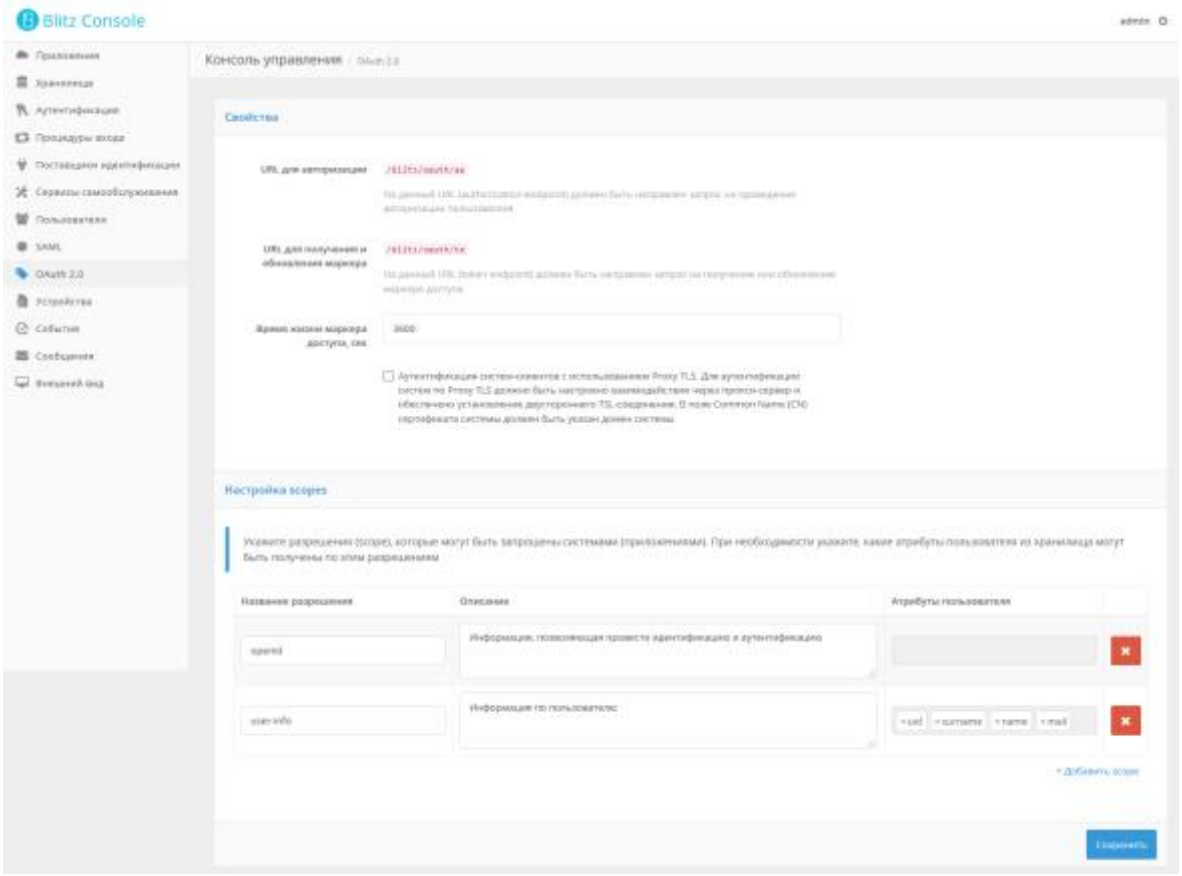

Рисунок 43 – Задание общих настроек OAuth 2.0 / OIDC

В разделе *OAuth 2.0* консоли управления можно посмотреть URL, которые далее потребуются для выполнения запросов:

- OAuth 2.0 Authorization Endpoint – для запроса идентификации и аутентификации и получения авторизационного кода;
- OAuth 2.0 Token Endpoint – для первичного получения маркера доступа / маркера идентификации и для обновления маркера.

При необходимости можно изменить время жизни маркера доступа.

Для корректной работы взаимодействия с приложениями по протоколу OAuth 2.0 необходимо определить разрешения (scope). Для этого нужно указать:

- название разрешения;
- описание разрешения (оно будет отображаться пользователю на странице согласия на предоставление доступа);
- атрибуты пользователя, которые будут предоставлены по данному разрешению (атрибуты должны быть определены в разделе *Хранилище*).

Для корректной работы аутентификации по OpenID Connect 1.0 нужно убедиться, что разрешение с названием *openid* определено в этом разделе консоли. Также можно прописать атрибуты, передаваемые по этому разрешению[14].

2.5. Настройка процедур входа в приложения

2.5.1. Общие сведения

Процедуры входа применяются для настройки правил доступа пользователей к различным приложениям. С помощью процедур можно определить, например, какие приложения должны быть доступны каким пользователям, при каких условиях должна требоваться усиленная аутентификация и какие методы подтверждения входа может применять пользователь. Применение процедур входа позволяет организации исполнить принятые в ней политики контроля доступа к приложениям.

Управление процедурами входа осуществляется в разделе *Процедуры входа* консоли управления Blitz Identity Provider (см. Рисунок 44). Эта функциональность доступна только в редакции Enterprise Edition.

[14] В этом случае указанные данные могут быть получены по маркеру доступа (access token), выданному на разрешение openid.

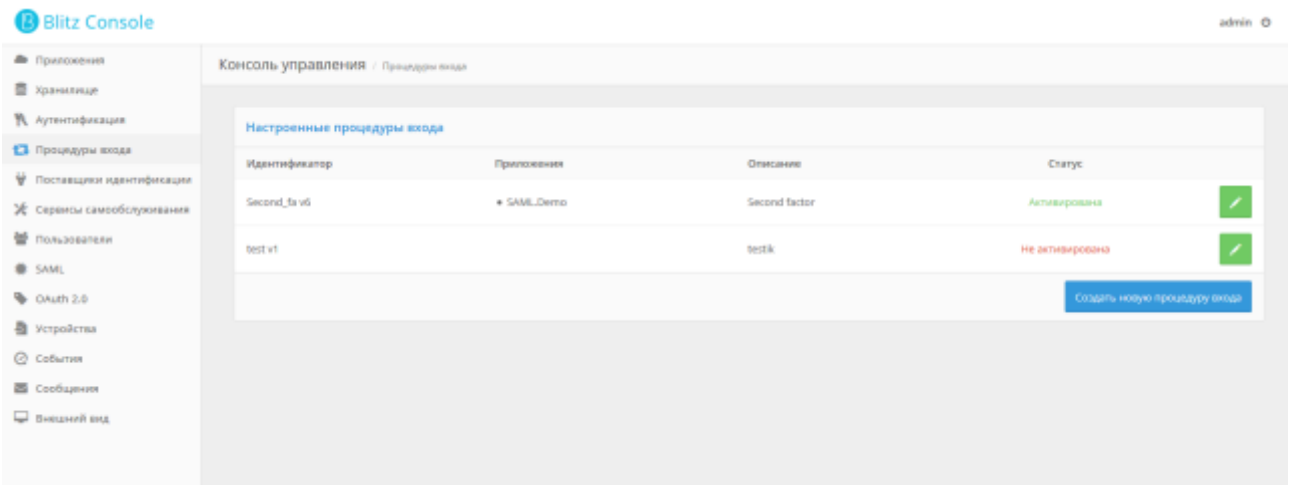

Рисунок 44 – Экран настроек процедур входа

Создание процедуры входа включает в себя следующие шаги:

1. Указание базовых параметров процедуры:
   - идентификатор процесса (процедуры);
   - описание процедуры;
   - приложения – перечень приложений, для которых будет применяться данная процедура.

Для каждого приложения может быть создана только одна процедура. Если для данного приложения не создано процедуры, к нему будет применяться стандартная процедура входа (процедура входа по умолчанию)[15]. Если процедура создана без указания приложений, то она заменит стандартную процедуру входа.

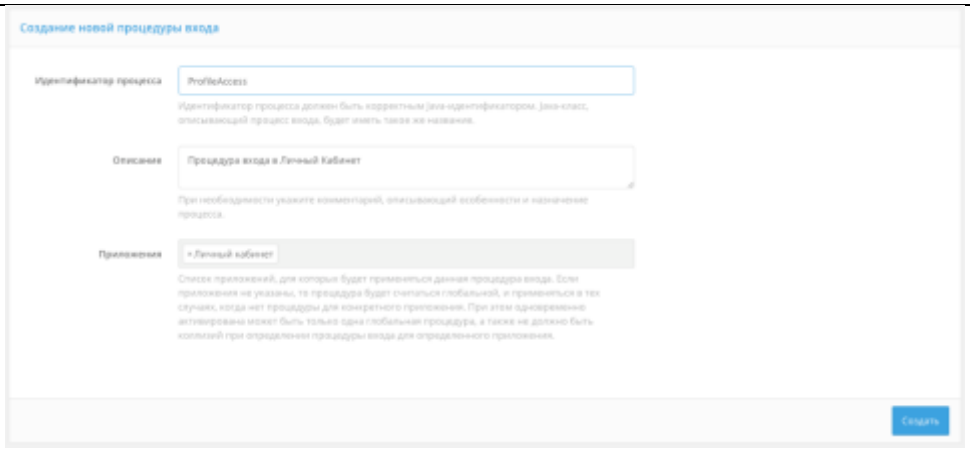

Рисунок 45 – Экран создания новой процедуры входа

2. Написание исходного кода процедуры (см. Рисунок 46). Для успешной работы процедуры входа необходимо написать на языке Java класс, реализующий необходимый интерфейс Strategy. Вся контекстная информация о пользователе, о текущем состоянии процедуры аутентификации и т.д. доступна в объекте Context.

[15] При создании новой процедуры в окне будет отображена стандартная процедура.

Процедура состоит из двух блоков, которые определяют:

- действия, предпринимаемые на начальном этапе процесса аутентификации. В этом блоке, например, можно определить, при каких условиях осуществлять переход в приложение в режиме SSO (если пользователь ранее был аутентифицирован);
- действия, предпринимаемые после первичной аутентификации пользователя. В этом блоке, например, можно определить, какие методы усиленной аутентификации при каких условиях использовать.

Чтобы было удобнее вести разработку процедуры, Blitz Identity Provider позволяет загрузить из консоли управления JAR-файл для использования разработчиками в IDE среде разработки. Для этого нужно нажать кнопку «Загрузить Blitz Development Kit».

3. После написания кода необходимо нажать на кнопку «Компилировать». При наличии ошибок некорректные фрагменты кода будут выделены цветом и подписаны ошибки.
4. Если компиляция прошла успешно, можно активировать процедуру (кнопка «Активировать» в шапке соответствующей процедуры).
5. Если необходимо отредактировать процедуру, ее необходимо сначала деактивировать.

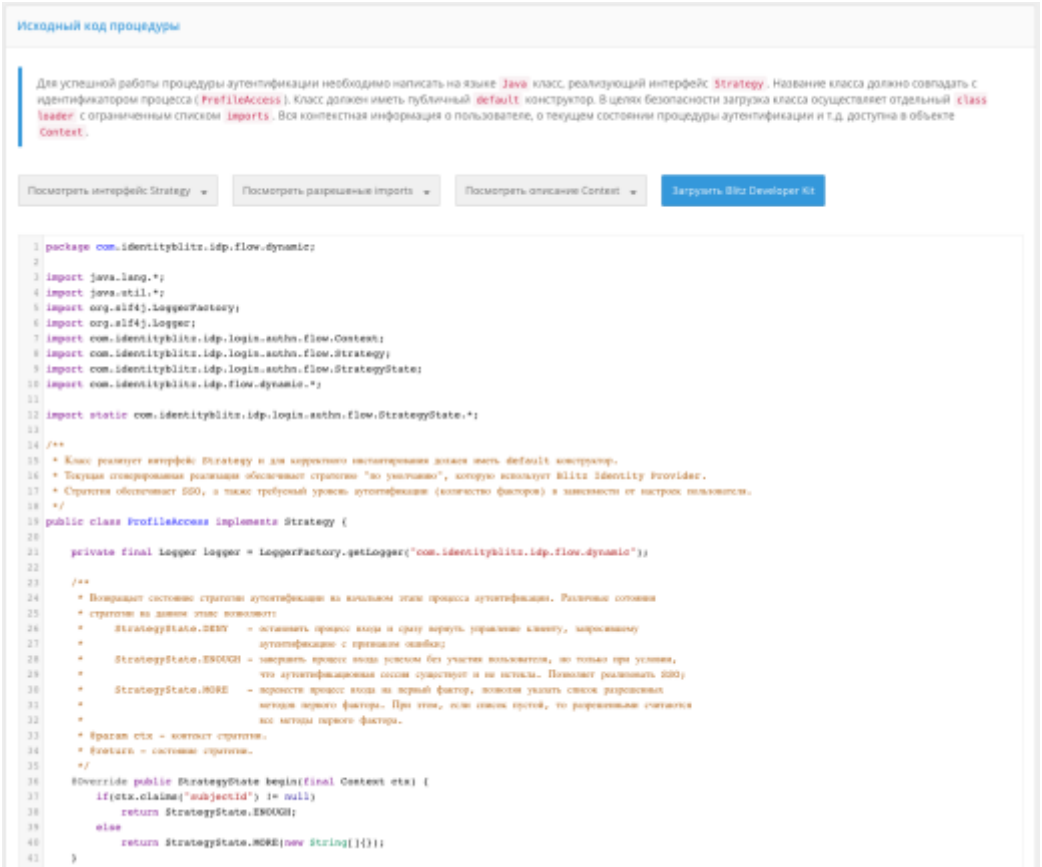

Рисунок 46 – Экран редактирования исходного кода процедуры входа (фрагмент)

2.5.2. Примеры процедур входа

Далее приводятся примеры некоторых процедур. Для удобства отладки можно выводить информацию о состоянии аутентификации в лог, воспользовавшись функцией *logger.debug()*. Например, следующая команда выведет в лог заданный уровень аутентификации для пользователя:

```
logger.debug("requiredFactor="+ctx.userProps("requiredFactor"));
```

2.5.2.1. Разрешить вход только при определенном значении некоторого атрибута у пользователя

Следующая процедура допускает пользователя в приложение при условии, что адрес его электронной почты равен ivanov@company.ru.

```
    @Override public StrategyState begin(final Context ctx) {
     if(ctx.claims("subjectId") != null){

// если пользователь уже аутентифицирован, то проверяем его email

          if("ivanov@company.ru".equals(ctx.claims("mail")))
          return StrategyState.ENOUGH;
                    else
                    return StrategyState.DENY;
     }

// если не аутентифицирован, то просим его пройти первый фактор аутентификации

       else
          return StrategyState.MORE(new String[]{});
    }


    @Override public StrategyState next(final Context ctx) {

// после первичной аутентификации проверяем его email, если он верный, то просим анализируем параметр requiredFactor
пользователя (требуемый уровень аутентификации) и в зависимости от этого просим пройти второй фактор

     if(!"ivanov@reaxoft.ru".equals(ctx.claims("mail")))
       return StrategyState.DENY;
     String reqFactor = ctx.userProps("requiredFactor");
          if(reqFactor == null)
             return StrategyState.ENOUGH;
          else {
             if(Integer.valueOf(reqFactor) < ctx.justCompletedFactor())
                return StrategyState.ENOUGH;
             else
                return StrategyState.MORE(new String[]{});
          }
      }
```

2.5.2.2. Принудительная двухфакторная аутентификация в приложение

Следующая процедура требует двухфакторной аутентификации для доступа к приложению. Если пользователь переходит в приложение в рамках единой сессии, то при наличии одного пройденного фактора у него будет дополнительно проверен второй фактор.

```
    @Override public StrategyState begin(final Context ctx) {
      if(ctx.claims("subjectId") != null){

// если пользователь уже аутентифицирован, то проверяем количество пройденных факторов

       if (ctx.sessionTrack().split(",").length < 2)
                    return StrategyState.MORE(new String[]{});
```

```
        else
            return StrategyState.ENOUGH;
                    }

// если он не аутентифицирован или факторов меньше двух, то требуем прохождения следующего уровня аутентификации

        else
            return StrategyState.MORE(new String[]{});
    }

  @Override public StrategyState next(final Context ctx) {

// если пользователь прошел один фактор аутентификации, то требуем пройти второй; если больше одного – допускаем в приложение

   if(ctx.justCompletedFactor() == 1)
       return StrategyState.MORE(new String[]{});
     else
       return StrategyState.ENOUGH;
   }
```

2.5.2.3. Принудительный вход в приложение

Если требуется, чтобы пользователь всегда входил в приложение, т.е. чтобы не работал переход в рамках единой сессии, то может использоваться такая процедура:

```
    @Override public StrategyState begin(final Context ctx) {

// независимо от наличия сессии требуем прохождения первого фактора

          return StrategyState.MORE(new String[]{});
    }

       @Override public StrategyState next(final Context ctx) {

// анализируем параметр requiredFactor пользователя (требуемый уровень аутентификации) и в зависимости от этого просим пройти второй фактор

        String reqFactor = ctx.userProps("requiredFactor");
           if(reqFactor == null)
              return StrategyState.ENOUGH;
           else {
              if(Integer.valueOf(reqFactor) == ctx.justCompletedFactor())
                 return StrategyState.ENOUGH;
              else
                 return StrategyState.MORE(new String[]{});
           }
        }
```

2.5.2.4. Принудительный вход в приложение, если с момента входа прошло более 60 секунд

Данная процедура устанавливает требование к продолжительности сессии: если пользователь был аутентифицирован более X секунд назад (в примере – более 60 секунд назад), то требуется аутентификация для входа в приложение, т.е. переход в рамках единой сессии невозможен. Для этого процедура анализирует следующие параметры сессии:

- sessionCreated – время создания сессии;
- sessionUpdated – время изменения сессионных данных.

До тех пор, пока хотя бы один из этих параметров имеет значение менее 60 секунд, повторная аутентификация в приложение не требуется. Пример процедуры:

```
  @Override public StrategyState begin(final Context ctx) {

       if(ctx.claims("subjectId") != null) {

// Проверяем, на сколько отличается значение параметров sessionCreated и sessionUpdated от текущего времени; если хотя
```

```
бы один из них менее 60000 мс, то этого достаточно для входа в приложение

        if((System.currentTimeMillis() - ctx.sessionCreated().getTime()) < 60000 ||
           (ctx.sessionUpdated() != null && System.currentTimeMillis() - ctx.sessionUpdated().getTime() < 60000))
          return StrategyState.ENOUGH;
        }
        return StrategyState.MORE(new String[]{});
    }

 @Override public StrategyState next(final Context ctx) {
    String reqFactor = ctx.userProps("requiredFactor");
     if(reqFactor == null)
          return StrategyState.ENOUGH;
        else {
          if(Integer.valueOf(reqFactor) == ctx.justCompletedFactor())
             return StrategyState.ENOUGH;
          else
             return StrategyState.MORE(new String[]{});
        }
    }
```

2.6. Настройка сервисов самообслуживания пользователей

Blitz Identity Provider включает в себя ряд сервисов (веб-приложений), с помощью которых пользователи самостоятельно могут выполнять ряд операций:

1. Личный кабинет – позволяет выполнить ряд операций с учетной записью, например, посмотреть/изменить свои данные, настроить способы аутентификации, посмотреть последние события, сменить пароль. Если включен, то доступен по адресу: {hostname}/blitz/profile.
2. Регистрация пользователей. Если включен, то пользователи смогут перейти со страницы входа (ссылка «Нет аккаунта? Зарегистрироваться») на форму самостоятельной регистрации.
3. Восстановление доступа – позволяет пользователю сменить пароль от своей учетной записи. Если сервис включен, то пользователи смогут перейти со страницы входа (ссылка «Забыли пароль?») на форму восстановления доступа.

Настройка данных сервисов осуществляется в разделе *Сервисы самообслуживания* консоли управления.

2.6.1. Общие настройки

На главной странице раздела *Сервисы самообслуживания* можно включить/выключить соответствующие сервисы, используя переключатель (). Следует при этом учесть, что переключатель включает/выключает сервисы только в Blitz Identity Provider Standard Edition. Для Enterprise Edition он влияет на отображение ссылок (например, «Забыли пароль?»), тогда как наличие самого сервиса зависит от того, было ли соответствующее приложение установлено администратором.

Кроме того, на главной странице можно настроить правила проверки атрибутов. Данные правила проверки применяются при изменении/добавлении атрибутов через сервисы самообслуживания и через API.

Каждое правило включает в себя:

- *атрибут*, добавление/изменение которого регулируется правилом;
- *тип подтверждения* – способ подтверждения изменения атрибута. Если он не задан, то смена атрибута производится сразу. Если по адресу электронной почты – то пользователь должен воспользоваться ссылкой, отправленной на электронную почту, либо ввести код из этого письма. Если по мобильному телефону, то пользователь должен ввести код из SMS-сообщения для того, чтобы изменение произошло.
- *атрибут подтверждения* – используется только в тех случаях, когда настроено подтверждение атрибута. Здесь с помощью строк подстановки необходимо указать адрес, на который отправлять сообщение. Например, при вводе ${attr_name} подтверждение будет отправлено на телефон / электронную почту, уже сохраненную в базе (attr_name – имя атрибута, заданное в *Хранилище*). Если подтверждение должно быть отправлено на новый контакт, введенный пользователем при изменении атрибута, то следует указать ${input}.
- *правило проверки*, задаваемое с помощью регулярных выражений[16]. Например, правило ^[0-9]{11}$ проверяет, что атрибут состоит из 11 символов и используются исключительно цифры. Правило ^(.+)@(.+)$ проверяет, что введенный атрибут содержит знак @.
- *обязательность*, т.е. должен ли этот атрибут обязательно быть задан у пользователя.

