Система управления доступом
к информационным ресурсам города Москвы (СУДИР)

Методика подключения к СУДИР

(внутренний контур)

Листов 29

Москва, 2019

Аннотация

Настоящий документ определяет общий порядок использования внутреннего контура автоматизированной информационной системы «Система управления доступом к информационным ресурсам города Москвы» (далее – СУДИР, Система), а также порядок и требования, которые необходимо выполнить на стороне информационной системы для подключения к СУДИР.

Содержание

Введение

СУДИР берет на себя решение задачи идентификации/аутентификации пользователя, взаимодействует с ЕСК (Active Directory) для получения данных аутентифицированного пользователя. В случае если ПК пользователя входит в домен, то в СУДИР также настроена интеграция с доменом для обеспечения сквозной идентификации без необходимости пользователем повторно вводить логин/пароль на странице входа СУДИР.

В данном документе содержится техническая информация о процессе подключения информационной системы, приложения ОИВ города Москвы (далее – приложения) к СУДИР. Этот процесс включает следующие шаги:

1. Выбор протокола взаимодействия с СУДИР. Рекомендации по выбору протокола приведены в главе 1 настоящего документа.
2. Подача заявки на подключение к СУДИР. Для выполнения этого шага необходимо руководствоваться Регламентом подключения к СУДИР (внутренний контур)[1]. Техническая информация для заполнения заявки приведена в главе 2 настоящего документа.
3. Конфигурирование приложения или его доработка для подключения к СУДИР по выбранному протоколу подключения. В главе 3 даются рекомендации по настройке приложений и по их доработке для реализации одного из протоколов взаимодействия с СУДИР.

[1] Актуальные версии документов «Методика подключения к СУДИР (внутренний контур)» и «Регламент подключения к СУДИР (внутренний контур)» размещены на ресурсе «Технологический портал СУДИР» и доступны по адресу: https://sudir.mos.ru/support

1. Выбор протокола взаимодействия

Для интеграции приложения с СУДИР для проведения идентификации и аутентификации пользователя следует выбрать один из протоколов взаимодействия:

- OpenID Connect 1.0 (OIDC)[2]/OAuth 2.0[3] – наиболее современный SSO-протокол, изначально ориентированный на работу с веб-приложениями в сети Интернет. Если создается новое приложение, то гораздо проще подключить его к СУДИР именно по OIDC/OAuth 2.0.
- SAML 1.0/1.1/2.0[4] – широко распространенный SSO-протокол, позволяющий подключить практически любое популярное корпоративное ПО или облачное приложение к сервису входа. Подключаемое приложение возможно имеет встроенную поддержку SAML или такая поддержка может быть добавлена в качестве дополнительной опцией или через установку интеграционного коннектора/плагина.

Выбор протокола во многом зависит от того, какое приложение требуется подключить к СУДИР:

- если приложение поддерживает один из SSO-протоколов, то стоит подключать его с использованием данного протокола;
- если предложение не поддерживает протоколы, то следует провести его доработку – в этом случае проще поддержать взаимодействие по OIDC;
- если приложение только создается, то на этой стадии целесообразно поддержать один из SSO-протоколов – поддержку OIDC реализовать проще, однако при использовании доступных библиотек SAML можно использовать и этот протокол.

В таблице ниже приведены некоторые особенности протоколов OIDC и SAML.

Таблица 1 – Особенности протоколов подключения

	OIDC/OAuth 2.0	**SAML 1.0/1.1/2.0**
Способ обеспечения доверия между приложением и СУДИР	Секрет приложения (обычно в виде строки), известный СУДИР	Электронная подпись. И запросы на аутентификацию, и ответы – это подписанные XML документы
Способ взаимодействия	Через браузер пользователя проходит инициируется аутентификация. Для завершения аутентификации серверная часть	Обычно запрос на аутентификацию и ответ проходят через браузер пользователя. Приложение и СУДИР могут не иметь сетевой