[16] См., например: http://www.regular-expressions.info/posixbrackets.html?wlr=1

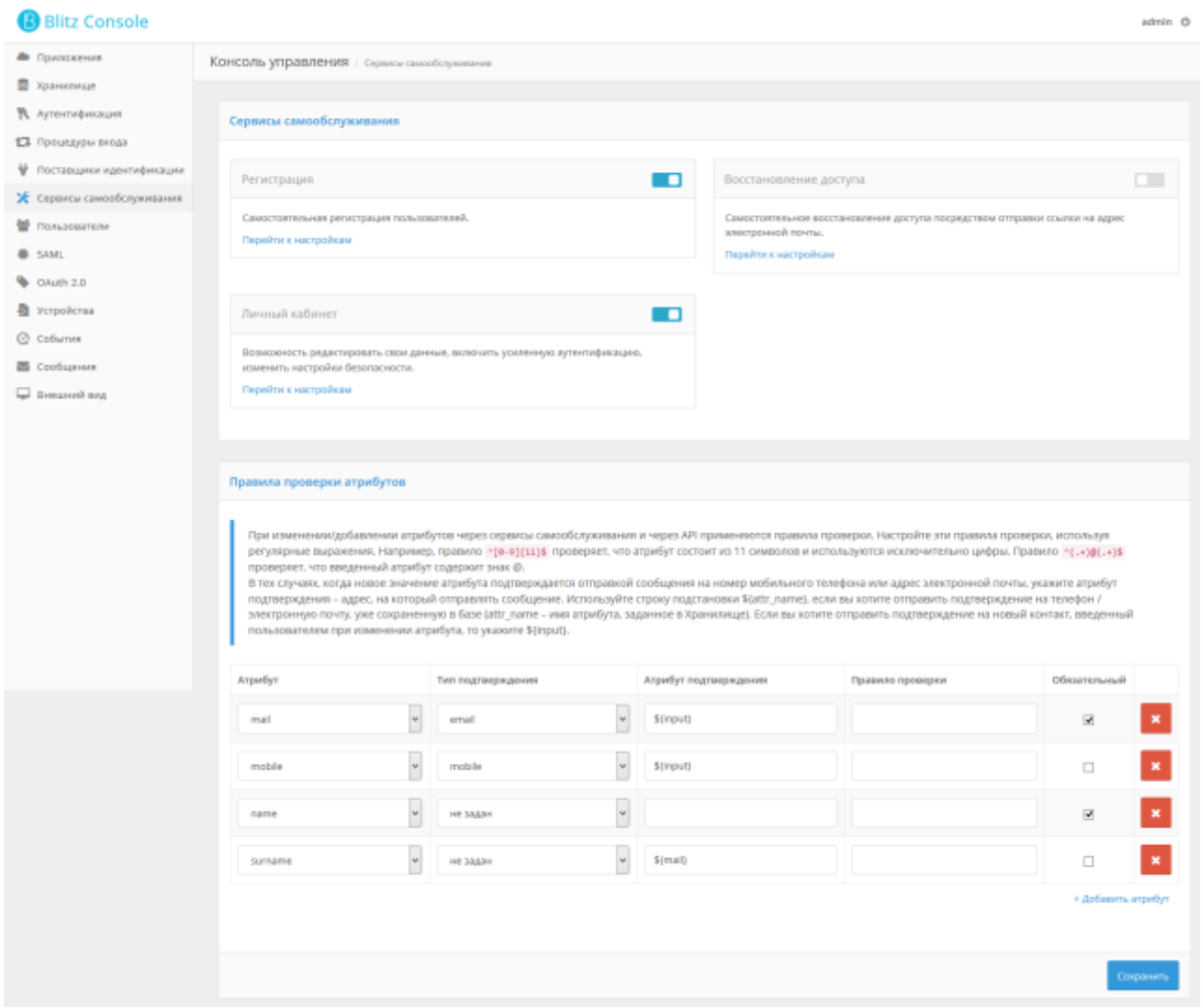

Рисунок 47 – Сервисы самообслуживания и их общие настройки

В подразделах осуществляется настройка каждого сервиса самообслуживания в отдельности.

2.6.2. Личный кабинет

Личный кабинет – веб-приложение, в котором пользователь может выполнить следующие действия:

- посмотреть или изменить данные своей учетной записи;
- посмотреть и настроить способы подтверждения входа (усиленной аутентификации);
- посмотреть последние события безопасности (например, события входа);
- сменить пароль;
- посмотреть привязанные учетные записи социальных сетей и привязать новые «внешние» учетные записи.

Настройка личного кабинета включает в себя конфигурирование способа отображения

атрибутов пользователя и изменение дополнительных параметров.

2.6.2.1. Отображение атрибутов пользователя

На основной странице Личного кабинета отображается блок с данными учетной записи. Пример этого блока представлен на рис. 48.

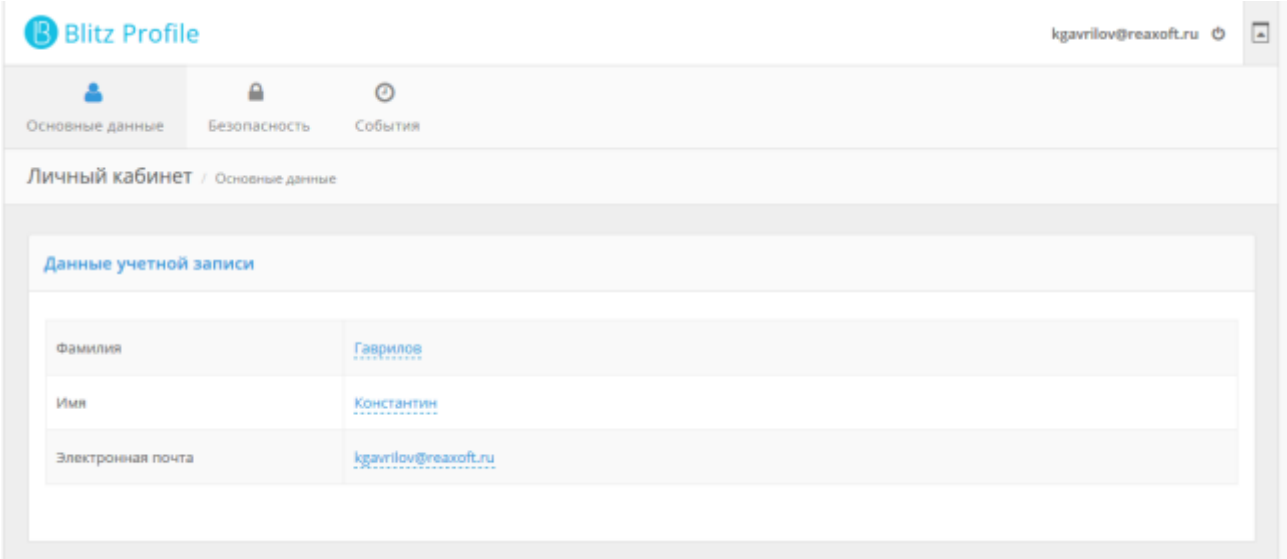

Рисунок 48 – Личный кабинет: данные учетной записи

Отображение данных пользователя определяется HTML-шаблоном. Шаблон представляет собой текстовый файл, который компилируется с помощью шаблонизатора Twirl[17]. В шаблоне необходимо разместить функции, позволяющие пользователю в Личном кабинете вводить и редактировать данные о себе.

В шаблоне доступны следующие функции:

- *@show(attrName)* – отображает значение атрибута;
- *@showStrings(attrName, values)* – отображает значение массива;
- *@editAsText(attrName, readableName, errorMsg)* – отображает значение атрибута и позволяет его изменить (параметр errorMsg необязательный);
- *@editAsBoolean(attrName, readableName)* – отображает значение логического типа (true/false) атрибута и позволяет его изменить;
- *@editAsStrings(attrName, readableName, values)* – отображает значение (массив) атрибута и позволяет его изменить.

В этих функциях используются следующие параметры:

- *attrName* – название атрибута, определенное в *Хранилище*;
- *readableName* – отображаемое в письме пользователю имя атрибута (можно

[17] https://github.com/playframework/twirl

задать как идентификатор из файла сообщений или как текст);

- *values* – значения, представляющие собой формат «ключ – описание», где ключ – значение массива, описание – читаемое значение ключа (например, ListMap("a" -> "значение a", "c" -> "значение c")), может задаваться как идентификатор из файла сообщений или как текст;
- *errorMsg* – описание ошибки, которое отображается в случае ошибочного ввода значения (можно задать как идентификатор из файла сообщений или как текст). Про файлы сообщений см. раздел 2.13.2.2 документа. Рекомендуется использовать файлы сообщений при необходимости поддержки мультиязычности.

Примеры функций:

Отображение атрибута mail:

```
@editAsText("mail", "Электронная почта")
```

Отображение атрибута mobile с возможностью его редактировать:

```
@editAsText("mobile", "Мобильный телефон", "Ошибка")
```

Отображение булевого атрибута *info* с возможностью его редактировать:

```
@editAsBoolean("info", "Подписка")
```

Отображение массива строк *massiv* с возможностью его редактировать (выбор значений):

```
@editAsStrings("massiv", "Подписки", ListMap("a" -> "Акции и бонусные программы", "b" -> "Новости компании", "c" -> "Дайджест событий за месяц"))
```

Пример отображения массива строк в интерфейсе Личного кабинета представлен на рис. 49.

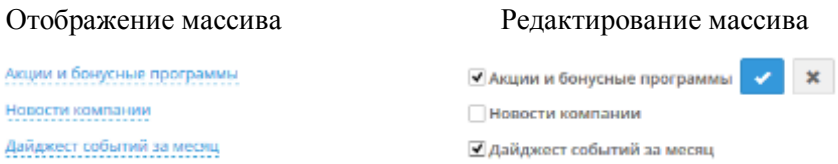

Рисунок 49 – Личный кабинет: массив строк (в режиме отображения и редактирования)

2.6.2.2. Дополнительные параметры

В качестве дополнительных параметров можно задать:

- шаблон приветствия – информацию, которая отображается в правом верхнем углу Личного кабинета. Допустимо использовать строки подстановки. Например, “${surname} ${surname}” позволит отобразить фамилию и имя пользователя;
- настройки подтверждения атрибутов, в которые входят следующие параметры:
  - время действия разового SMS-кода подтверждения (в секундах);
  - длина разового SMS-кода подтверждения;

- длина кода подтверждения, отправляемого по электронной почте.

– доступные пользователям функции, т.е. функции, которые могут быть использованы из Личного кабинета. Возможно включить/выключить следующие функции:
  - смена пароля;
  - просмотр и привязка социальных сетей;
  - привязка HOTP-генераторов;
  - привязка TOTP-генераторов;
  - настройка подтверждения входа по SMS-коду.

2.6.3. Регистрация пользователей

Регистрация пользователей – веб-приложение, позволяющее пользователю самостоятельно создать свою учетную запись. Настройка регистрации включает в себя конфигурирование формы регистрации и изменение дополнительных параметров.

2.6.3.1. Форма регистрации

Пример простейшей формы регистрации на рис. 50. На данном рисунке красным выделен блок с перечнем данных пользователя, который может быть настроен.

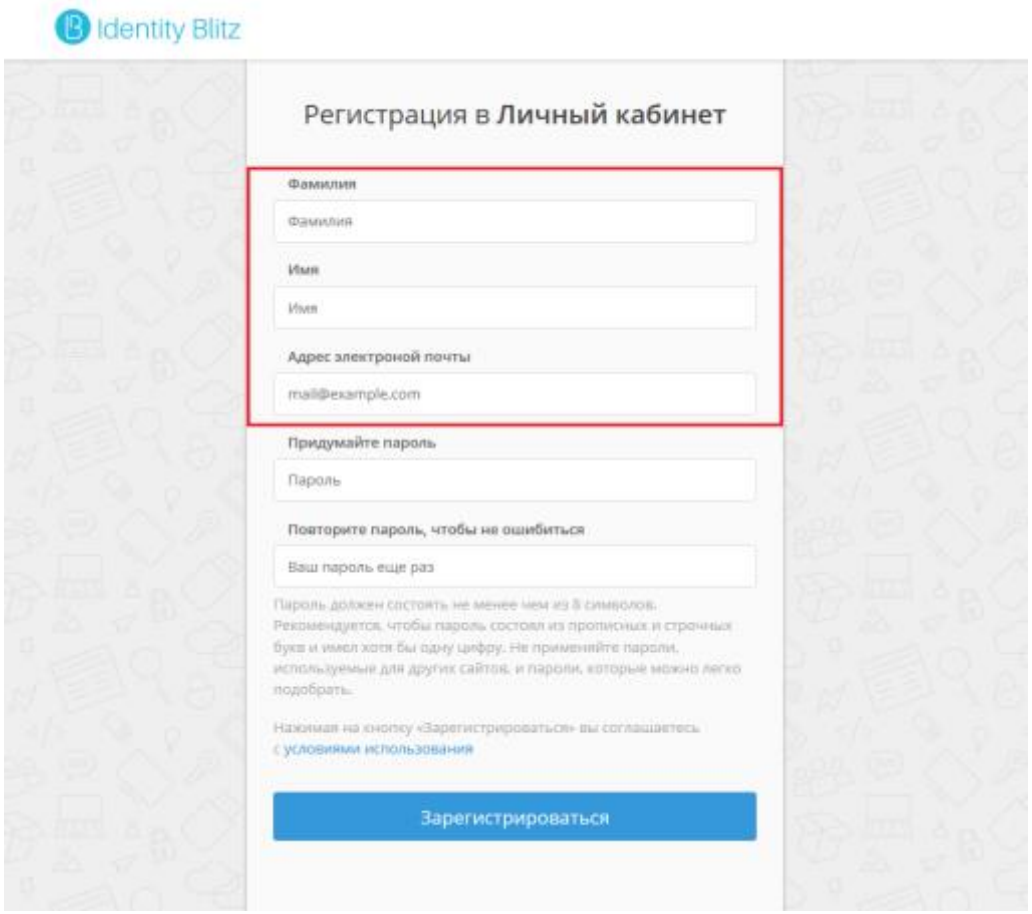

Рисунок 50 – Форма регистрации

Перечень запрашиваемых данных пользователя определяется HTML-шаблоном. Шаблон представляет собой текстовый файл, который компилируется с помощью шаблонизатора Twirl. В шаблоне необходимо разместить функции, позволяющие пользователю при регистрации вводить данные о себе.

В шаблоне доступны следующие функции:

- *@editAsText(label, attrName, Map("placeholder" -> "placeholderLabel", "error-message" -> "errMsgLabel", "help-block" -> "helpMsgLabel"))* – отображает текстовое поле для определения атрибута (последний параметр не обязательный);
- *@editAsStrings(attrName, values, Map("help-block" -> "helpMsgLabel"))* – отображает чекбоксы для определения значения атрибута типа массив (последний параметр не обязательный);
- *@editAsBoolean(attrName, Map("help-block" -> "helpMsgLabel"))* – отображает чекбокс для определения значения булевого атрибута (последний параметр необязательный).

В этих функциях используются следующие параметры:

- *label* – название атрибута, отображаемое пользователю; можно задать как идентификатор из файла сообщений или как текст;
- *attrName* – название атрибута, определенное в хранилище;
- *values* – значения, представляющие собой формат "ключ - описание", где ключ - значение массива, описание - читаемое значение ключа (например, ListMap("a" -> "значение a", "c" -> "значение c")), может задаваться как идентификатор из файла сообщений или как текст;
- *placeholder* – подсказка в поле для ввода атрибута (можно задать как идентификатор из файла сообщений или как текст);
- *error-message* – сообщение об ошибке при неправильно введенном значении атрибута (можно задать как идентификатор из файла сообщений или как текст);
- *help-block* – подсказка под элементом/элементами (можно задать как идентификатор из файла сообщений или как текст).

Примеры функций:

Ввести в атрибут mail адрес электронной почты:

```
@editAsText("Адрес электроной почты", "mail", Map("placeholder" -> "mail@example.com", "error-message" -> "Обязательное поле"))
```

Ввести данные булевого атрибута info:

```
@editAsBoolean("Хочу получать рассылку", "info")
```

Выбрать предложенные варианты из массива (атрибут massiv):

```
@editAsStrings("Виды рассылок:", "massiv", ListMap("a" -> "Акции и бонусные программы", "b" -> "Новости компании", "c" -> "Дайджест событий за месяц"))
```

Пример отображения булевого атрибута массива строк в интерфейсе Регистрации представлен на рис. 51.

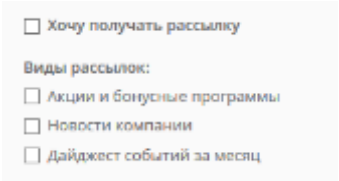

Рисунок 51 – Пример ввода при регистрации булевого атрибута и массива

2.6.3.2. Дополнительные параметры

В качестве дополнительных параметров можно задать:

- параметры сервиса регистрации, в которые входят:
  - выбор хранилища для учетной записи – нужно выбрать одно из сконфигурированных хранилищ (раздел Хранилище) для сохранения учетной записи;
  - способ подтверждения – учетные записи могут быть созданы сразу после заполнения формы регистрации («Не требовать активации»), либо только после подтверждения владения адресом электронной почты («По электронной почте») – в этом случае пользователь должен ввести код из письма или пройти по ссылке;
  - атрибут электронной почты – атрибут с адресом электронной подписи, на который должен быть отправлено письмо с подтверждением (если в качестве способа подтверждения выбрано «По электронной почте»).
- правила связывания атрибутов со значениями других атрибутов. В том случае, если необходимо сохранить введенные пользователем значения в другие атрибуты (те, которые не указаны в HTML-шаблоне), то следует указать эти атрибуты и настроить правила, по которым они будут задаваться. Например, правило uid=${mail} означает, что атрибуту uid в Хранилище будет присвоено значение атрибута mail, заданного пользователем при регистрации.

Скриншот фрагмента страницы настроек регистрации (в части дополнительных настроек) представлен на рис. 52.

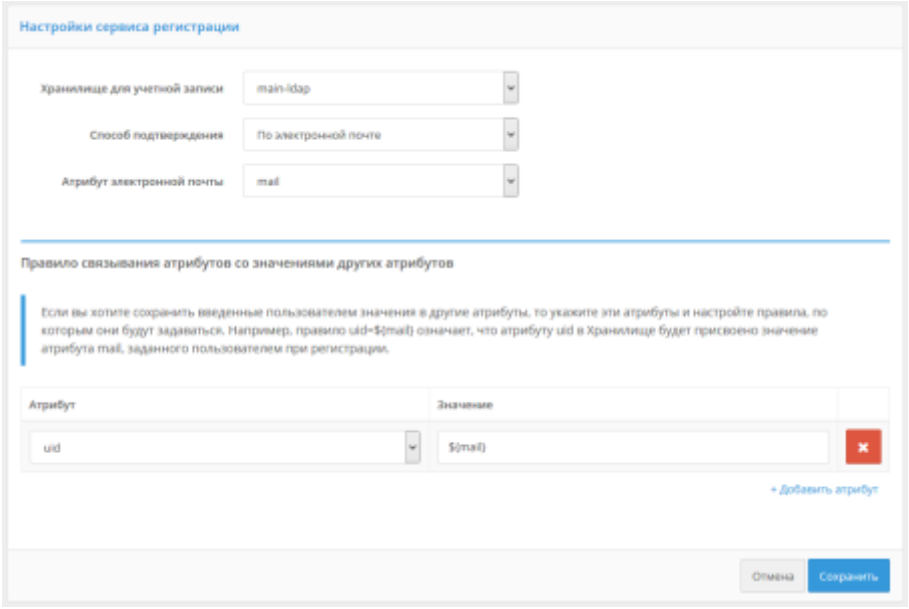

Рисунок 52 – Скриншот настроек регистрации (фрагмент)

2.6.4. Восстановление доступа

Настройка сервиса восстановления доступа включает в себя указание атрибута, в котором хранится адрес электронной почты пользователя (рис. 53). На этот адрес будет отправлено письмо, содержащее ссылку для восстановления доступа к учетной записи.

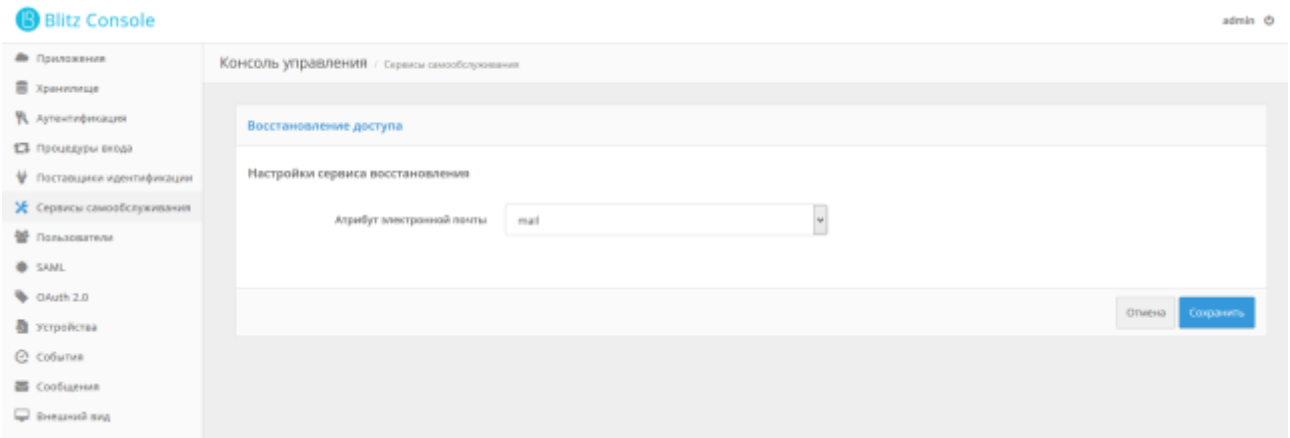

Рисунок 53 – Восстановление доступа

2.7. Вход через внешние поставщики идентификации

Настройка входа через внешние поставщики идентификации включает в себя следующие шаги:

1. Сконфигурировать конкретного поставщика идентификации в разделе «Поставщики идентификации» на стороне Blitz Identity Provider.
2. Сконфигурировать этого поставщика идентификации на стороне самого поставщика идентификации.
3. Включить возможность входа через данный поставщик идентификации в разделе

«Аутентификации» (см. раздел 2.3.4).

Для настройки используется раздел *Поставщики идентификации* в консоли управления. Начальный экран показывает уже настроенные поставщики идентификации и позволяет выбрать для настройки требуемый тип поставщика идентификации (см. Рисунок 54). Настройки поставщиков каждого из типов описаны далее в подразделах.

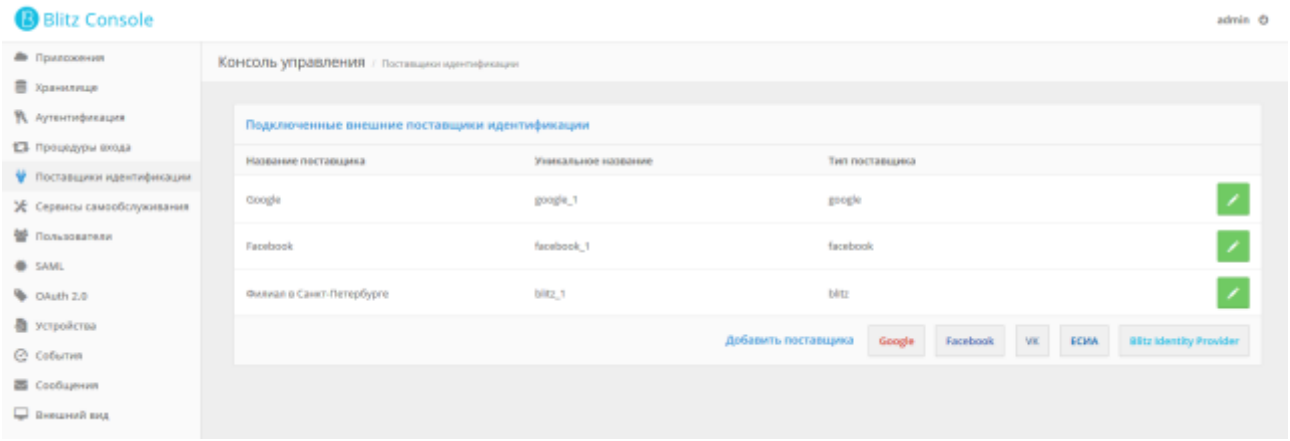

Рисунок 54 – Вид экрана настройки поставщиков идентификации

2.7.1. Вход через Google

Для конфигурирования входа через учетную запись Google следует выполнить следующие шаги в разделе *Поставщики идентификации* консоли управления:

1. Добавить поставщика, имеющего тип *Google*.
2. Ввести идентификатор поставщика (можно не менять предложенный системой идентификатор).
3. Ввести название поставщика. Именно это название будет отображаться на странице входа Blitz Identity Provider.

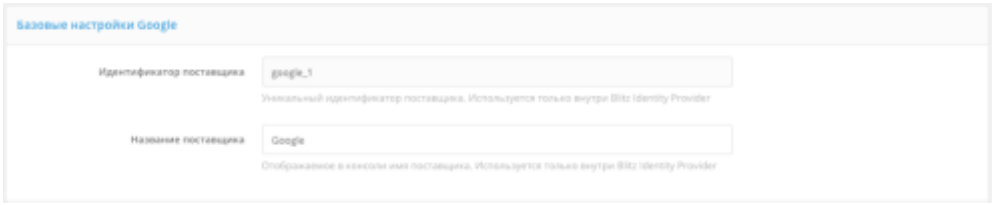

Рисунок 55 – Базовые настройки поставщика идентификации Google

4. Перейти в Диспетчер API Google (см. Рисунок 56)[18], в котором выполнить следующие операции:
   - перейти в раздел «Учетные данные»;
   - создать проект и создать новые учетные данные типа «Идентификатор клиента OAuth»;
   - выбрать тип нового идентификатора клиента (например, веб-приложение) и

[18] https://console.developers.google.com

дать ему название;

- ограничения не задавать, они будут указаны позже;
- Google сгенерирует идентификатор и секрет клиента, они потребуются для последующего ввода в Blitz Identity Provider.

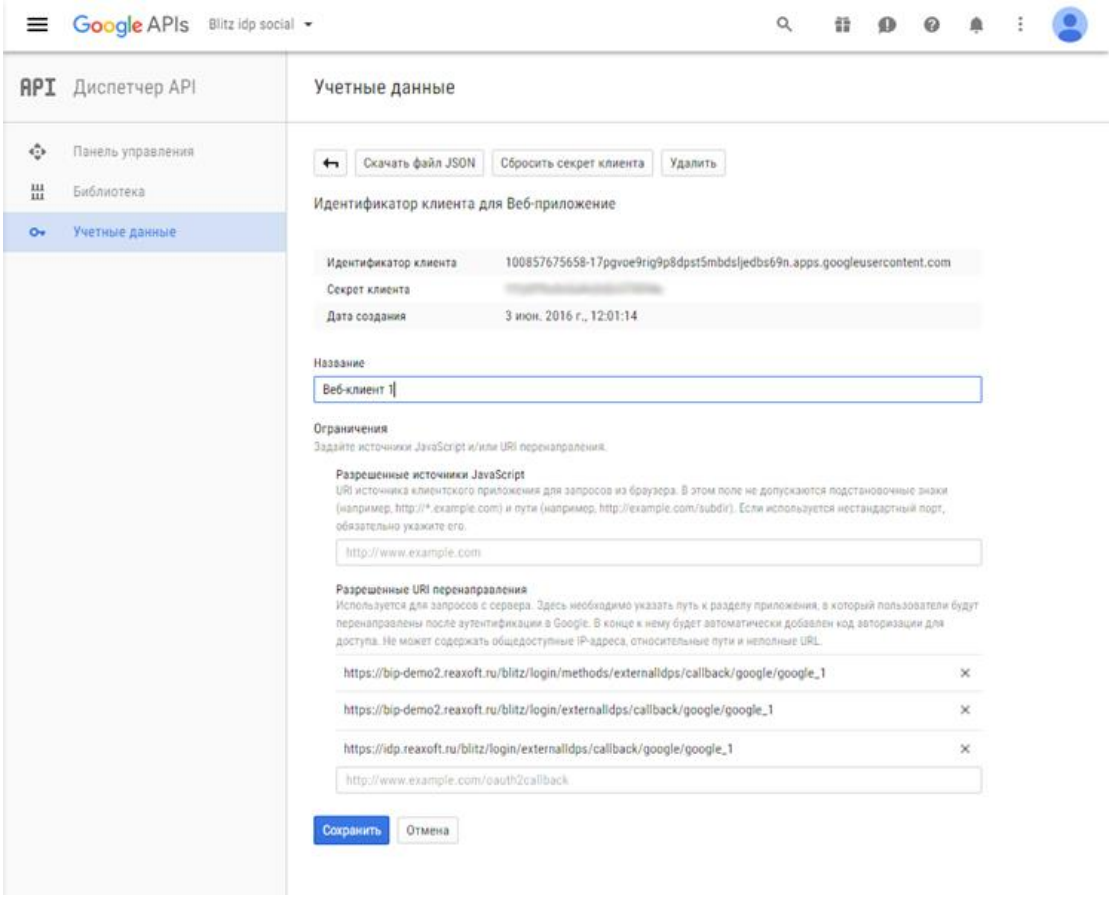

Рисунок 56 – Настройки в Диспетчере API Google

5. Перейти в Blitz Identity Provider и заполнить дополнительные настройки поставщика идентификации (см. Рисунок 57), которые включают в себя:
   - идентификатор клиента (Client ID), полученный в Диспетчере API Google;
   - секрет клиента (Client secret), полученный в Диспетчере API Google;
   - запрашиваемые разрешения (scope), предусмотренные в Google[19];
   - атрибут Google, который будет использоваться для идентификации учетной записи в Blitz Identity Provider. Этот атрибут по смыслу соответствует базовому идентификатору, определяемому в разделе *Хранилище*;
   - правила соответствия (маппинга) между атрибутами, определенными в Blitz Identity Provider, и атрибутами Google. Например, правило mail=${email} означает, что атрибут с именем *mail* в Blitz Identity Provider будет заполняться значением из атрибута *email* учетной записи Google (для пользователей,

[19] См.: https://developers.google.com/+/web/api/rest/oauth#authorization-scopes

воспользовавшихся этим поставщиком идентификации);

- проставить галочку у тех атрибутов, по которым должен осуществляться поиск (в разделе *Пользователи*);

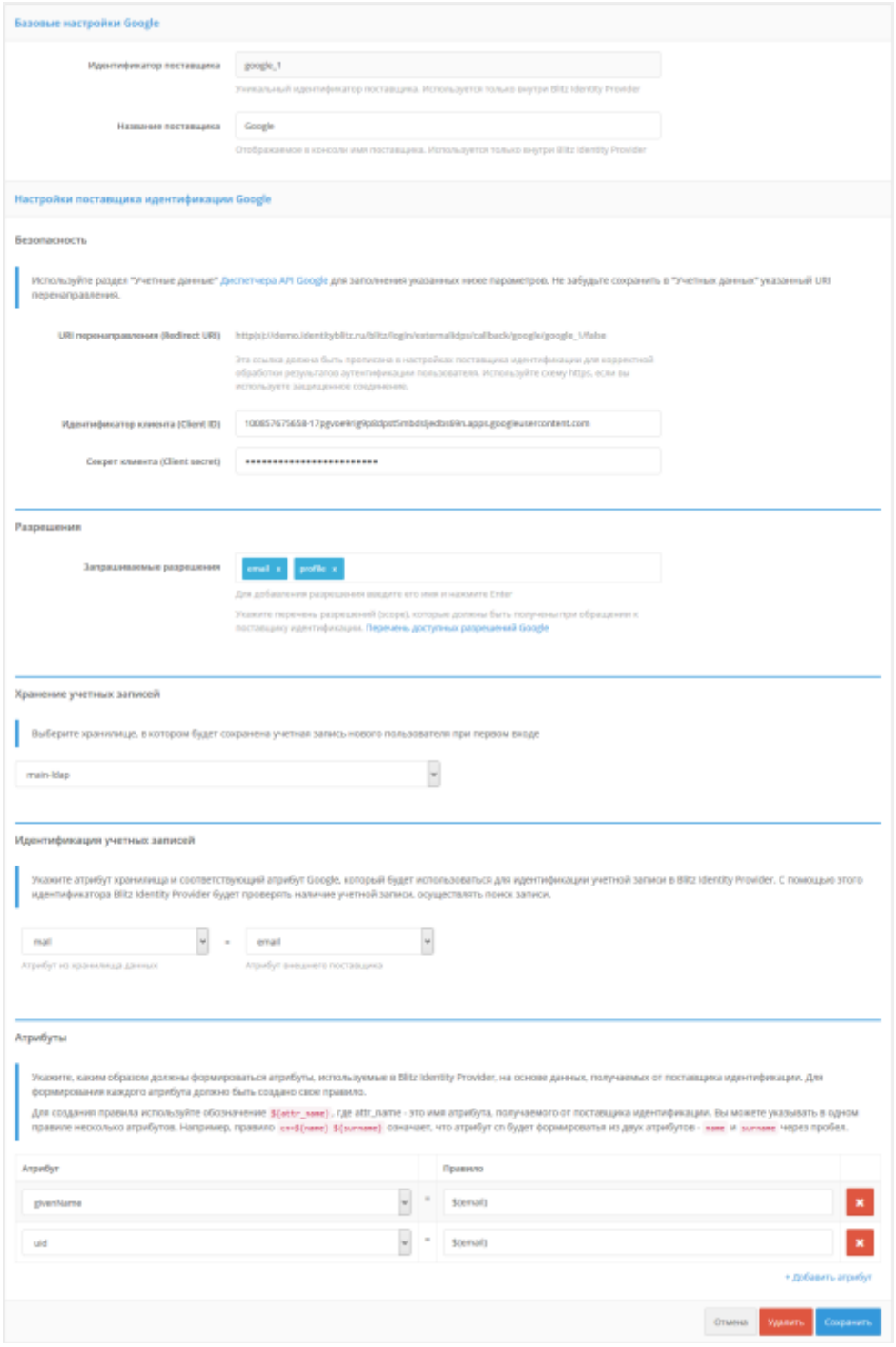

Рисунок 57 – Дополнительные настройки поставщика идентификации Google

6. Перейти в Диспетчер API Google и указать в качестве разрешенного URI перенаправления значение, указанное в параметре «URI перенаправления (Redirect URI)» консоли управления.

7. Сохранить данные и в Blitz Identity Provider, и в Диспетчере API Google.

2.7.2. Вход через Facebook

Для конфигурирования входа через учетную запись Facebook следует выполнить следующие шаги в разделе *Поставщики идентификации*:

1. Добавить поставщика, имеющего тип *Facebook*.
2. Ввести идентификатор поставщика (или не менять предложенный идентификатор).
3. Ввести название поставщика. Именно это название будет отображаться на странице аутентификации.

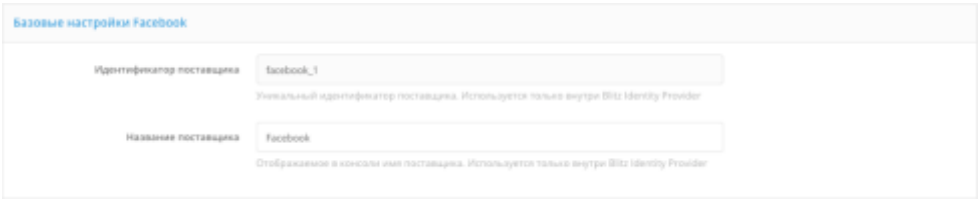

Рисунок 58 – Базовые настройки поставщика идентификации Facebook

4. Перейти в панель Facebook для разработчиков (см. Рисунок 59)[20], в котором выполнить следующие операции:
   - войдите с помощью своей учетной записи Facebook и при необходимости зарегистрируйтесь в качестве разработчика;
   - добавьте новое приложение, указав его название, адрес электронной почты для связи и категорию приложения;
   - создайте идентификатор приложения;
   - перейдите в настройки приложения, раздел «Основное». В этом разделе указать параметр «Домены приложения» (должен соответствовать домену, на котором установлен Blitz Identity Provider) и добавить сайт с аналогичным URL.
   - Перейти в раздел «Проверка приложения» и активировать пункт *Сделать приложение «...» доступным для всех?*

[20] См.: https://developers.facebook.com/apps/

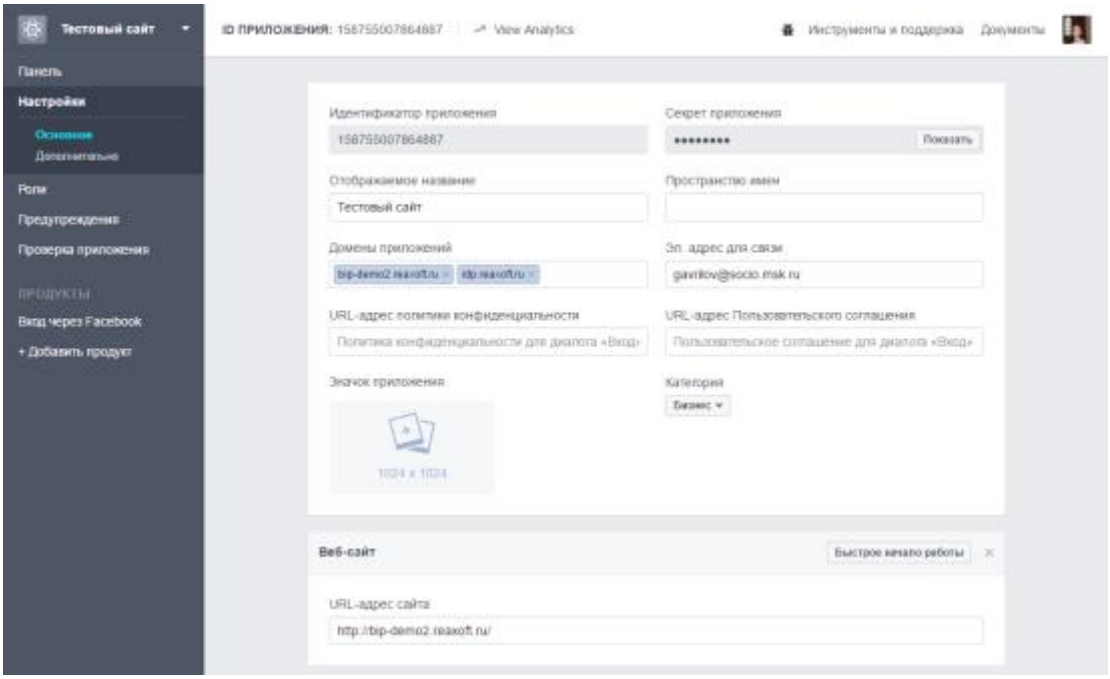

Рисунок 59 – Настройки в панели Facebook для разработчиков

5. Перейти в Blitz Identity Provider и заполнить дополнительные настройки поставщика идентификации (см. Рисунок 60), которые включают в себя:
   - идентификатор приложения (App ID), полученный в панели Facebook для разработчиков;
   - секрет приложения (App Secret), полученный в панели Facebook для разработчиков;
   - запрашиваемые разрешения (scope), предусмотренные в Facebook[21];
   - запрашиваемые атрибуты, предусмотренные в Facebook; допустимо указывать только те атрибуты, которые предусмотрены выбранными разрешениями;
   - атрибут Facebook, который будет использоваться для идентификации учетной записи в Blitz Identity Provider. Этот атрибут по смыслу соответствует базовому идентификатору, определяемому в разделе *Хранилище*;
   - правила соответствия (маппинга) между атрибутами, определенными в Blitz Identity Provider, и атрибутами Facebook. Например, правило mail=${email} означает, что атрибут с именем *mail* в Blitz Identity Provider будет заполняться значением из атрибута *email* учетной записи Facebook (для пользователей, воспользовавшихся этим поставщиком идентификации);
   - проставить галочку у тех атрибутов, по которым должен осуществляться поиск (в разделе *Пользователи*);

[21] См.: https://developers.facebook.com/docs/facebook-login/permissions/

6. Сохранить данные и в Blitz Identity Provider, и в панели Facebook для разработчиков.

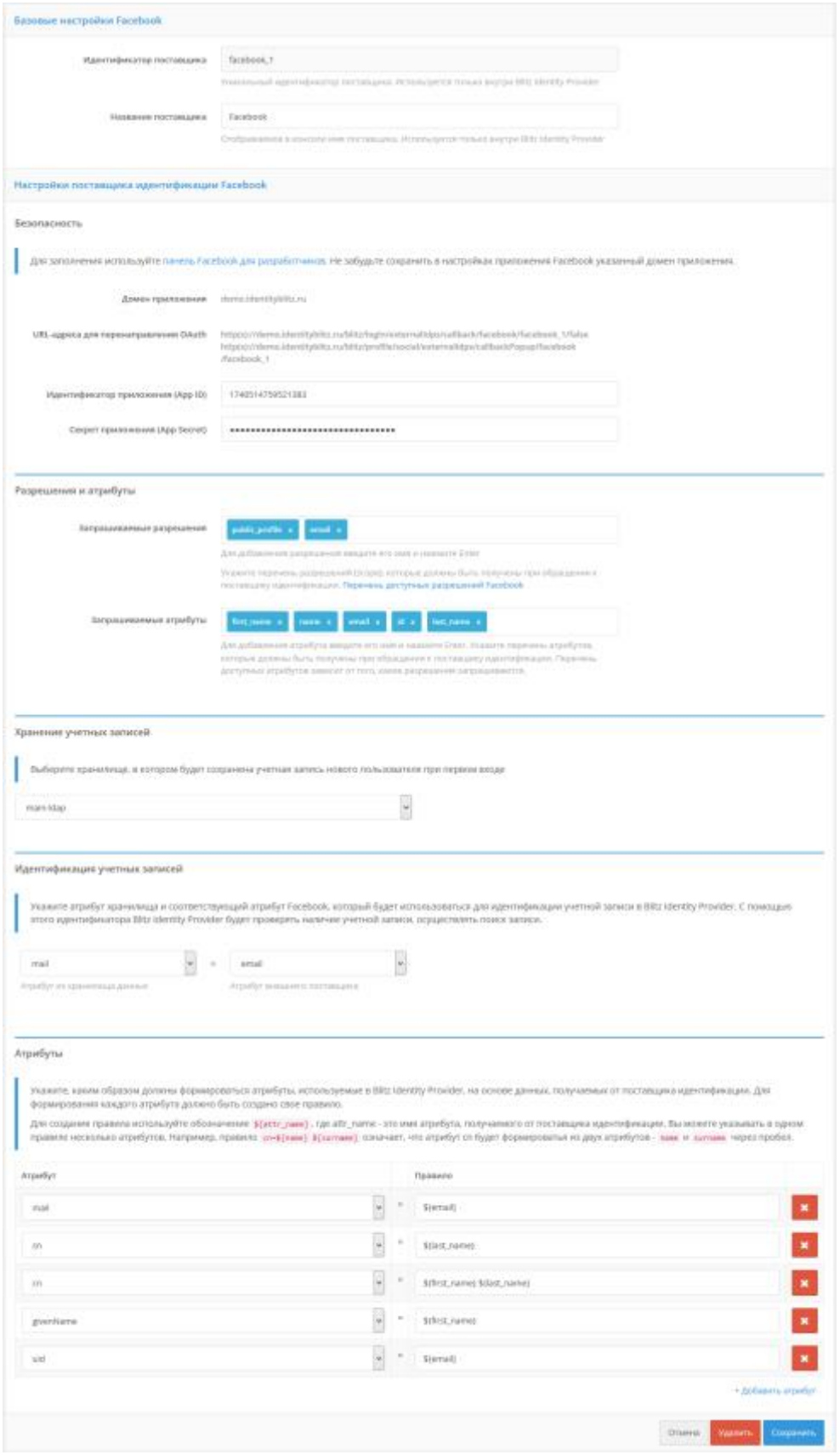

Рисунок 60 – Дополнительные настройки поставщика идентификации Facebook

2.7.3. Вход через ВКонтакте

Для конфигурирования входа через учетную запись ВКонтакте следует выполнить следующие шаги в разделе «Поставщики идентификации»:

1. Добавить поставщика, имеющего тип *VK*.
2. Ввести идентификатор поставщика (или не менять предложенный идентификатор).
3. Ввести название поставщика. Именно это название будет отображаться на странице аутентификации.