[2] http://openid.net/specs/openid-connect-core-1_0.html

[3] https://tools.ietf.org/html/rfc6749

[4] https://www.oasis-open.org/standards#samlv2.0

	приложения должна сформировать HTTP-запрос в адрес СУДИР	связности
Получение сведений о пользователе	Два способа получения данных о пользователе: – Приложение обращается к REST-сервису СУДИР и получает данные о пользователе в формате json. Приложение может продолжать получать данные о пользователе, даже когда пользователь завершает свою онлайн-сессию – Приложение получает данные пользователя из маркера идентификации (id_token в форме JWT), полученного от СУДИР по результатам аутентификации	Данные о пользователе содержатся в ответе на запрос на аутентификацию в формате XML. Приложение может получать от СУДИР данные только в момент входа пользователя
Поддерживаемые приложения	Веб-приложения и мобильные приложения	Веб-приложения

OIDC позволяет реализовать все основные сценарии SAML, но при этом используется более простой JSON/REST-протокол. Существенное преимущество OIDC – поддержка мобильных приложений.

2. Техническая информация по подготовке заявки подключения к СУДИР

2.1. Подключение приложения по протоколу OIDC/OAuth 2.0

Аутентификация в терминологии OIDC/OAuth 2.0 является результатом взаимодействия трех сторон:

- сервис авторизации (Authorization Server) или поставщик ресурса (Resource Server), в качестве которых выступает СУДИР;
- система-клиент (Client), в качестве которого выступает приложение, которое запрашивает доступ ресурсу (аутентификационной информации и данным пользователя);
- владелец ресурса (Resource Owner), в качестве которого выступает пользователь, так как в ходе аутентификации он разрешает доступ к данным о себе.

Приложение как система-клиент должно зарегистрироваться в СУДИР и получить следующие данные, необходимые для формирования запросов на проведение аутентификации:

- идентификатор приложения (client_id);
- секрет приложения (client_secret).

Для регистрации необходимо направить заявку согласно Регламенту подключения к СУДИР. Заявка включает в себя следующие специфические для данного протокола поля:

1. Разрешенные адреса возврата (списки redirect_uri и post_logout_redirect_uri).
2. Перечень запрашиваемых разрешений (список scope), необходимость в получении refresh_token.
3. Перечень дополнительных атрибутов, которые СУДИР должен добавить в маркер идентификации (дополнительные атрибуты для передачи в составе id_token).

2.1.1. Определение разрешенных адресов возврата

Запрос на проведение аутентификации содержит ссылку возврата при авторизации (redirect_uri), куда должен быть возвращен пользователь после прохождения аутентификации. Допустимые ссылки возврата должны соответствовать зарегистрированным в СУДИР разрешенным префиксам. Как правило, префиксы ссылок возврата – это доменные имена приложений. Например, если после проведения аутентификации требуется вернуть пользователя на https://domain.ru/callback, то в качестве адреса возврата следует указать в заявке https://domain.ru. Если в запросе на аутентификацию указана ссылка возврата и она не соответствует ни одному из указанных префиксов, то в аутентификации будет отказано.

Запрос на проведение логаута содержит ссылку возврата при логауте (post_logout_redirect_uri), куда должен быть возвращен пользователь после инициированного приложением логаута. Допустимые ссылки возврата должны соответствовать зарегистрированным в СУДИР разрешенным префиксам. Если в запросе на логаут указана ссылка возврата и она не соответствует ни одному из указанных префиксов, то в аутентификации будет отказано.

2.1.2. Определение запрашиваемых разрешений

Разрешения (scope в терминологии OIDC/OAuth 2.0) определяют, какие данные и права на выполнение каких операций получит приложение по результатам аутентификации. Перечень предусмотренных в СУДИР разрешений представлен в таблице 2.

Таблица 2 – Доступные разрешения (scope)

№	Разрешение	Описание	Состав получаемых атрибутов
1.	openid	Техническое разрешение, указывающее на то, что аутентификация проводится согласно спецификации OIDC 1.0	При запросе этого scope СУДИР предоставляет приложению id_token. Из id_token приложение может получить нужные ему атрибуты пользователя (см. главу 3.1.3.4.2.).
2.	profile	Основные данные профиля пользователя	− objectGUID - уникальный идентификатор − uid – уникальный идентификатор (userPrincipalName из ЕСК) − logonname – логин входа в домен (sAMAccountName из ЕСК) − surname – фамилия − firstname – имя − middlename – отчество − email – служебный адрес электронной почты
3.	employee	Служебные данные сотрудника	− company – название организации сотрудника − department – подразделение организации, где работает сотрудник − workphone – служебный номер телефона сотрудника − position – должность − email – служебный адрес электронной почты
4.	userinfo	Основные данные профиля пользователя (названия атрибутов по спецификации OIDC)	− sub – уникальный идентификатор (uid) пользователя − family_name – фамилия − given_name – имя − email – служебный адрес электронной почты