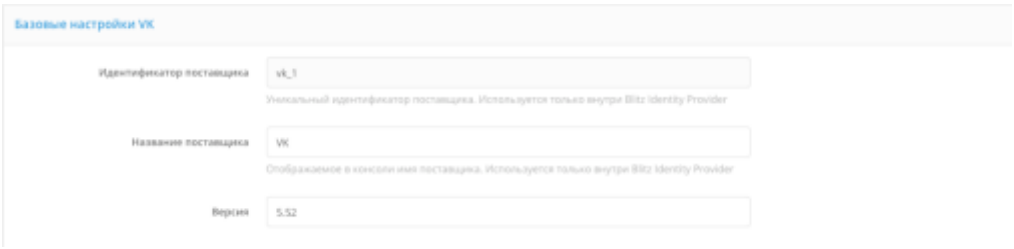

Рисунок 61 – Базовые настройки поставщика идентификации ВКонтакте

4. Перейти в панель VK для разработчиков (см. Рисунок 62)[22], в котором выполнить следующие операции:
   - войдите с помощью своей учетной записи ВКонтакте;
   - перейти в раздел «Мои приложения»;
   - выбрать пункт «Создать приложение»;
   - выбрать тип создаваемого приложения – «Веб-сайт», указать его название, адрес, и домен;
   - в появившемся окне настоек приложения прописать базовый домен приложения (должен совпадать с доменом, на котором установлен Blitz Identity Provider).

[22] См.: https://new.vk.com/dev

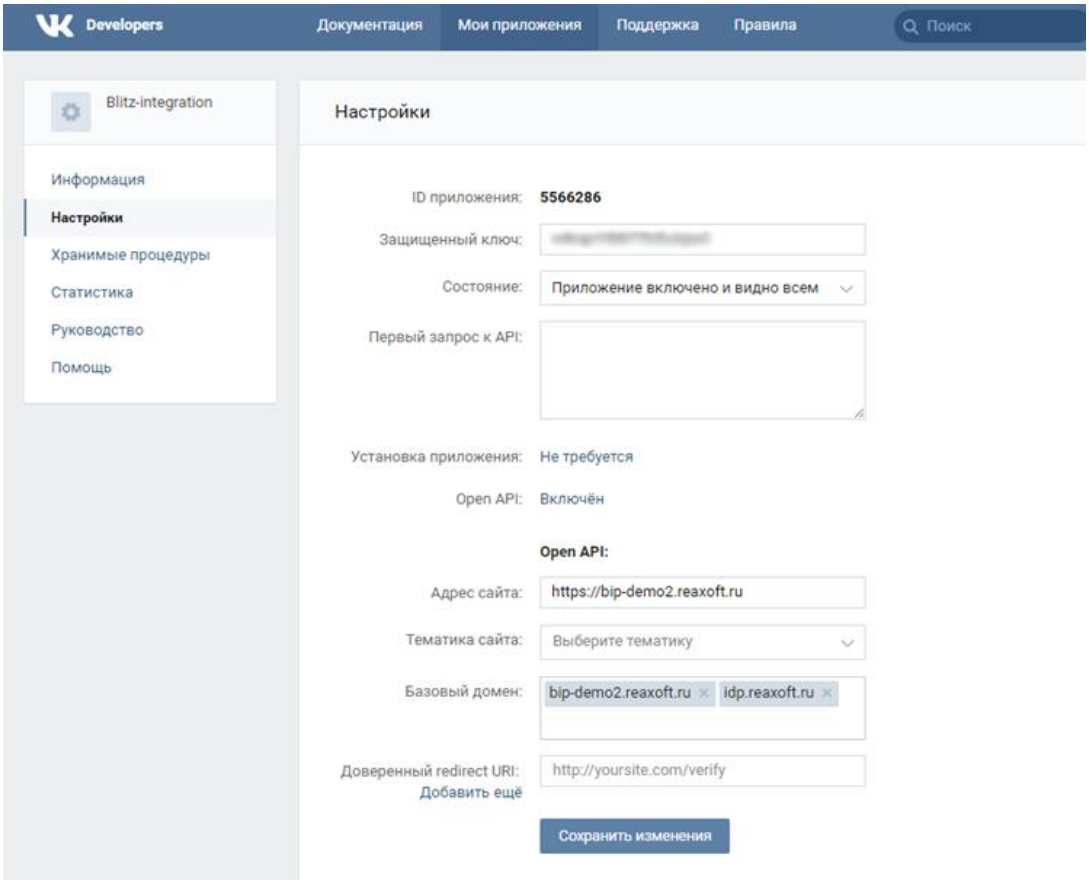

Рисунок 62 – Настройки в панели VK для разработчиков

5. Перейти в Blitz Identity Provider и заполнить дополнительные настройки поставщика идентификации (см. Рисунок 63), которые включают в себя:
   - ID приложения, полученный в панели VK для разработчиков;
   - защищенный ключ, полученный в панели VK для разработчиков;
   - запрашиваемые разрешения, предусмотренные в ВКонтакте[23];
   - атрибут ВКонтакте, который будет использоваться для идентификации учетной записи в Blitz Identity Provider. Этот атрибут по смыслу соответствует базовому идентификатору, определяемому в разделе *Хранилище*;
   - правила соответствия (маппинга) между атрибутами, определенными в Blitz Identity Provider, и атрибутами ВКонтакте. Например, правило mail=${email} означает, что атрибут с именем *mail* в Blitz Identity Provider будет заполняться значением из атрибута *email* учетной записи ВКонтакте (для пользователей, воспользовавшихся этим поставщиком идентификации);
   - проставить галочку у тех атрибутов, по которым должен осуществляться поиск (в разделе *Пользователи*);
6. Сохранить данные и в Blitz Identity Provider, и в панели VK для разработчиков.

[23] См.: https://new.vk.com/dev/permissions

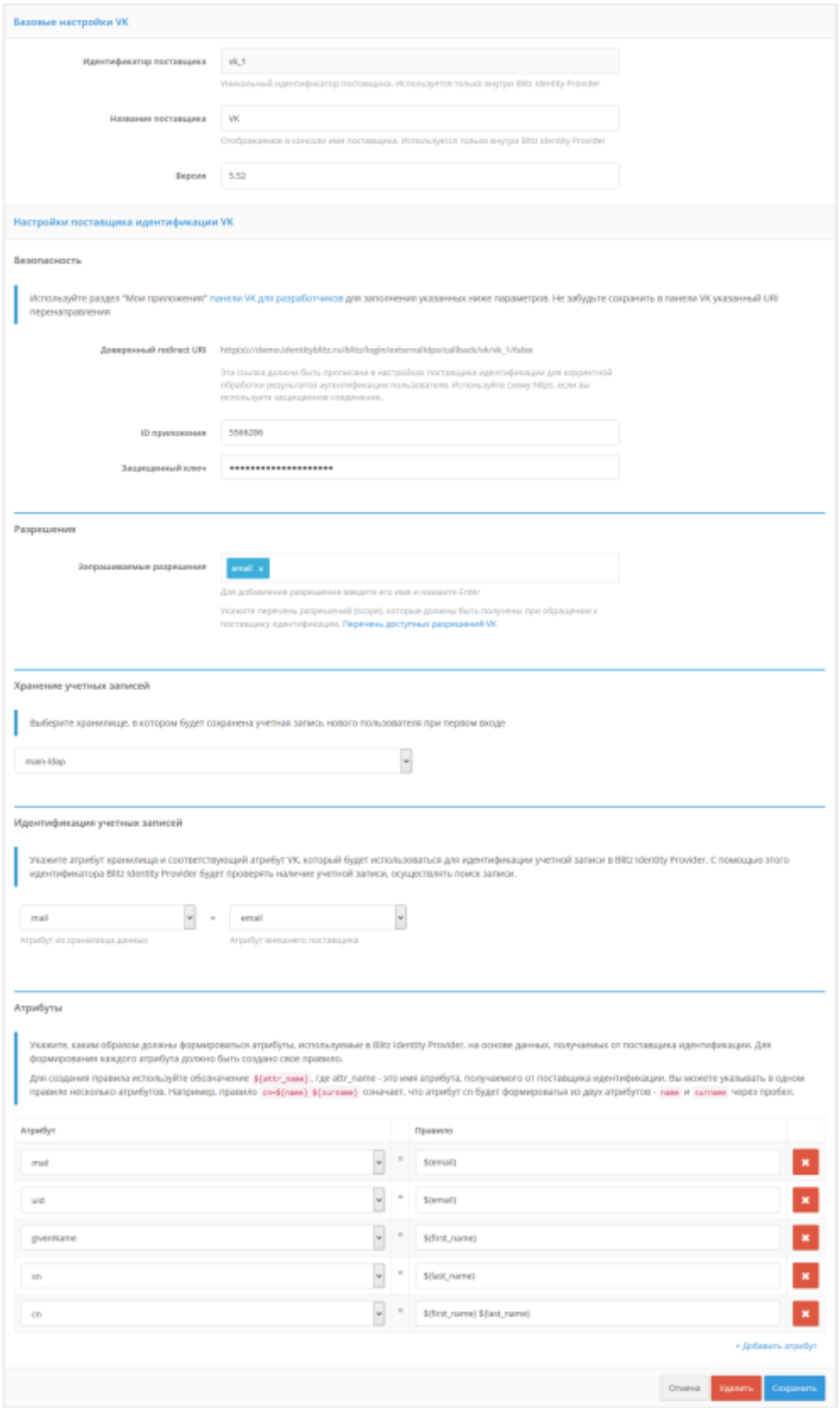

Рисунок 63 – Дополнительные настройки поставщика идентификации ВКонтакте

2.7.4. Вход через Единую систему идентификации и аутентификации (ЕСИА)

Для конфигурирования входа через учетную запись ЕСИА следует выполнить следующие шаги в разделе «Поставщики идентификации»:

1. Добавить поставщика, имеющего тип *ЕСИА*.
2. Ввести идентификатор поставщика (или не менять предложенный идентификатор).
3. Ввести название поставщика. Именно это название будет отображаться на странице аутентификации.

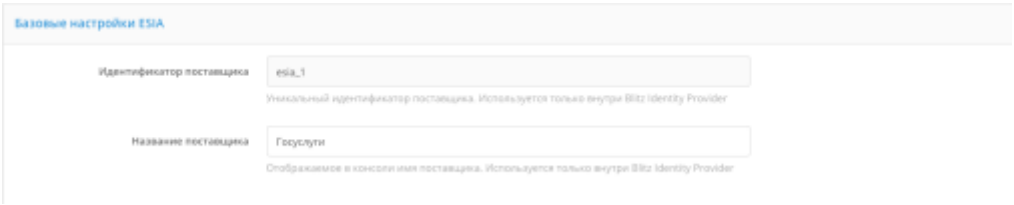

Рисунок 64 – Базовые настройки поставщика идентификации ЕСИА

4. Осуществить регистрацию информационной системы организации через Технологический портал ЕСИА[24], в котором выполнить следующие операции:
   - нажать на кнопку «Добавить систему»;
   - указать название системы, отображаемое название, мнемонику системы и выбрать ответственного сотрудника (см. Рисунок 65);
   - сохранить данные и перейти к настройке сертификатов информационной системы;
   - сгенерировать ключ электронной подписи для подключаемой информационной системы и экспортировать его сертификат; загрузить сертификат для зарегистрированной информационной системы на Технологическом портал (см. Рисунок 66);

[24] См.: https://esia.gosuslugi.ru/console/tech/ До регистрации ИС в ЕСИА необходимо зарегистрировать учетную запись организации в ЕСИА и дать одному из сотрудников доступ к Технологическому порталу.

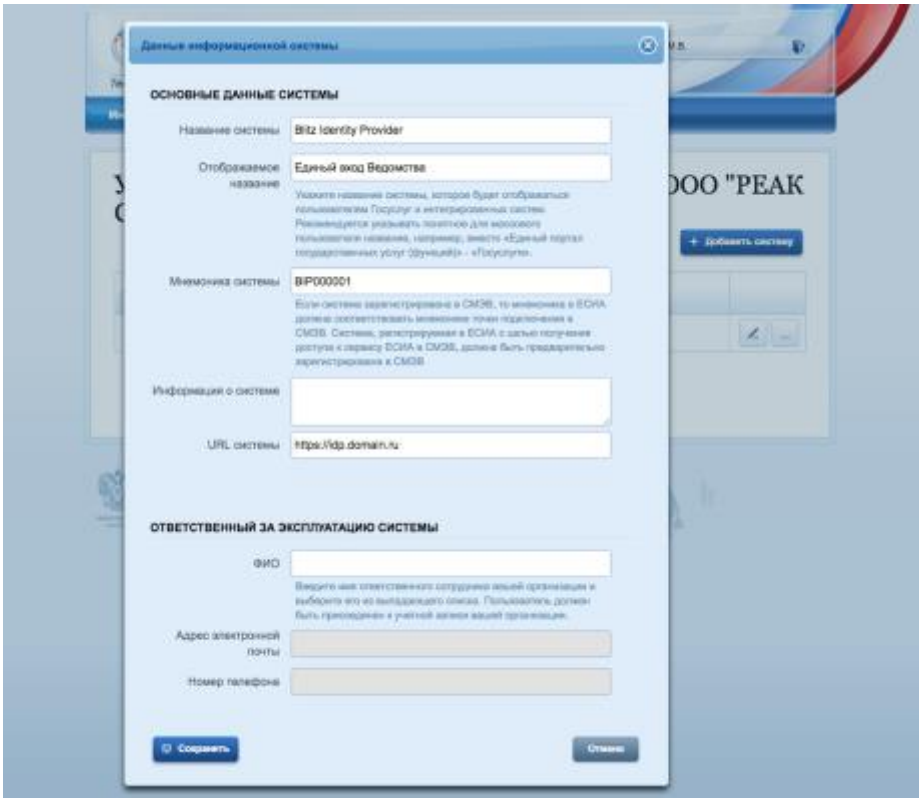

Рисунок 65 – Добавление системы в Технологическом портале ЕСИА

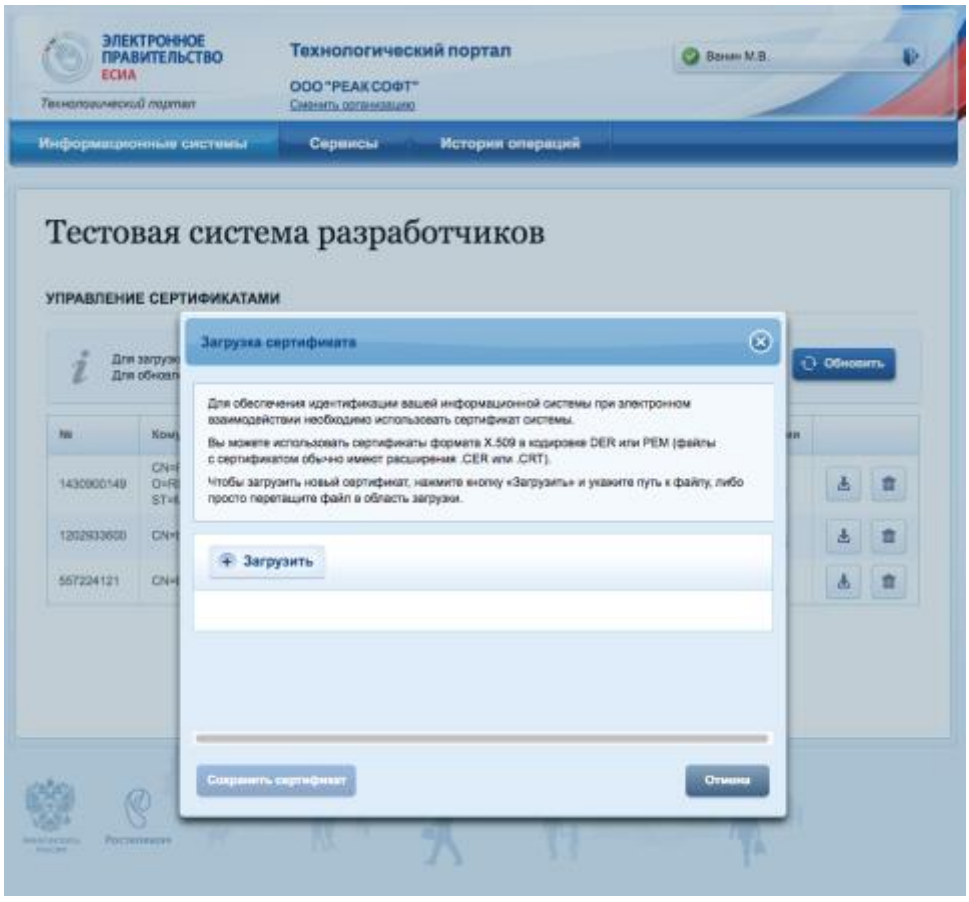

Рисунок 66 – Добавление сертификата системы в Технологическом портале ЕСИА

5. Перейти в Blitz Identity Provider и заполнить дополнительные настройки поставщика идентификации (см. Рисунок 67), которые включают в себя:
   - URI внешнего поставщика – домен среды ЕСИА, к которой производится подключение, например, https://esia.gosuslugi.ru;
   - мнемоника системы, указанная ранее в Технологическом портале ЕСИА;
   - идентификатор ключа электронной подписи (alias) – идентификатор ключа электронной подписи, загруженный в хранилище Blitz Identity Provider (хранилище, указанное в разделе keystore конфигурационного файла Blitz Identity Provider); именно сертификат ключа этой электронной подписи должен быть загружен в Технологический портал ЕСИА;
   - запрашиваемые разрешения – перечень запрашиваемых разрешений из ЕСИА;
   - запрашиваемые данные пользователя – необходимо отметить те данные, которые следует получать из ЕСИА; эти данные должны быть доступны по запрашиваемым разрешениям;
   - правила соответствия (маппинга) между атрибутами, определенными в Blitz Identity Provider, и атрибутами ЕСИА. Например, правило mail=${email} означает, что атрибут с именем *mail* в Blitz Identity Provider будет заполняться значением из атрибута *email* учетной записи ЕСИА (для пользователей, воспользовавшихся этим поставщиком идентификации);
   - проставить галочку у тех атрибутов, по которым должен осуществляться поиск (в разделе *Пользователи*);
6. Сохранить данные в Blitz Identity Provider.

Чтобы вход через ЕСИА заработал, необходимо получить официальное разрешение на проведение идентификации и аутентификации пользователей с помощью зарегистрированной системы и получить доступ к тестовой / промышленной среде ЕСИА[25].

[25] Подробнее см.: http://identityblitz.ru/services/esia-integration#moreESIAhelp

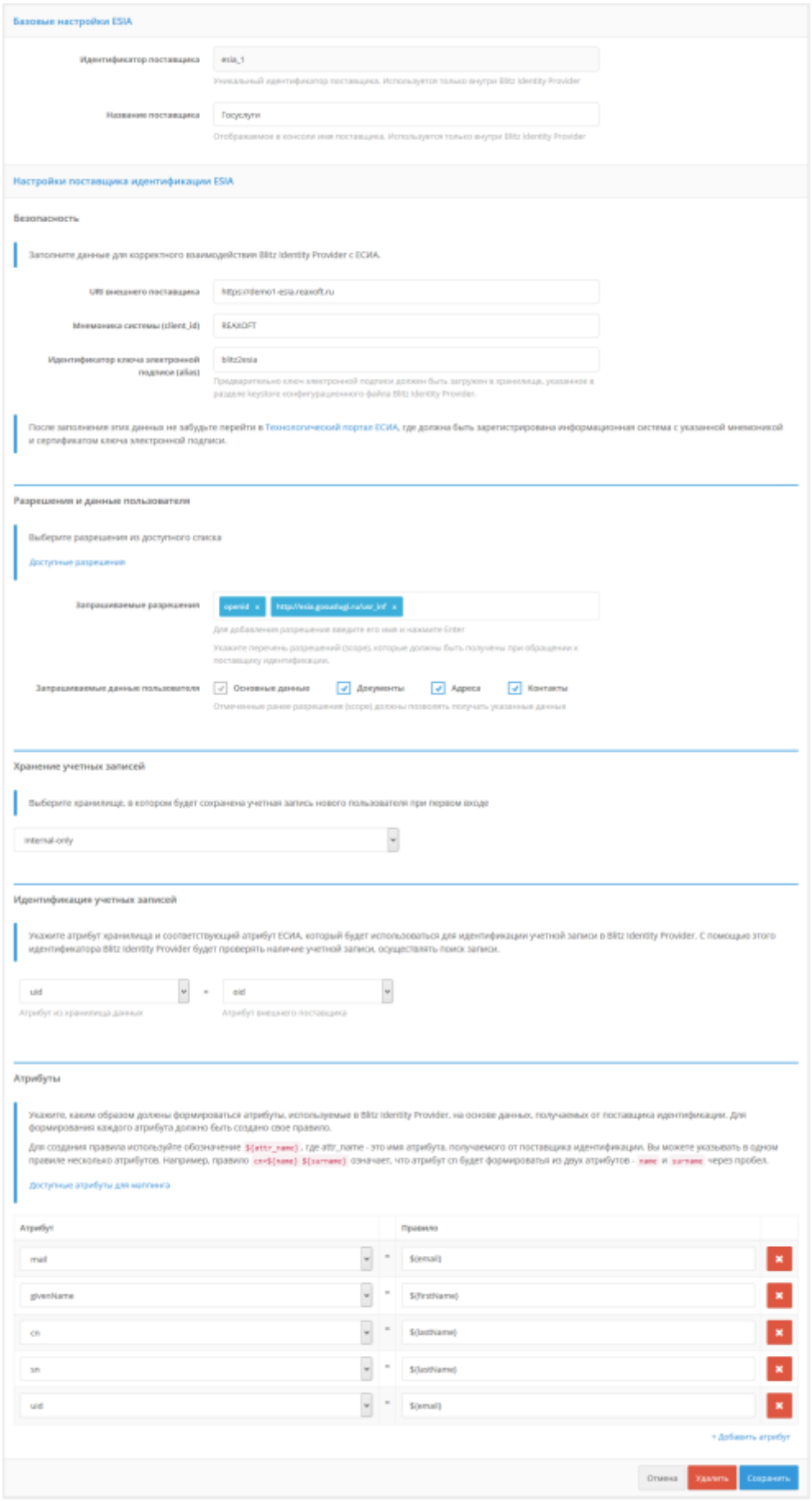

Рисунок 67 – Дополнительные настройки поставщика идентификации ЕСИА

2.7.5. Вход через другую установку Blitz Identity Provider

Для конфигурирования входа через учетную запись другого Blitz Identity Provider (например, установленного в другой организации, далее – *доверенный Blitz Identity Provider*) следует выполнить следующие шаги в разделе Поставщики идентификации:

1. Добавить поставщика, имеющего тип *Blitz Identity Provider*.
2. Ввести идентификатор поставщика (или не менять предложенный идентификатор).
3. Ввести название поставщика. Именно это название будет отображаться на странице аутентификации.

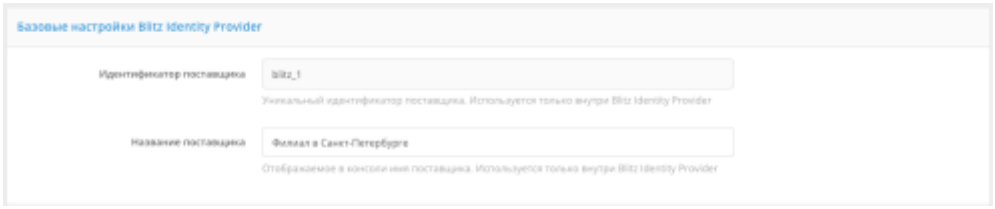

Рисунок 68 – Базовые настройки поставщика идентификации Blitz Identity Provider при настройке федеративного доступа

4. Открыть консоль управления доверенного Blitz Identity Provider (или попросить администратора другого Blitz Identity Provider это сделать) и выполнить следующие операции:
   - перейти в раздел *Приложения*;
   - нажать на кнопку «Добавить приложение» (см. Рисунок 69);
   - указать идентификатор приложения, название и домен приложения;
   - сохранить приложение и перейти к его настройке;
   - выбрать протокол подключения OAuth 2.0 (см. Рисунок 70);
   - указать секрет (client_secret), либо оставить предзаполненный вариант;
   - указать префикс ссылки возврата, в качестве которой указать URL основной Blitz Identity Provider, в который будет осуществляться вход;
   - произвести настройку необходимых разрешений в разделе *OAuth 2.0*.

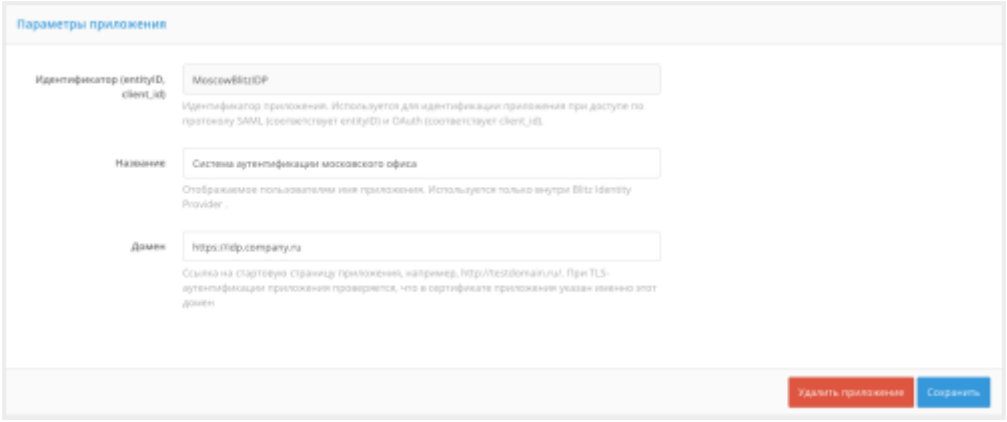

Рисунок 69 – Настройки подключения на стороне внешнего поставщика идентификации Blitz Identity Provider – создание приложения

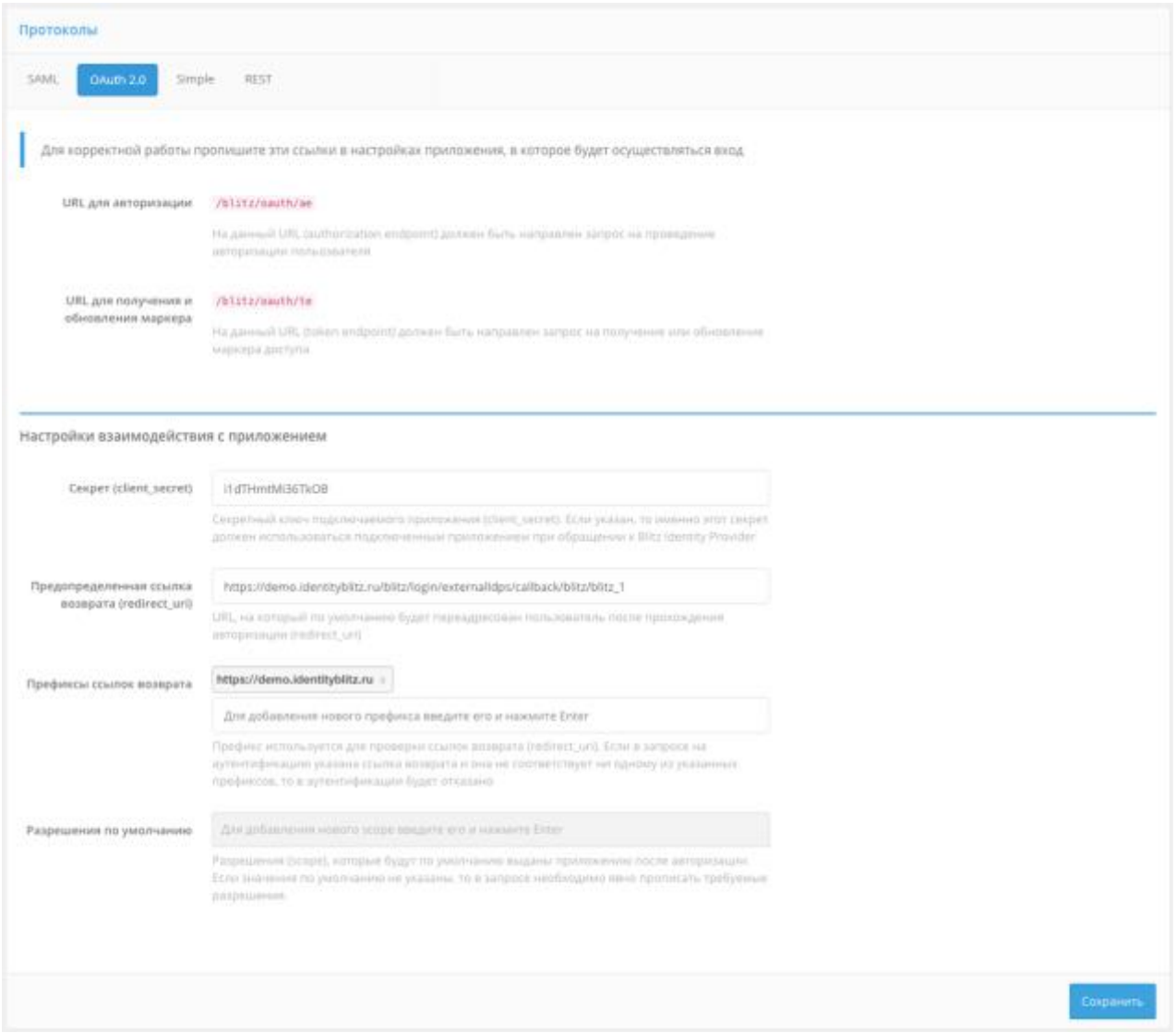

Рисунок 70 – Настройки подключения на стороне внешнего поставщика идентификации Blitz Identity Provider – настройка OAuth 2.0

5. Перейти в Blitz Identity Provider и заполнить дополнительные настройки поставщика идентификации (см. Рисунок 71), которые включают в себя:
   - URI внешнего поставщика – домен, на котором установлен доверенный Blitz Identity Provider;
   - идентификатор (client_id), указанный в настройках доверенного Blitz Identity Provider;
   - секрет (client_secret), указанный в настройках доверенного Blitz Identity Provider;
   - запрашиваемые разрешения, данные разрешения должны быть определены в разделе *OAuth 2.0* доверенного Blitz Identity Provider;
   - имя пользователя – атрибут доверенного Blitz Identity Provider, который будет использоваться в качестве имени пользователя (обеспечивает проверку наличия пользователя с таким именем);
   - идентификатор – атрибут доверенного Blitz Identity Provider, который будет использоваться в качестве идентификатора пользователя (обеспечивает уникальность учетной записи даже при изменении атрибута, отвечающего за имя пользователя);
   - правила соответствия (маппинга) между атрибутами, определенными в Blitz Identity Provider, и атрибутами доверенного Blitz Identity Provider;
   - проставить галочку у тех атрибутов, по которым должен осуществляться поиск (в разделе *Пользователи*);
6. Сохранить данные в Blitz Identity Provider.

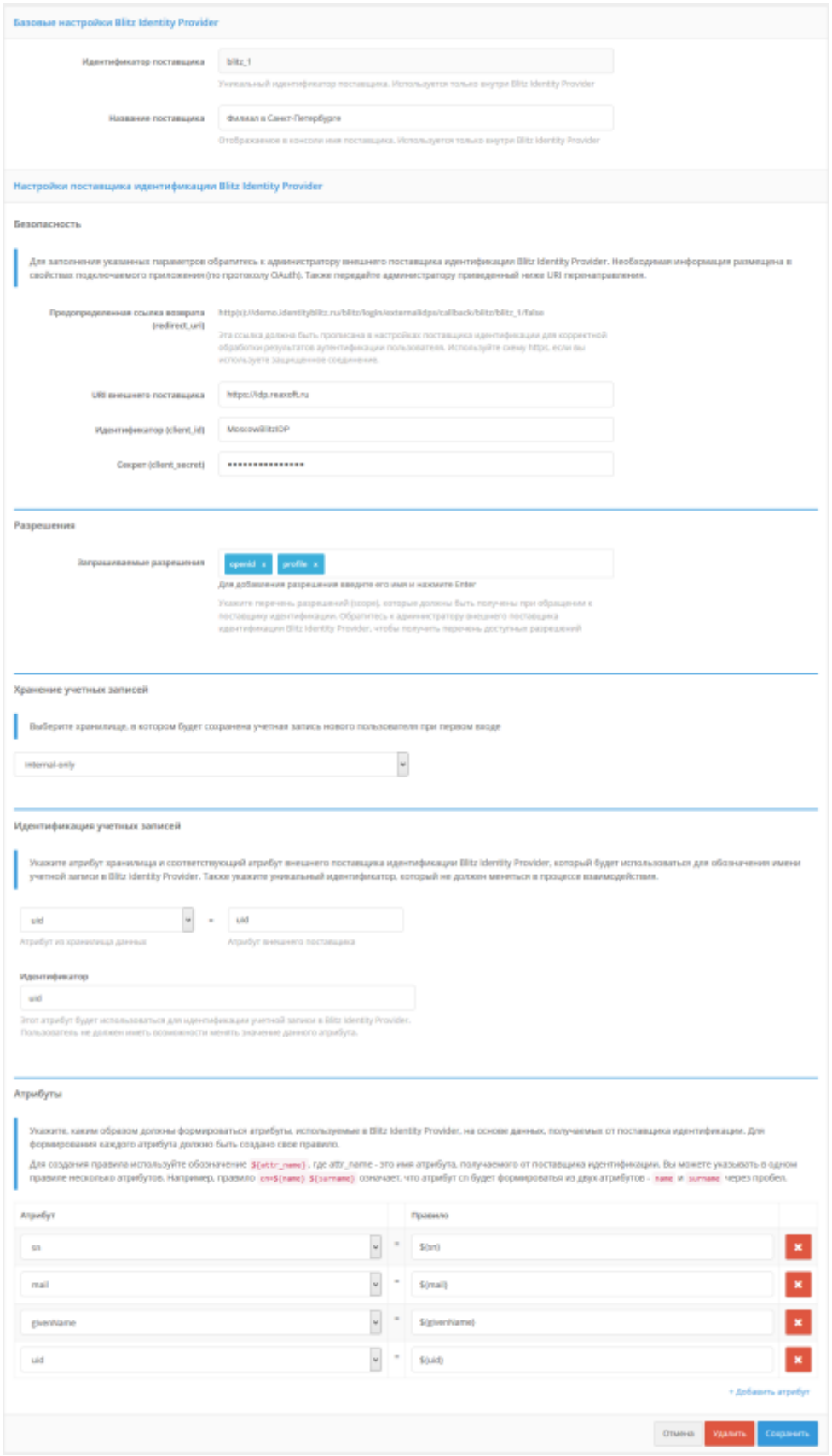

Рисунок 71 – Настройки подключения к внешнему поставщику идентификации Blitz Identity Provider (фрагмент)

7. В разделе *Аутентификация* консоли управления включить использование метода аутентификации с использованием соответствующего внешнего сервиса идентификации (см. раздел 2.3.4).

Если все настроено корректно, то можно попробовать войти в приложение одной организации (см. Рисунок 72, кнопка «Войти через …») с использованием учетной записи другой организации (см. Рисунок 73).

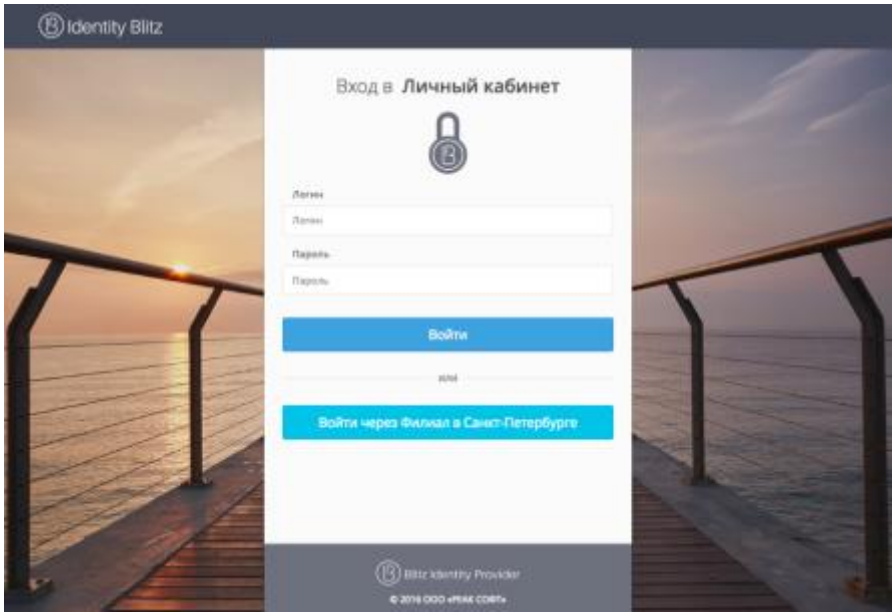

Рисунок 72 – Пример экрана входа с кнопкой входа через внешний поставщик идентификации

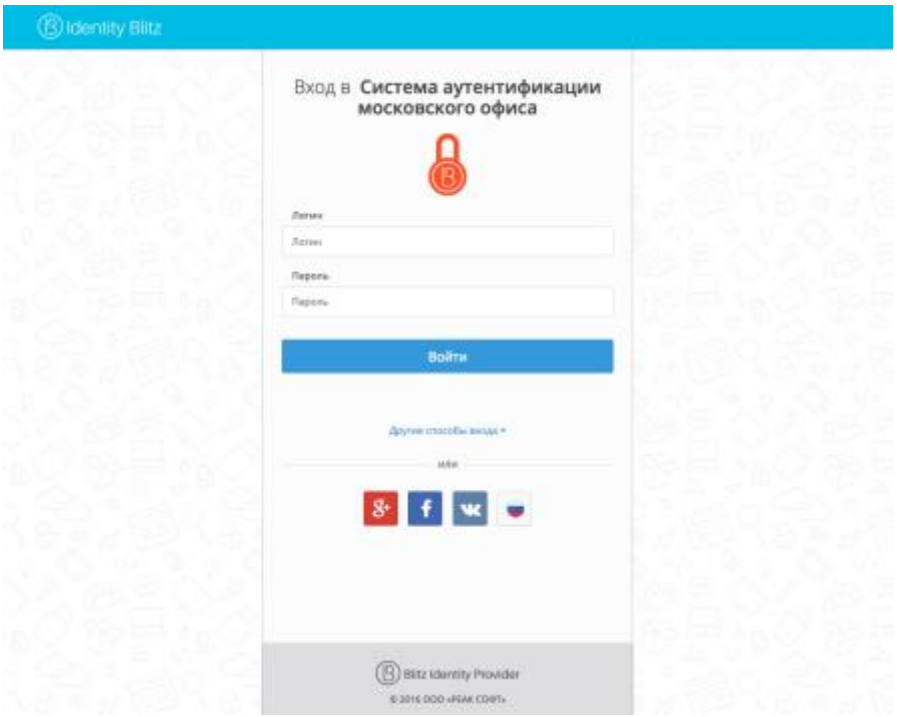

Рисунок 73 – Пример запроса идентификации внешним поставщиком идентификации

2.8. Управление данными пользователей

В разделе *Пользователи* консоли управления администратор Blitz Identity Provider может осуществлять следующие операции:

- поиск учетных записей пользователей;
- добавление учетных записей пользователей;
- просмотр идентификационных данных пользователей;
- привязка учетных записей внешних систем;
- задание значений дополнительных атрибутов пользователей;
- привязку устройств для проведения усиленной аутентификации.

Общий вид страницы управления данными пользователей представлен на рисунке 74.

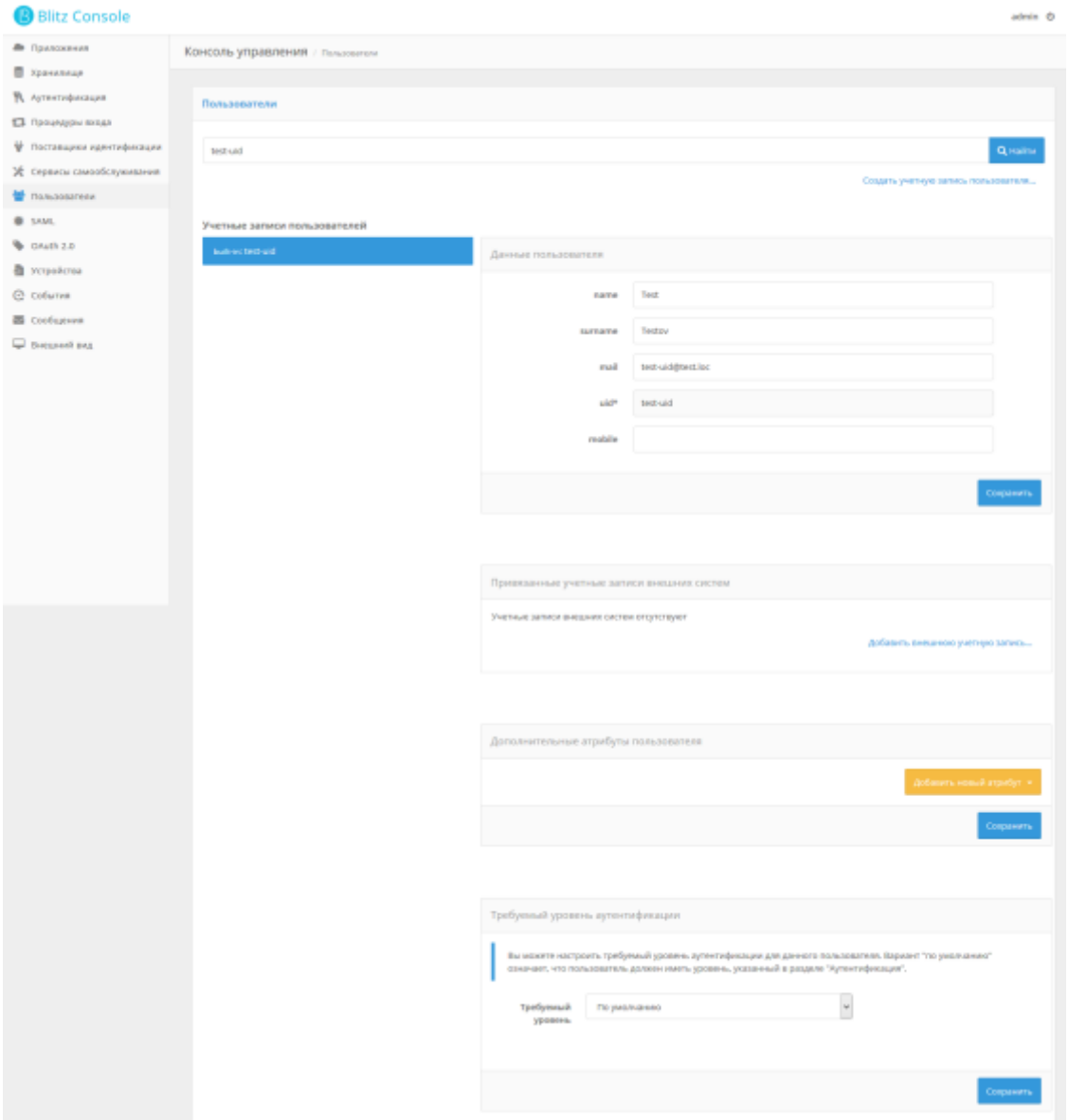

Рисунок 74 – Вид страницы управления пользователями (фрагмент)

2.8.1. Поиск учетных записей пользователей

Для поиска пользователей необходимо ввести идентификатор пользователя (или его часть со знаком *) и нажать на кнопку «Найти». В качестве отображаемого идентификатора используется атрибут, определенный в разделе *Хранилище* в качестве базового идентификатора, а также атрибуты, отмеченные как поисковые.