2.1.3. Определение дополнительных атрибутов для включения в id_token

Обычно нет необходимости получать атрибуты пользователя непосредственно из маркера идентификации (id_token) – более простым и рекомендуемым способом является получение данных пользователя через вызов REST-сервиса (см. главу 3.1.3.4.1).

Если все же необходимо получать сведения о пользователе в составе id_token (см. главу 3.1.3.4.2), то в заявке на подключение необходимо перечислить необходимые для включения в маркер идентификации дополнительные атрибуты. Доступные для получения атрибуты приведены в таблице 3.

Таблица 3 – Возможные дополнительные атрибуты пользователя в id_token

№	Атрибут	Описание
1.	logonname	Логин пользователя в домене (sAMAccountName из ЕСК)
2.	surname	Фамилия
3.	firstname	Имя
4.	middlename	Отчество
5.	email	Служебный адрес электронной почты
6.	company	Название организации сотрудника
7.	department	Подразделение организации, где работает сотрудник
8.	workphone	Служебный номер телефона сотрудника
9.	position	Должность

2.2. Подключение приложения по протоколу SAML

Аутентификация в терминологии SAML является результатом взаимодействия трех сторон:

- поставщик идентификации (Identity Provider), в качестве которого выступает СУДИР;
- поставщик услуги (Service Provider), в качестве которого выступает подключаемое приложение;
- браузер пользователя (User Agent).

Приложение (поставщик услуг) должно быть зарегистрировано в СУДИР. Для этого необходимо направить заявку согласно Регламенту подключения к СУДИР. Заявка включает XML-файл с метаданными поставщика услуг или значения параметров, необходимые для самостоятельной подготовки метаданных администраторами СУДИР.

2.2.1. Подготовка метаданных поставщика услуг

Метаданные поставщика услуг описывают настройки подключения приложения к СУДИР (например, URL конечных точек приложения, ключи для проверки ЭП). Для описания метаданных используется язык XML[5].

Метаданные должны быть подготовлены по результатам выполнения работ по добавлению поддержки протокола (см. п. 3.2 настоящего документа).

Если приложение является готовым ПО, поддерживающим SAML, то метаданные должны быть получены согласно документации на это ПО. Обычно такое ПО предоставляет URL, по которому может быть получены метаданные.

[5] Подробнее про метаданные SAML см. здесь: http://docs.oasis-open.org/security/saml/v2.0/saml-metadata-2.0-os.pdf

Если ПО подключаемого приложение не предусматривает выгрузку метаданных, но в документации на ПО описаны параметры, которые должны быть настроены для подключения приложения, то можно указать эти параметры, так, чтобы метаданные на их основе были самостоятельно подготовлены Администратором СУДИР. В этом случае необходимо указать следующие параметры:

1. Идентификатор поставщика услуг (EntityID) – следует указать, только если приложению необходим конкретный EntityID. Иначе EntityID будет самостоятельно присвоен Администратором СУДИР.
2. Сертификат[6] открытого ключа приложения (поставщика услуг) – должен быть указан только в случае, если приложение подписывает SAML-запрос при отправке к СУДИР. Должны использоваться ключи RSA-2048. Допустимо использовать самоподписанные сертификаты с длительным сроком действия.
3. URL для приема от СУДИР SAML-ответа – приложение должно предоставлять обработчик, осуществляющий прием от СУДИР SAML-ответов с результатами входа. Обычно эта настройка приложения называется Assertion Consumer Service.
4. URL для приема от СУДИР запроса на логаут – опциональная настройка. Если приложение поддерживает единый логаут, то оно может предоставлять обработчик единого логаута. Обычно эта настройка приложения называется Single Logout Service Location.
5. URL для перенаправления пользователя в приложение после успешного логаута – опциональная настройка. Если приложение поддерживает единый логаут и может инициировать единый выход, то оно может предоставлять URL для возврата пользователя после логаута. Обычно эта настройка приложения называется Single Logout Service Response Location.
6. Перечень запрашиваемых атрибутов (SAML Assertion).
7. Признак необходимости передачи атрибутов в зашифрованном виде[7].