Для просмотра всех пользователей необходимо нажать «Найти» с вводом символа * в поле для поиска (полный перечень не будет отображаться при большом числе учетных записей в системе).

Перечень найденных пользователей содержит:

- значение атрибутов пользователя (все атрибуты, отмеченные как поисковые в Хранилище);
- хранилище, в котором найден пользователь.

Нажатие на любую из найденных учетных записей открывает детальную информацию о пользователе.

2.8.2. Добавление учетных записей пользователей

Для добавления новой учетной записи требуется нажать на ссылку «Создать учетную запись пользователя…». В отрывшемся окне:

- указать хранилище, в которым следует сохранить данные пользователя;
- задать все необходимые атрибуты;
- нажать на кнопку «Создать».

При создании учетной записи следует учитывать те ограничения, которые настроены для хранилища данных, в которое осуществляется запись. Например, если сохранение производится в LDAP-каталог, то должны быть заполнены все обязательные атрибуты, не нарушены ограничения на уникальность атрибутов и пр.

При этом с точки зрения Blitz Identity Provider обязательным является только идентификатор (соответствующий атрибут отмечен знаком «звездочка» (*)).

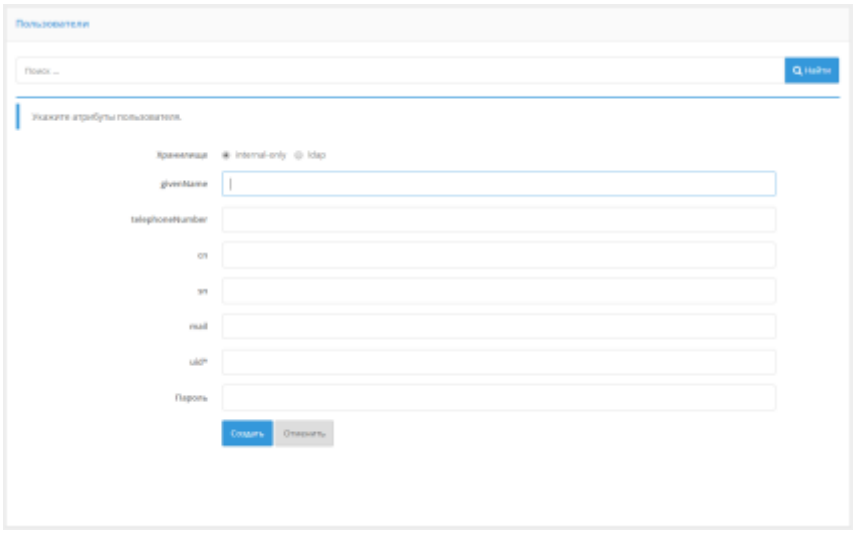

Рисунок 75 – Создание учетной записи пользователя

2.8.3. Добавление / назначение учетных записей пользователей для последующего входа через социальные сети.

В ряде случаев возникает необходимость обеспечить доступ только определенных пользователей через учетные записи внешних поставщиков идентификации. Этот режим входа «внешних» пользователей активируется в случае, если для выбранного внешнего поставщика идентификации выбрана опция, что доступ имеют «Только пользователи, привязанные администратором» (см. раздел 2.3.4).

Если учетная запись данного внешнего пользователя отсутствует, то следует нажать ссылку «Создать учетную запись пользователя…» и выполнить следующие действия:

1. Указать хранилище, в котором будет сохранена учетная запись «внешнего» пользователя.
2. Ввести идентификатор этого пользователя (атрибут, отмеченный «звездочкой» (*)).
3. Указать значение атрибута, по которому должна быть произведена привязка (например, адрес электронной почты), при необходимости – другие атрибуты.
4. Перейти к редактированию данных этой учетной записи, в раздел «Привязанные учетные записи внешних систем» и нажать на кнопку «Добавить внешнюю учетную запись». В появившемся окне:
   - выбрать необходимого поставщика идентификации;
   - указать значение атрибута, по которому будет произведено связывание.

Если учетная запись пользователя, которому необходимо дать возможность входить через внешнего поставщика, уже существует, то ее следует найти в разделе *Пользователи* и выполнить шаги 3–4, описанные выше.

2.8.4. Просмотр и изменение атрибутов пользователей

При нажатии на идентификатор любого найденного пользователя отображается информация о нем – карточка пользователя. Она содержит значения атрибутов, которые были определены в разделе *Хранилище*, а также привязанные учетные записи внешних поставщиков идентификации, заданные значения дополнительных атрибутов, привязанные средства аутентификации.

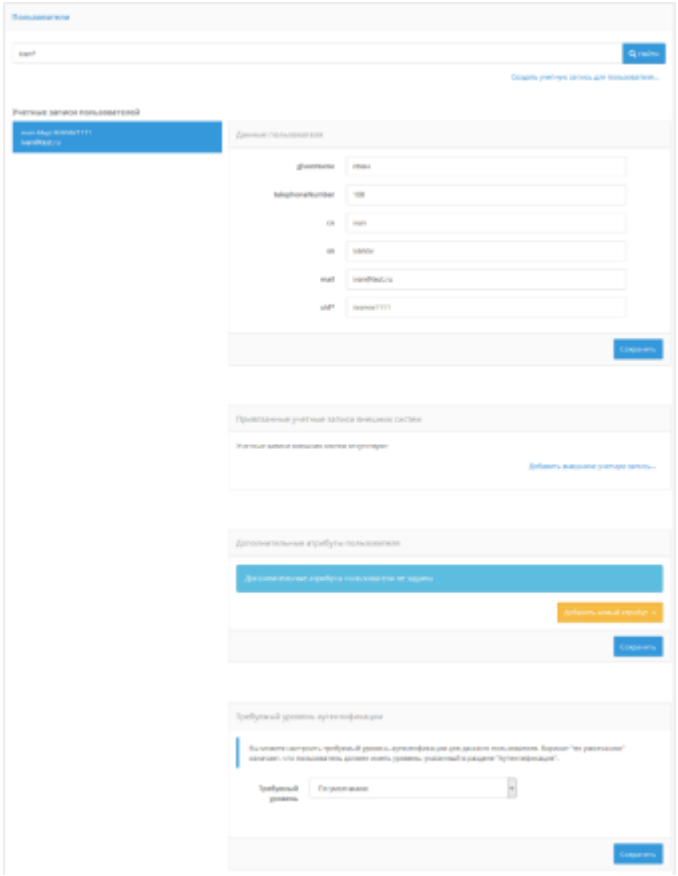

Рисунок 76 – Просмотр информации о пользователе (фрагмент)

На карточке пользователя можно совершать следующие операции:

- редактировать основные атрибуты пользователя;
- привязывать внешние учетные записи (см. п. 2.8.3);
- задавать значения дополнительным атрибутам (см. п. 2.8.4.1);
- изменять требуемый уровень аутентификации для пользователя;
- привязывать устройства для проведения аутентификации (генераторы разовых паролей) – 2.8.4.2.

2.8.4.1. Задание значений дополнительных атрибутов пользователей

При просмотре карточки выбранной учетной записи пользователя администратор может задать учетной записи значение нового дополнительного атрибута пользователя (выбрав его из выпадающего списка при нажатии кнопки «Добавить новый атрибут») или скорректировать ранее заданное значение атрибута (см. Рисунок 77).

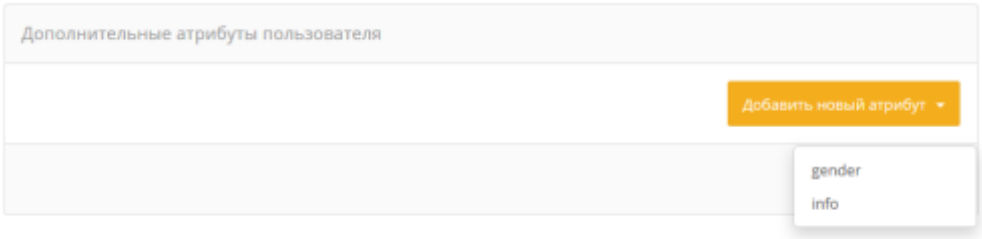

Рисунок 77 – Задание значений дополнительных атрибутов пользователей

2.8.4.2. Привязка устройств для проведения усиленной аутентификации

Администратор может привязать к учетной записи выбранного пользователя средство для проведения усиленной аутентификации. Например, можно привязать аппаратный HOTP/TOTP генератор по серийному номеру (Рисунок 78) либо привязать к учетной записи по QR-коду мобильное приложение, осуществляющее выработку TOTP-кодов (Рисунок 79).

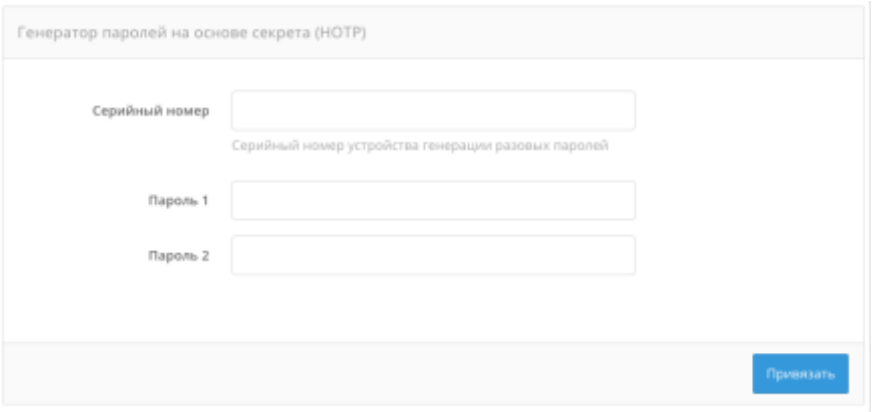

Рисунок 78 – Привязка HOTPустройства по серийному номеру администратором

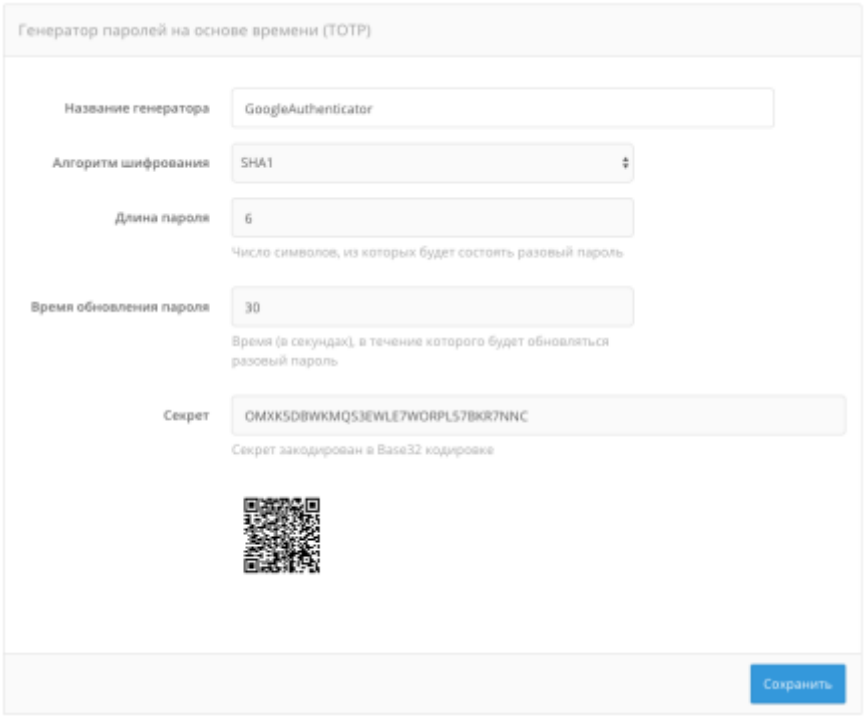

Рисунок 79 – Привязка TOTP-приложения по QR-коду администратором

2.9. Загрузка сведений о HOTP/TOTP-устройствах

Для возможности использования в качестве средств для проведения аутентификации второго фактора аппаратных ключей, осуществляющих генерацию кодов подтверждения по алгоритмам HOTP/TOTP, необходимо предварительно осуществить загрузку файла с описаниями партии устройств, полученного от поставщика HOTP/TOTP-генераторов. Файл содержит сведения о серийном номере генератора, векторе инициализации и ряд другой важной сервисной информации. Blitz Identity Provider поддерживает загрузку файлов различных популярных форматов (специализированные XML-файлы, CSV-файлы), используемых популярными производителями для описания устройств.

Для загрузки описаний устройств используется раздел *Устройства* консоли управления (см. Рисунок 80).

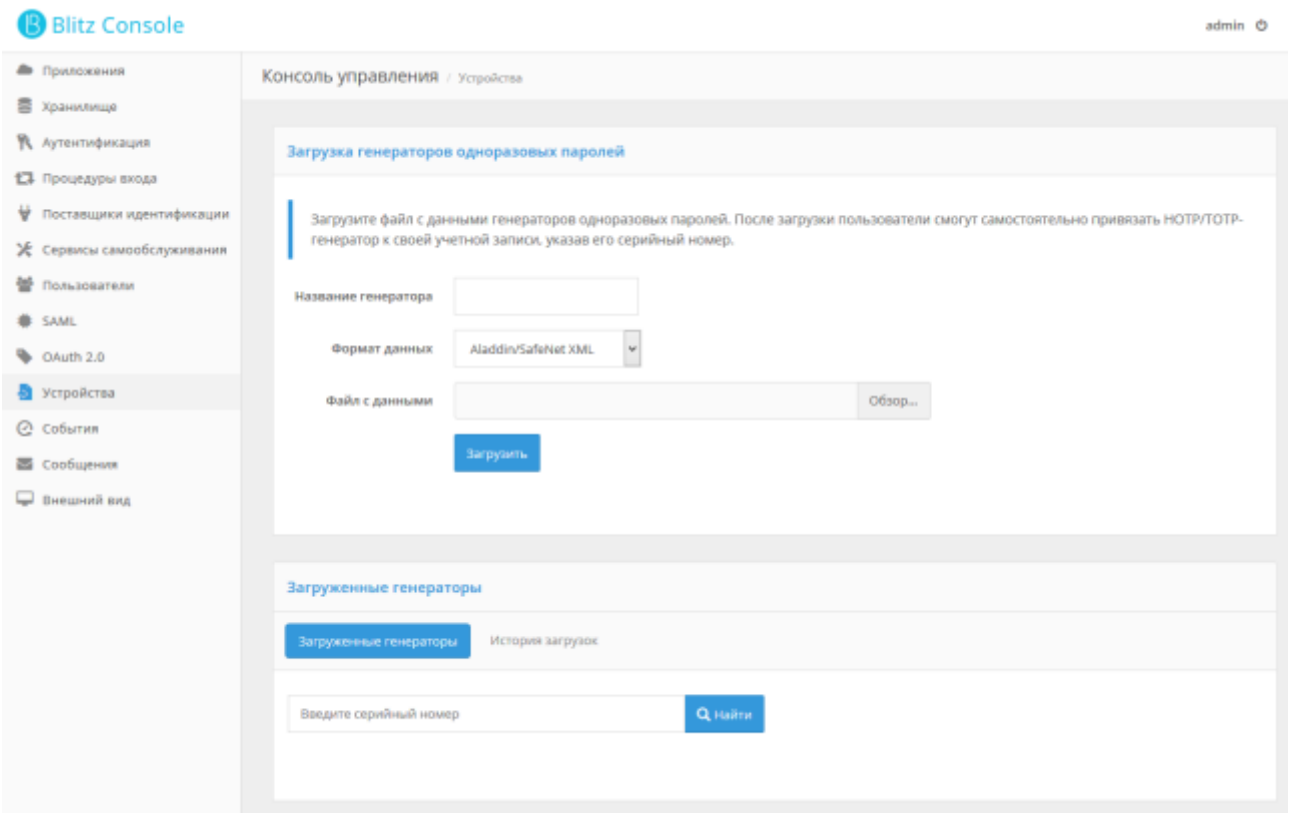

Рисунок 80 – Загрузка файлов с описаниями устройств генерации кодов

Для выполнения загрузки файла нужно задать имя для загружаемый генераторов (это может быть, например, имя устройства), формат данных, а также путь к файлу с описаниями устройств. По нажатии кнопки «Загрузить» Blitz Identity Provider сообщит, сколько записей устройств было загружено (и сколько возможно были отброшены, если их описание в файле было некорректно, либо запись об устройстве уже присутствуют в системе).

Также в данном разделе можно выполнить поиск устройства в базе по серийному номеру, и посмотреть, к какой учетной записи было привязано устройство, если оно было выдано пользователю.

2.10. Просмотр событий безопасности

Для ведения аудита безопасности и для просмотра зарегистрированных в журнале Blitz Identity Provider событий безопасности используется раздел *События* консоли управления. В этом разделе администратор может осуществлять фильтрацию событий безопасности по интересующим его критериям (по пользователю, за диапазон дат, по конкретному приложению, по группам событий, по IP-адресам) и просмотр детальной информации о найденных событиях.

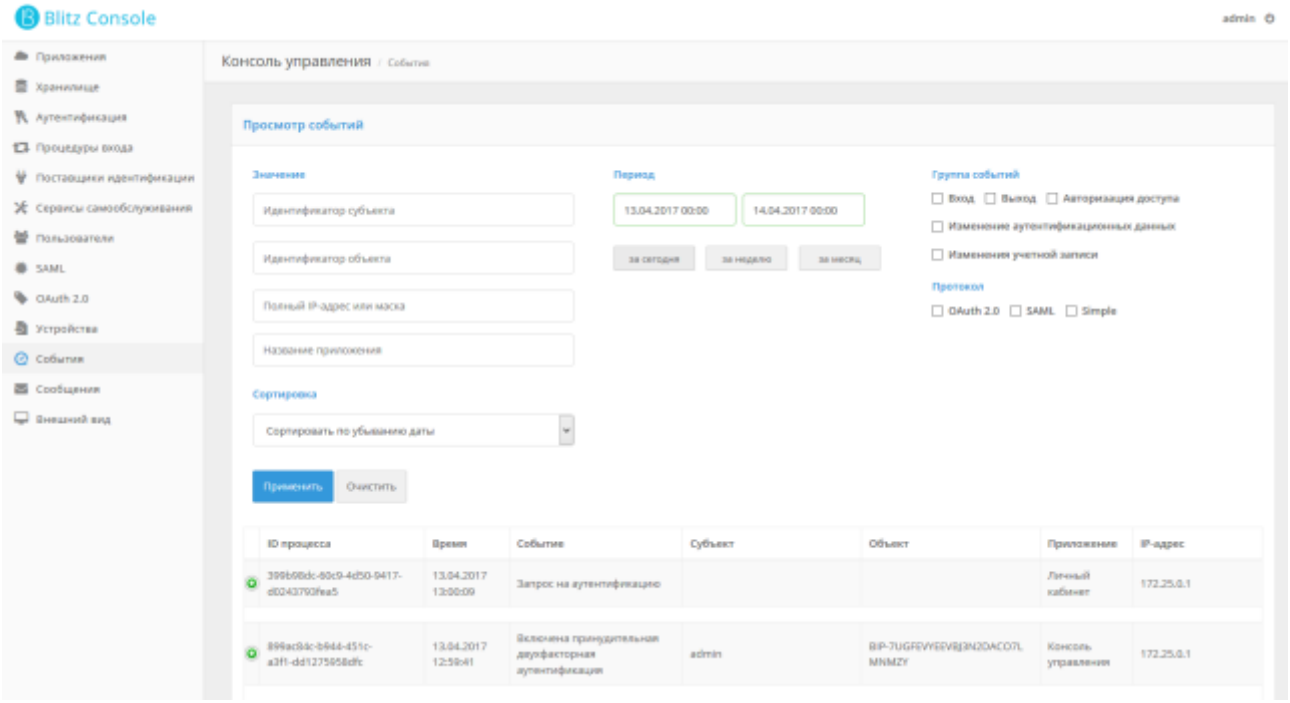

Рисунок 81 – Просмотр событий безопасность

2.11. Настройка подключений к системам отправки сообщений

Для задания настроек подключения к системам отправки сообщений используется раздел *Сообщения* консоли управления Blitz Identity Provider (см. Рисунок 82). В этом разделе можно настроить подключение к SMS-шлюзу и к SMTP-шлюзу организации.