2.2.2. Перечень запрашиваемых атрибутов (SAML Assertion)

В метаданных указывается перечень атрибутов пользователя (SAML Assertion). Предусмотренные в СУДИР атрибуты пользователя приведены в таблице 4.

[6] Сертификат поставщика услуг – это не тоже самое, что SSL/TLS-сертификат подключаемого веб-сайта.

[7] Атрибуты в SAML-сообщении всегда передаются подписанными. Включать шифрование атрибута целесообразно, если пользователь не должен иметь возможности прочитать значение атрибута

Таблица 4 – Доступные атрибуты пользователя (SAML Assertion)

№	Атрибут	Описание
1.	logonname	Логин пользователя в домене (sAMAccountName из ЕСК)
2.	surname	Фамилия
3.	firstname	Имя
4.	middlename	Отчество
5.	email	Служебный адрес электронной почты
6.	company	Название организации сотрудника
7.	department	Подразделение организации, где работает сотрудник
8.	workphone	Служебный номер телефона сотрудника
9.	position	Должность

2.3. Добавление элементов дизайна приложения на странице входа СУДИР

СУДИР позволяет поместить элементы дизайна приложения на страницу входа СУДИР. При желании создать для подключаемой системы персонифицированную страницу входа нужно отметить соответствующий пункт в заявке на подключение. В этом случае при обработке заявки Администратор СУДИР вместе с данными о зарегистрированных параметрах подключения вышлет шаблон оформления страницы входа.

Шаблон оформления страницы входа представляет собой zip-архив, внутри которого записаны HTML каркаса страницы входа и используемые на странице таблица стилей, изображения, JavaScript-обработчики.

Ответственные за подключаемую систему могут скорректировать присланное содержимое, адаптировав тем самым верстку страницы входа под требования дизайна подключаемой системы. Подготовленный архив темы страницы входа нужно передать Администратору СУДИР для загрузки и проверки в тестовом контуре. Если полученный внешний вид страницы входа будет приемлемым, то Администратор СУДИР перенесет настройки внешнего вида в ПРОД-контур СУДИР.

3. Настройка и доработка приложений для интеграции с СУДИР

3.1. Подключение приложения по протоколу OIDC/OAuth 2.0

3.1.1. Данные для подключения по результатам рассмотрения заявки

По результатам исполнения заявки на подключение к СУДИР будет предоставлена следующая информация:

- идентификатор, присвоенный приложению в СУДИР (client_id);
- секрет приложения (client_secret);
- зарегистрированные для приложения URL возврата при авторизации;
- зарегистрированные для приложения URL возврата при логауте;
- зарегистрированные для приложения разрешения (scope).

Приложение взаимодействует с сервисами СУДИР, используя следующие адреса:

- URL для проведения авторизации и аутентификации:
  - https://sudir-test.mos.ru/blitz/oauth/ae (тестовая среда)
  - https://sudir.mos.ru/blitz/oauth/ae (продуктивная среда)
- URL для получения и обновления маркера доступа:
  - https://sudir-test.mos.ru/blitz/oauth/te (тестовая среда)
  - https://sudir.mos.ru/blitz/oauth/te (продуктивная среда)
- URL для получения данных пользователя:
  - https://sudir-test.mos.ru/blitz/oauth/me (тестовая среда)
  - https://sudir.mos.ru/blitz/oauth/me (продуктивная среда)
- URL для получения данных о маркере доступа:
  - https://sudir-test.mos.ru/blitz/oauth/introspect (тестовая среда)
  - https://sudir.mos.ru/blitz/oauth/introspect (продуктивная среда)
- URL для выполнения логаута (https://redirect_uri – адрес, по которому должен быть возвращен пользователь после логаута):
  - https://sudir-test.mos.ru/blitz/login/logout?post_logout_redirect_uri=https://redirect_uri (тестовая среда)
  - https://sudir.mos.ru/blitz/login/logout?post_logout_redirect_uri=https://redirect_uri (продуктивная среда)

Все эти URL, а также дополнительные сведения, размещены по адресу динамически обновляемых настроек (метаданных) каждой среды СУДИР[8]:

- https://sudir-test.mos.ru/blitz/oauth/.well-known/openid-configuration (тестовая среда)

[8] Созданы в соответствие с: https://tools.ietf.org/html/draft-ietf-oauth-discovery-10

– https://sudir.mos.ru/blitz/oauth/.well-known/openid-configuration (продуктивная среда)

Разработчики приложений могут не прописывать все указанные URL в конфигурации своего приложения, а использовать в настойках единую ссылку на метаданные СУДИР.

3.1.2. Готовые библиотеки

Для интеграции приложения с СУДИР по протоколу OIDC/OAuth 2.0 можно использовать одну из множества готовых OAuth 2.0 библиотек[9] или реализовать взаимодействие самостоятельно.

Далее рассмотрены шаги, которые должен выполнить сайт, чтобы провести аутентификацию пользователя по протоколу OIDC/OAuth 2.0 и получить данные о нем.

3.1.3. Разработка взаимодействия приложения с СУДИР по OIDC/OAuth 2.0

3.1.3.1. Общие сведения

Взаимодействие приложения с СУДИР по OIDC включает в себя следующие шаги[10]:

– приложение отправляет запрос на аутентификацию в адрес СУДИР;
– СУДИР аутентифицирует пользователя;
– СУДИР получает согласие пользователя на проведение аутентификации и на передачу идентифицирующей информации о нем в приложение;
– СУДИР перенаправляет пользователя обратно в приложение и передает авторизационный код;
– приложение формирует запрос с использованием авторизационного кода на получение маркера идентификации, маркера обновления, маркера доступа;
– приложение получает ответ, содержащий необходимые маркеры;
– приложение запрашивает данные пользователя по маркеру доступа. При необходимости приложение может провести валидацию маркера идентификации и извлечь из этого маркера идентификатор пользователя и дополнительные атрибуты.

На схемах ниже представлены процессы получения авторизационного кода, маркеров, данных пользователя.

[9] https://oauth.net/code/#client-libraries

[10] Данный процесс совпадает с моделью авторизации приложения Authorization Code Grant, предусмотренной спецификацией OAuth 2.0.