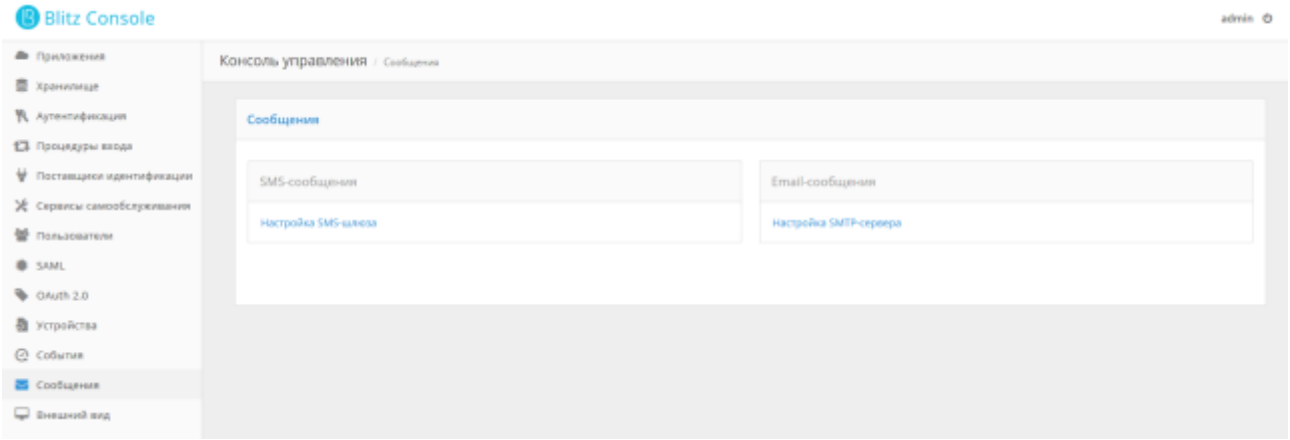

Рисунок 82 – Настройка подключения к системам отправки сообщений

2.11.1. Настройка подключения к SMS-шлюзу

Blitz Identity Provider необходима возможность отправлять SMS-сообщения, если используются следующие функции:

- усиленная аутентификация на основе отправки по SMS кода подтверждения;
- информирование о важных событиях безопасности по SMS;

- изменение номера мобильного телефона через «Профиль пользователя»;
- восстановление забытого пароля с использованием мобильного телефона как канала подтверждения владения учетной записью.

Настройки задаются в консоли управления Blitz Identity Provider в разделе «Сообщения». Экран настроек приведен на рисунке 83.

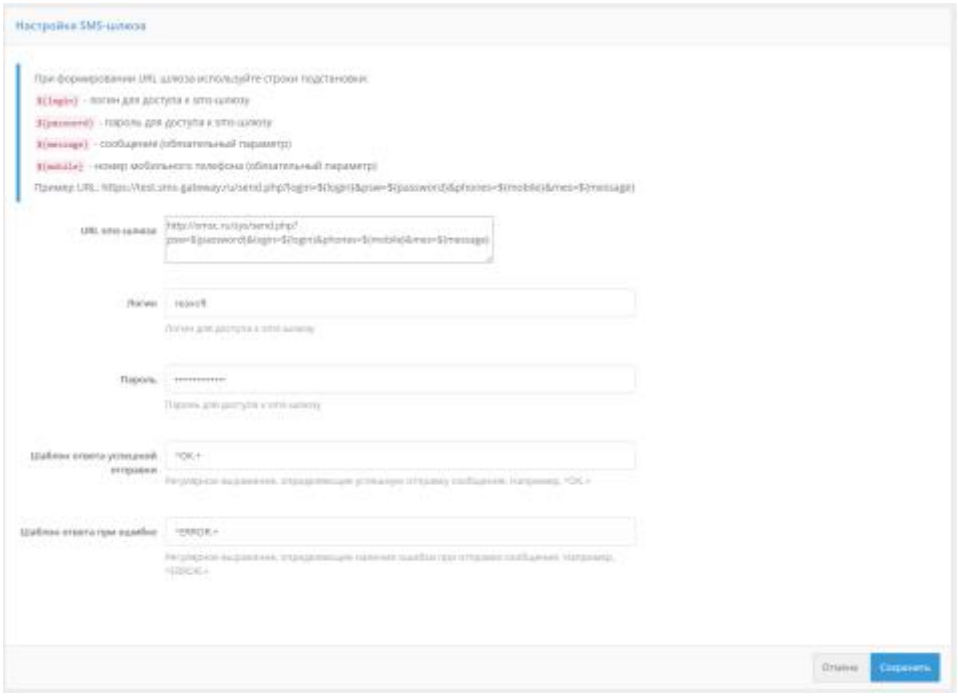

Рисунок 83 – Настройка подключения к SMS-шлюзу

Необходимо задать следующие настройки:

- URL sms-шлюза – задается в виде паттерна для формирования HTTP GET запроса к SMS-шлюзу для инициирования отправки им SMS. Пример настройки для SMS-шлюза smsc.ru:

```
https://smsc.ru/sys/send.php?psw=${password}&login=${login}&phones=${mobile}&mes=${message}
```

- логин и пароль для доступа к sms-шлюзу;
- шаблон проверки ответа от шлюза, означающего успешную отправку. Задается в виде регулярного выражения;
- шаблон проверки ответа от шлюза, означающего ошибку отправки сообщения. Задается в виде регулярного выражения.

2.11.2. Настройка подключения к SMTP-шлюзу

В Blitz Identity Provider необходимо настроить возможность отправлять email-сообщения, если используются следующие функции:

- информирование о важных событиях безопасности по email.
- изменение адреса электронной подписи через «Профиль пользователя».
- восстановление забытого пароля с использованием email как канала подтверждения

владения учетной записью.

- активация зарегистрированной учетной записи пользователя.

Настройки задаются в консоли управления Blitz Identity Provider в разделе Сообщения. Экран настроек приведен на рисунке 84.

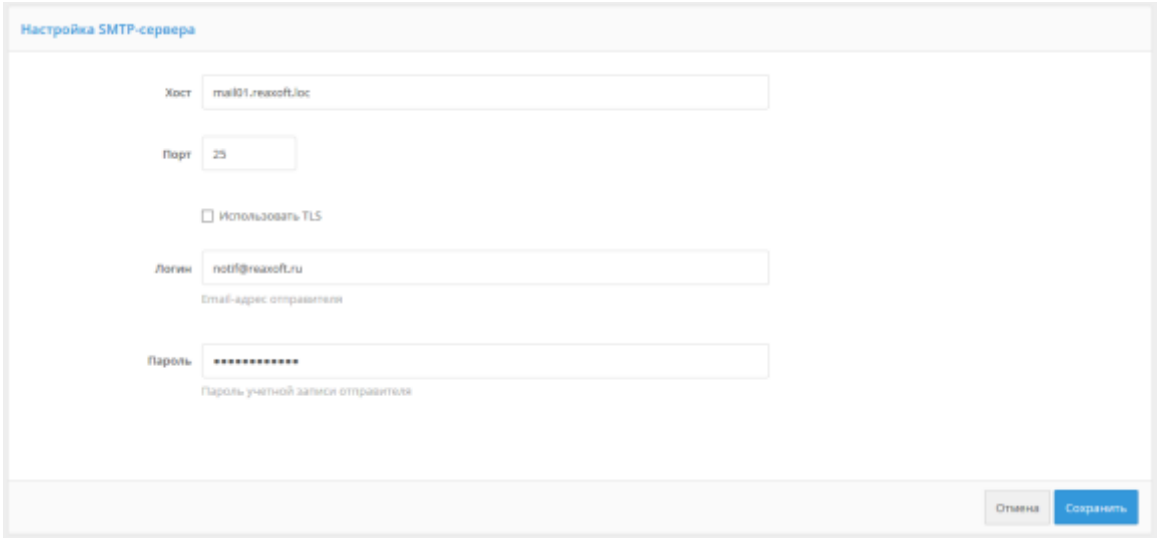

Рисунок 84 – Настройка подключения к SMTP-шлюзу

Необходимо задать следующие настройки:

- имя хоста SMTP-шлюза;
- порт хоста SMTP-шлюза;
- необходимо или нет использовать TLS для защищенного подключения к шлюзу;
- логин (email) учетной записи на SMTP-шлюзе, от имени которой Blitz Identity Provider будет производить отправку email;
- пароль от учетной записи на SMTP-шлюзе, от имени которой Blitz Identity Provider будет производить отправку email.

2.12. Настройка внешнего вида страницы входа

В разделе *Внешний вид* консоли управления администратор может настроить параметры отображения единой страницы входа. Если применяются приложения Blitz Identity Provider по регистрации пользователей и восстановлению пароля, то их внешний вид также будет соответствовать заданным настройкам внешнего вида единой страницы входа.

При входе в раздел *Внешний вид* отображается перечень настроенных шаблонов страницы входа. Каждый шаблон описывается:

- идентификатором;
- названием;
- перечнем приложений;
- описанием.

По умолчанию создан шаблон с идентификатором *default* – он используется для всех приложений, подключенных к Blitz Identity Provider, а также для страниц единого логаута.

Редактирование шаблона по умолчанию осуществляется с помощью специального конструктора (см. п. 2.12.1).

Также имеется возможность:

- создавать и изменять новые шаблоны с помощью конструктора и назначать их разным приложениям (п. 2.12.2);
- создавать и изменять новые шаблоны в ручном режиме (п. 2.12.3).

2.12.1. Редактирование шаблона по умолчанию

При открытии страницы редактирования шаблона по умолчанию отображается информация о самом шаблоне (идентификатор шаблона, название шаблона, описание и перечень приложений), а также интерфейс конструктора страницы входа.

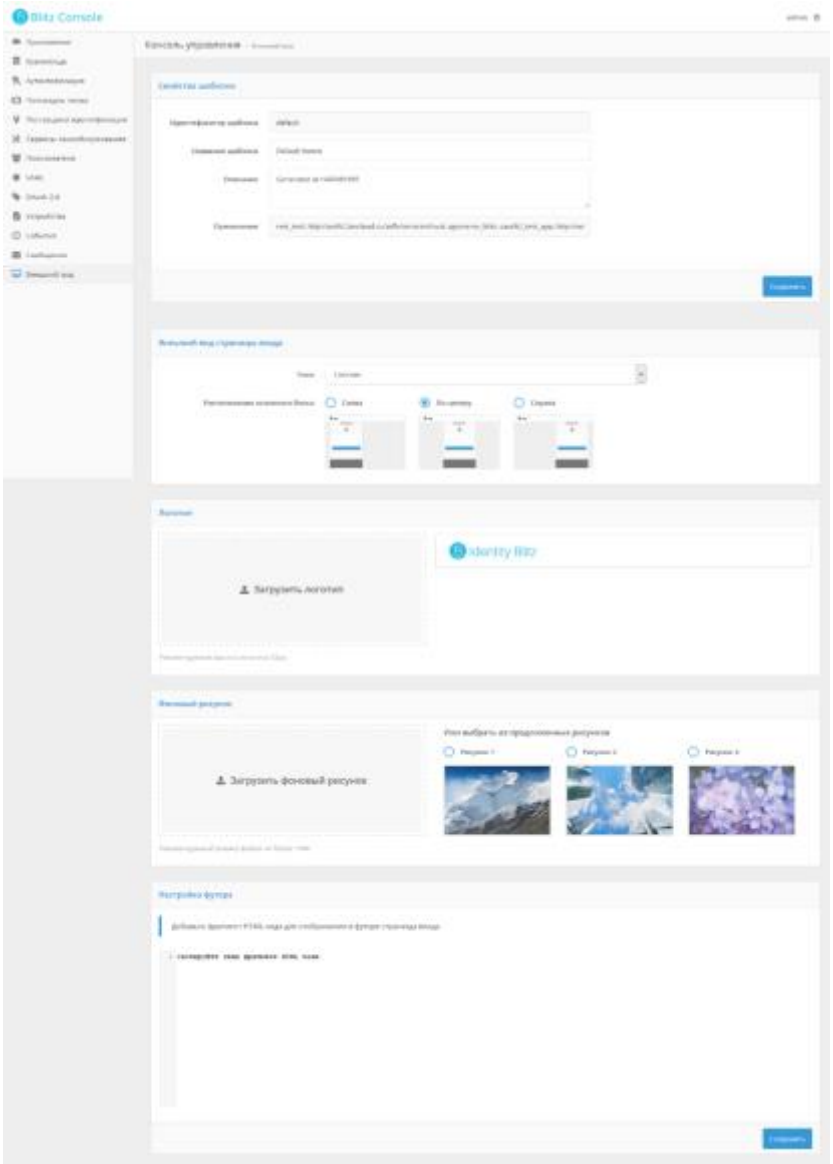

Рисунок 85 – Настройка внешнего вида страницы входа

В стандартной поставке конструктор Blitz Identity Provider предоставляет следующие возможности:

- три цветовых темы оформления элементов интерфейса;
- возможность определить местоположения блока ввода сведений (идентификации и аутентификации, регистрации, восстановления пароля);
- возможность загрузки логотипа компании для отображения в заголовке страницы;
- выбор фонового рисунка (можно выбрать из 3 стандартных рисунков в каждой теме оформления, либо загрузить свой собственный фоновый рисунок);
- настройка содержания футера страницы входа.

На рисунках 86, 87, 88 приведены некоторые примеры страниц входа, полученных в результате стандартной настройки.

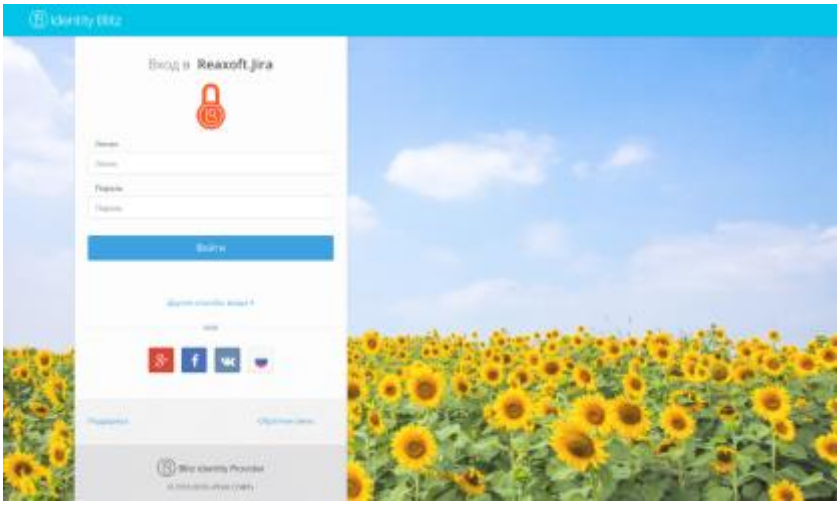

Рисунок 86 – Пример страницы входа с social login и дополнительным футером

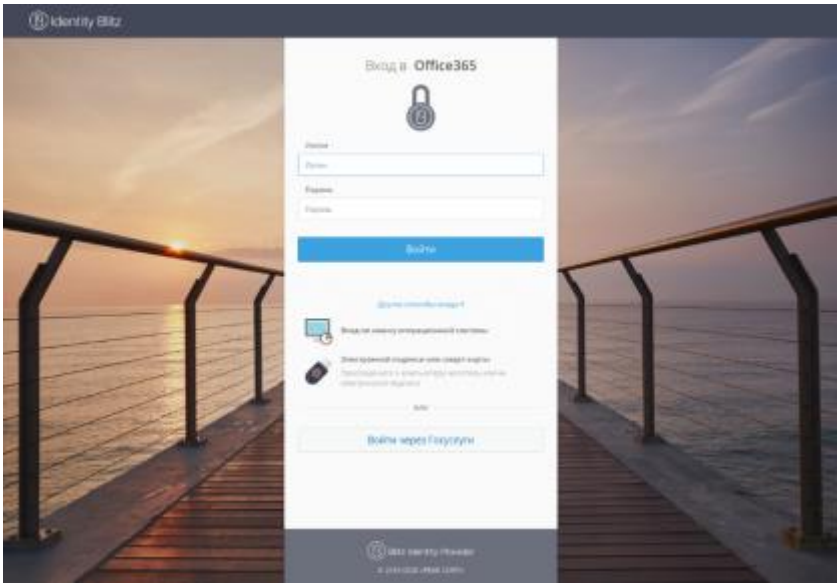

Рисунок 87 – Пример страницы входа в темном интерфейсе и с режимами входа по электронной подписи, сеансу операционной системы или через ЕСИА

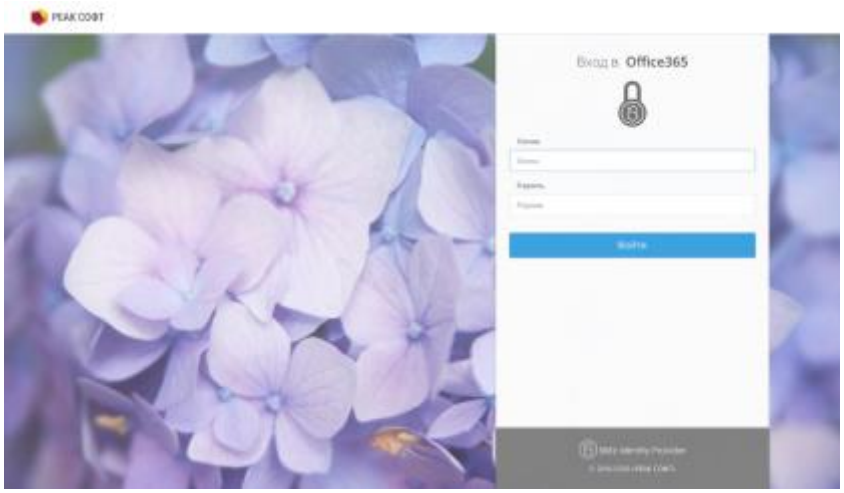

Рисунок 88 – Пример страницы входа в светлом интерфейсе, с кастомным логотипом в заголовке страницы и без специальных режимов входа

2.12.2. Создание и изменение новых шаблонов с помощью конструктора

Blitz Identity Provider позволяет настроить разный вид страниц входа для случая входа пользователя в различные подключенные приложения. Для этого необходимо создавать новые шаблоны входа – проще всего это сделать на базе существующего default-шаблона, нажав на кнопку «Копировать». После этого будет создан новый шаблон, который можно редактировать с помощью конструктора.

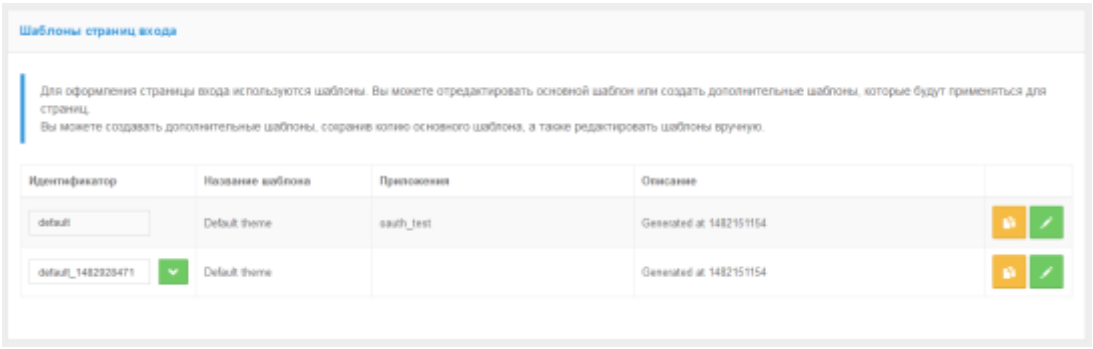

Рисунок 89 – Создание нового шаблона на базе существующего основного шаблона

Чтобы новый шаблон использовался при входе в некоторое приложение, необходимо в разделе *Приложения* перейти к редактированию нужного приложения и выбрать требуемый шаблон страниц.

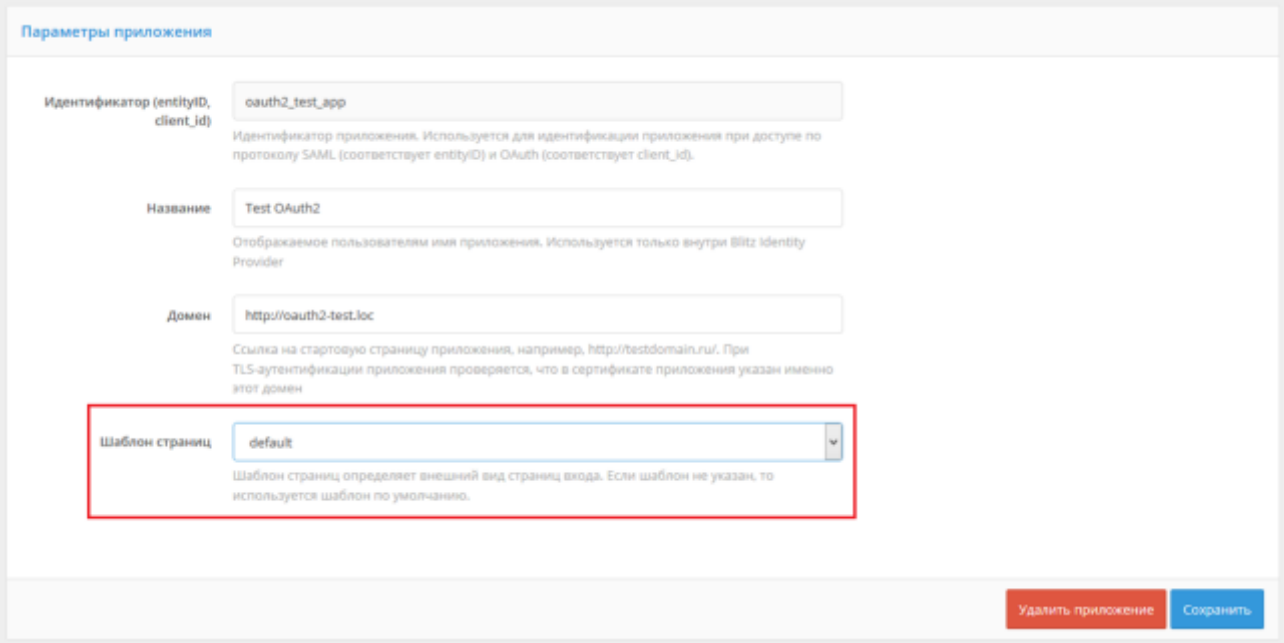

Рисунок 90 – Назначение шаблона страницы входа приложению

2.12.3. Создание и изменение новых шаблонов в ручном режиме

Для Enterprise-редакции можно настроить вид страницы входа под индивидуальные требования организации, т.е. нет необходимости ограничиваться только возможностями конструктора.

Каждый шаблон страницы входа представляет собой zip-архив. Все шаблоны размещены в директории:

```
\assets\themes
```

Самый простой способ перейти к ручному редактированию шаблона – выполнить следующие шаги:

- создать копию существующего шаблона (например, default-шаблона), нажав в консоли кнопку  ;
- перейти в соответствующую директорию с шаблонами;
- распаковать архив с только что созданным шаблоном;
- отредактировать файл *meta.conf*, содержащийся в архиве, удалив параметр builder (рис. 91);
- обратно заархивировать файлы шаблона, убедившись, что файл *meta.conf* находится в корневой директории.

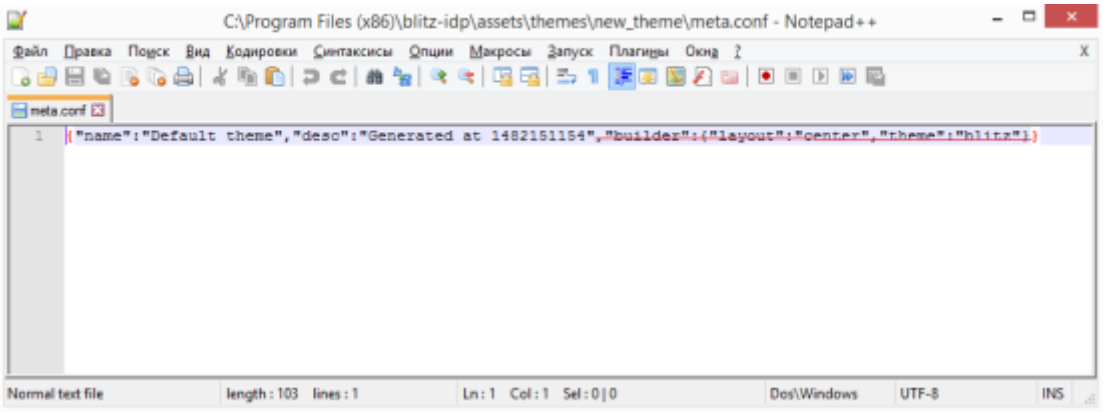

Рисунок 91 – Удаление параметра builder из файла meta.conf

После выполнения этих шагов появится возможность редактирования темы в ручном режиме. Помимо стандартных полей, описывающих саму тему, доступен блок «Шаблон страниц». Он позволяет создать / изменить шаблон – текстовый файл, который компилируется с помощью шаблонизатора Twirl[26].

Шаблон должен иметь сигнатуру:

```
@(headers: Html, form: Html, scripts: Html, pathAssets: String)(implicit request: RelyingPartyRequest[_], messages: Messages)
```

В качестве параметров при создании шаблона следует использовать:

- headers – html-код заголовка страницы, который необходимо расположить в теге head;
- form – html-код основной формы, который необходимо расположить в теге body;
- scripts – html-код с javasctips, необходимый для корректной работы формы, который необходимо расположить в теге body;
- pathAssets – контекстный путь к ресурсам шаблона.

Листинг простейшего шаблона приведен ниже:

```
@(headers: Html, form: Html, scripts: Html, path: String)(implicit request: RelyingPartyRequest[_], messages: Messages)

<!DOCTYPE html>
<html>

<head>
   @headers

</head>


<body>
```

[26] См.: https://www.playframework.com/documentation/2.5.x/ScalaTemplates

```
  <div id="main">
    <section id="content_wrapper">
          @form
    </section>
    <div>
      <div>
        @Html(messages("author.copyright"))
      </div>
    </div>
  </div>
  @scripts

 </body>

</html>
```

При использовании такого шаблона страница входа будет иметь вид, приведенный на рис. 92.

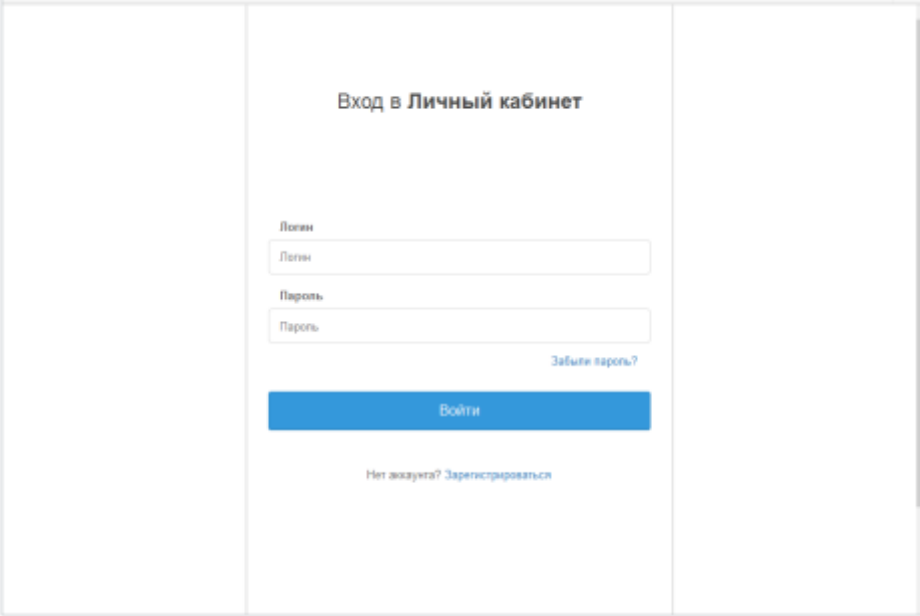

Рисунок 92 – Внешний вид простейшей страницы входа

При формировании шаблона страницы входа имеется возможность использовать ресурсы – например, таблицы стилей или рисунки.

Для их загрузки следует использовать блок «Ресурсы» внешнего вида страницы, который позволяет загрузить необходимые файлы в zip-архиве. Чтобы соответствующие файлы были доступны, их следует размещать в директории архива с названием *assets*. Необходимые ресурсы также можно вручную включить в состав исходного zip-архива с шаблоном страницы.

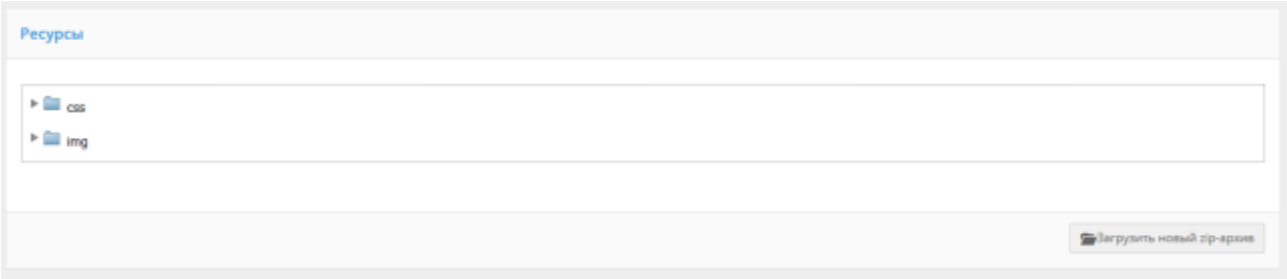

Рисунок 93 – Внешний вид: доступные ресурсы

2.13. Административные и прочие настройки

В данном разделе описаны настройки, не доступные из консоли управления. Их изменение предполагает запуск скриптов, входящих в поставку Blitz Identity Provider, или редактирование конфигурационных файлов.

2.13.1. Добавление администраторов и изменение паролей

Добавление администраторов и изменение паролей осуществляется посредством редактирования файла *credentials*, размещенного в директории *conf*.

Для добавления пользователя следует добавить следующую структуру в этот файл:

```
{
        "pswdHash" : "123456",
        "username" : "test"
    }
```

Здесь *username* – это логин администратора, *pswdHash* – его пароль. После перезапуска Blitz Identity Provider пароль будет зашифрован. Пример файла *credentials*:

```
{
  "users" : [
    {
      "pswdHash" :
"{SHA256}A8r5o3MkRZQnUeW2KEg7B3Fk3606cu47FVtAqIn39J+g8gFA9MI0YUm3yWt1UrZHprkcIJJF7BXk0NCECbYtvw"
,
      "username" : "admin"
    },
    {
      "pswdHash" :
"{SHA256}ZzfFerFp3MtzyaMs2Ymv4YyprOcukYQopYZCbCbV8WW1hy5zBZEb6dpLmy5o4kiHseUmMTO0rUwAEc4SqhcYz
g",
      "username" : "test"
    }
  ]
}
```

Для изменения пароля следует отредактировать параметр pswdHash и перезапустить систему.

2.13.2. Мультиязычность и кастомизация текстовых сообщений

2.13.2.1. Мультиязычность

Веб-интерфейс Blitz Identity Provider поддерживает мультиязычность. По умолчанию предусмотрено два языка – русский и английский.

Переключение языка осуществляется посредством изменения основного языка ввода (языка отображения веб-страниц) в используемом браузере. Например, для изменения языка в браузере Chrome нужно выполнить шаги:

- перейти к настройкам браузера (chrome://settings/);
- выбрать пункт «Показать дополнительные настройки»;
- нажать на кнопку «Изменить языковые настройки»;
- переместить нужный язык на первое место в списке (рис. 94).

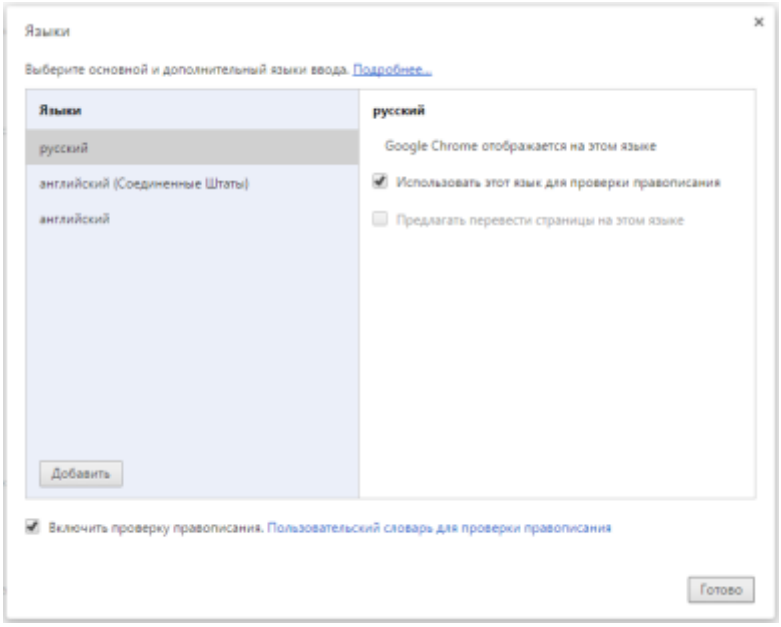

Рисунок 94 – Настройка языка для браузера Chrome

Для изменения языка в браузере Firefox нужно выполнить шаги:

- перейти к настройкам браузера (about:preferences);
- выбрать раздел «Содержимое» настроек;
- в подразделе «Языки» нажать на кнопку «Выбрать»;
- переместить нужный язык на первое место в списке:

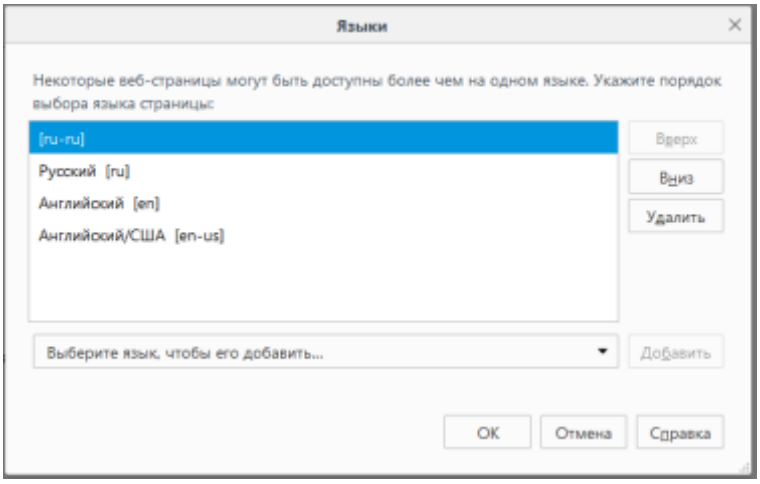

Рисунок 95 – Настройка языка для браузера Firefox

2.13.2.2. Модификация текстовых сообщений веб-интерфейса

Blitz Identity Provider позволяет менять текстовые строки, используемые в интерфейсе системы. Для этого необходимо отредактировать файл *messages*, размещенный в директории *conf\custom_messages*, добавив строку вида «параметр=значение», где параметр – идентификатор текстовой строки, а значение – необходимый текст. Например, следующая строка отвечает за текст на форме регистрации, где размещена ссылка на условия использования:

```
reg.page.reg.action.agreement=Нажимая на кнопку &laquo;Зарегистрироваться&raquo; вы соглашаетесь с <a href="{0}" target="_blank">условиями использования</a>
```

Для корректного отображения файл должен быть сохранен в кодировке UTF-8.

При необходимости изменить английский язык следует добавить в указанную директорию файл *messages.en* и изменить в нем необходимые файлы.

Далее в табл. 1 представлен перечень некоторых строк, используемых на страницах входа, Личного кабинета, регистрации и восстановления доступа.

Таблица 1

Примеры некоторых текстовых сообщений

Идентификатор строки	Текст строки
Страница входа	
auth.page.login.title	Вход в
login.methods.password.head.title	Войти
auth.methods.link	Другие способы входа
auth.registration.link	Нет аккаунта? Зарегистрироваться
login.methods.password.recovery.link	Забыли пароль?
Регистрация	
reg.page.title	Регистрация в

reg.page.password.blockquote	Пароль должен состоять не менее чем из 8 символов. Рекомендуется, чтобы пароль состоял из прописных и строчных букв и имел хотя бы одну цифру. Не применяйте пароли, используемые для других сайтов, и пароли, которые можно легко подобрать.
reg.page.reg.action.agreement	Нажимая на кнопку «Зарегистрироваться» вы соглашаетесь с условиями использования
setPswd.page.agreement	Нажимая на кнопку «Зарегистрироваться» вы соглашаетесь с условиями использования
reg.page.btn.reg	Зарегистрироваться
Восстановление доступа	
recovery.page.title	Восстановление доступа в
recovery.page.btn.reg	Восстановить доступ
Личный кабинет	
profile.sidebar.menu.data	Основные данные
page.profile.data.metaTitle	Основные данные
profile.sidebar.menu.events	События
profile.sidebar.menu.security	Безопасность
page.profile.data.account.title	Данные учетной записи
page.profile.data.events.title	Последние события
page.profile.security.password.metaTitle	Пароль
page.profile.security.authn.metaTitle	Подтверждение входа
page.profile.security.social.metaTitle	Социальные сети
page.profile.security.password.altPwd.info	Периодически меняйте свой пароль. Рекомендуется использовать пароль из прописных, строчных букв и хотя бы с одной цифрой. Не применяйте пароли, используемые для других сайтов, и пароли, которые можно легко подобрать.
page.profile.security.authn.toggle.info	Защитите учетную запись, настроив способ подтверждения входа (второй фактор аутентификации). Тогда после ввода пароля потребуется ввести код подтверждения.
page.authn.setting.sms.row	Доставка SMS-кодов подтверждения
page.authn.setting.hotp.row	Коды подтверждения из специального генератора
page.authn.setting.totp.row	Коды подтверждения из мобильного приложения
page.hotp.attach.prompt	Введите серийный номер устройства, генерирующего разовые пароли.
page.totp.attach.prompt	Используйте ваш смартфон для генерирования разовых

	паролей.
page.totp.attach.set.prompt	Установите специальное приложение
page.profile.security.social.fedAccounts.title	Привязанные учетные записи
page.profile.security.social.fedAccounts.notFound	Учетные записи не привязаны

2.13.2.3. Модификация шаблонов писем и SMS-сообщений

Шаблоны писем представляют собой текстовые строки, сохраняемые аналогично обычным строкам в веб-интерфейсе. Их изменение производится аналогичным образом (см п. 2.13.2.2 документа).

В таблице ниже представлен перечень шаблонов писем, отправляемых по электронной почте. В шаблонах можно использовать HTML-форматирование.

Таблица 2

Шаблоны писем, отправляемых по электронной почте

Идентификатор шаблона	Пример шаблона
Регистрация пользователя	
message.email.subject.registration	Успешная регистрация
message.email.subject.registration.with.generated.pswd	Успешная регистрация
message.email.subject.activation	Подтвердите регистрацию
message.email.subject.activation.by.link	Подтвердите регистрацию
message.email.body.registration	Уважаемый пользователь, Вы зарегистрировали учетную запись.
message.email.body.registration.with.generated.pswd	Уважаемый пользователь, Вы успешно зарегистрировали учетную запись. Ваши данные: Email: {0} Пароль: {1}
message.email.body.activation	Уважаемый пользователь, От вашего имени подана заявка на регистрацию учетной записи. Для завершения регистрации вам необходимо подтвердить адрес электронной почты. Для этого введите в форме регистрации код подтверждения {1}, либо просто перейдите по ссылке
message.email.body.activation.by.link	Уважаемый пользователь, От вашего имени подана заявка на регистрацию учетной записи. Для завершения регистрации вам необходимо подтвердить адрес электронной почты. Для этого просто перейдите по ссылке
Восстановление доступа	
message.email.subject.recovery.confirmation	Подтвердите восстановление пароля
message.email.body.recovery.confirmation	Уважаемый пользователь, От вашего имени подана заявка на восстановление пароля. Для того

	чтобы восстановить доступ к учетной записи вам необходимо подтвердить адрес электронной почты. Для этого введите в форме восстановления код подтверждения {1}, либо просто перейдите по ссылке
Изменение атрибута из Личного кабинета	
message.email.subject.profile.confirmation	Подтвердите изменение атрибута
message.email.body.profile.confirmation	Вы запросили смену "{0}" атрибута {1}. Чтобы подтвердить операцию, перейдите по ссылке. Или введите код подтверждения: {3} Если вы не инициировали это действие, возможно, ваша учетная запись была взломана.

В таблице ниже представлен перечень шаблонов писем, отправляемых в виде SMS-сообщений.

Таблица 2

Шаблоны писем, отправляемых в виде SMS-сообщений

Идентификатор шаблона	Пример шаблона
Вход в систему (второй фактор в виде SMS-кода)	
messaging.template.second.factor.sms	Kod podtverzhdeniya: {0}
Изменение атрибута из Личного кабинета	
page.profile.mobile.cfm.code	Kod podtverzhdeniya: {0}

2.13.2.4. Модификация сообщений и шаблонов в разрезе приложений

Возможно изменение всех текстовых сообщений и шаблонов таким образом, чтобы использовались специфические тексты/шаблоны для разных приложений. Таким образом можно, например, брендировать письма, отправляемые при регистрации на разных сайтах, подключенных к одной установке Blitz Identity Provider, или давать ссылку на скачивание различных правил использования ресурса.

Для привязки набора шаблонов к конкретному приложению следует выполнить шаги:

1. Создать экземпляр файла с текстами, который будет использоваться исключительно для данного приложения. Для этого в директории *conf\custom_messages* создать текстовый файл *messages.ru-123456* (*messages.en-123456*) для данного приложения, где 123456 – последовательность из 5-8 символов (допускаются как цифры, так и буквы латинского алфавита).
2. Отредактировать файл *messages.ru-123456* (*messages.en-123456*), добавив в него специфические строки для данного приложения (подробнее см. п. 2.13.2.2). Все остальные строки будут взяты из базы строк по умолчанию.

3. Отредактировать файл *blitz.conf*, размещенный в директории *conf*, следующим образом:
   - в разделе *apps* файла найти идентификатор приложения, который должен использовать созданный файл шаблона;
   - добавить параметр вида "lang-variant" : "123456", где 123456 – использованная для маркировки шаблона последовательность символов. Пример:

```
"demo-application" : {
  "domain" : "http://testdomain.ru",
  "lang-variant" : "123456",
  "name" : "test",
  "oauth" : {
    "autoConsent" : false,
    "clientSecret" : "1234567890",
    "defaultScopes" : [],
    "enabled" : true,
    "redirectUriPrefixes" : [
      "http://localhost"
    ]
  },
  "theme" : "default"
},
```

После этого при входе в данное приложение будет использоваться специально созданный файл сообщений.

2.13.3. Изменение правил использования

По умолчанию правила использования размещены в виде pdf-файла *user_agreement_ru.pdf* (русская версия) и *user_agreement_en.pdf* (версия, доступная при скачивании при работе в английской версии). Данные файлы размещены в архиве *assets.zip*, расположенном в директории *assets* установки Blitz Identity Provider.

В свою очередь, правила использования размещены в директории documents\user_agreement данного архива.

Для изменения правил использования следует распаковать архив, заменить файлы *user_agreement_ru.pdf* и *user_agreement_en.pdf* и заархивировать архив с сохранением исходной структуры[27].

Также возможно изменить ссылку на правила использования. Для этого следует отредактировать строку reg.page.reg.action.agreement и setPswd.page.agreement (см. раздел 2.13.2.2). Такой способ рекомендуется применять, если правила использования размещены на внешнем ресурсе, например, в виде отдельной веб-страницы.

[27] Для подготовки правил можно использовать предложенный шаблон. Его Word-версию можно загрузить по ссылке: https://identityblitz.ru/wp-content/uploads/2015/07/agreement_template.docx

2.13.4. Изменение домена

Для изменения домена Blitz Identity Provider Standard Edition в среде Linux следует выполнить следующий скрипт:

```
/usr/share/blitz-idp/scripts/change_domain.sh
```

Для изменения домена Blitz Identity Provider Standard Edition в среде Windows следует выполнить скрипт:

```
blitz-idp\scripts\change_domain.cmd
```

Для изменения домена Blitz Identity Provider Enterprise Edition необходимо отредактировать файлы *blitz.conf, relying-party.xml, idp-metadata.xml,* заменив в них старое значение домена на новое.