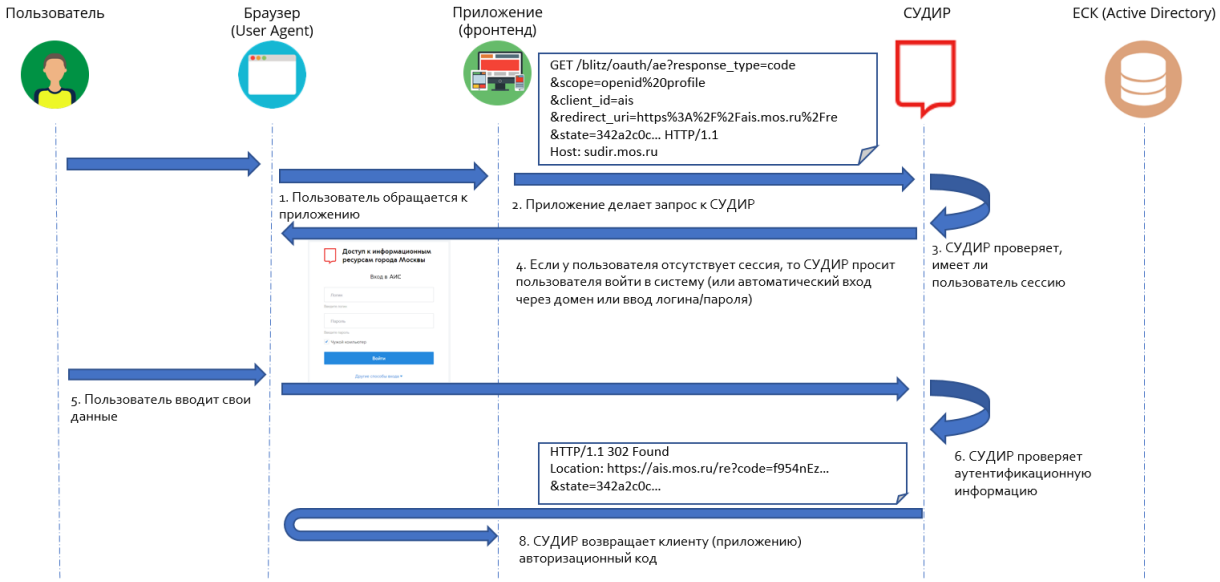

Рисунок 1 – Процесс получения авторизационного кода

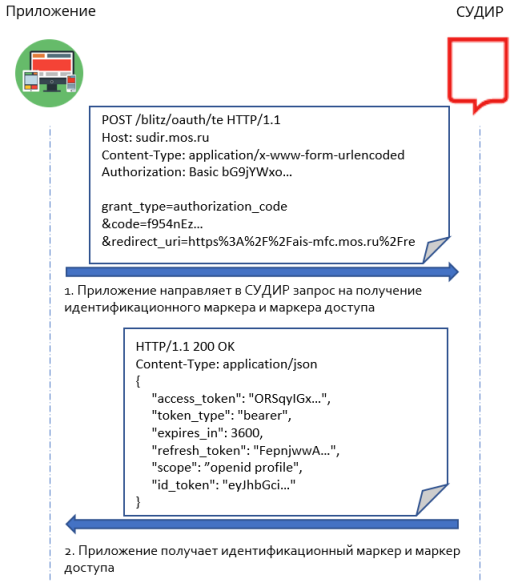

Рисунок 2 – Процесс получения маркера доступа и маркера идентификации

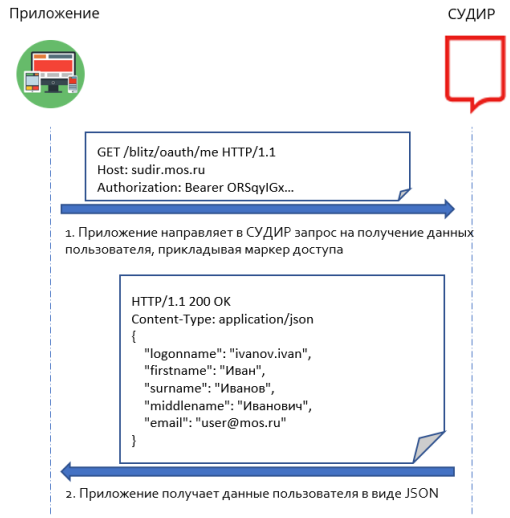

Рисунок 3 – Процесс получения данных пользователя

3.1.3.2. Запрос на аутентификацию (получение авторизационного кода)

Для проведения аутентификации приложение должно направить пользователя на URL для проведения авторизации и аутентификации, передав в качестве параметров:

- scope – запрашиваемые разрешения (scope), для проведения аутентификации должно быть передано разрешение openid и необходимые дополнительные scope для получения данных пользователя, например, profile (при запросе нескольких scope они передаются одной строкой и отделяются друг от друга пробелом);
- response_type – тип ответа (принимает значение “code”);
- redirect_uri – ссылка для возврата пользователя в приложение, ссылка должна соответствовать одному из зарегистрированных значений;
- access_type – необходимо ли приложению получать refresh_token, необходимый для получения сведений о пользователе в дальнейшем, когда пользователь будет оффлайн (принимает значение “online“/“offline“, refresh_token предоставляется при access_type=offline) – опциональный параметр, если значение не задано, то поведение определяется настройкой, заданной для указанного приложения в СУДИР;
- state – набор случайных символов, имеющий вид 128-битного идентификатора запроса (используется для защиты от перехвата), это же значение будет возвращено в ответе – опциональный параметр;
- client_id – идентификатор клиента;
- prompt (необязательный параметр) – может принимать значение prompt=none. Если при выполнении запроса у СУДИР возникнет потребность отобразить пользователю экран запроса идентификации/аутентификации, то при prompt=none СУДИР не будет этого делать, а вернет системе на ее redirect_uri ошибку login_required. Вызов с параметром prompt=none нужно делать в случае, если приложение хочет проверить наличие у пользователя сессии СУДИР, но не хочет, чтобы при выполнении такой проверки пользователю отобразился экран входа СУДИР.

Пример запроса на получение авторизационного кода (запрошена аутентификация и маркер доступа с разрешениями openid и profile):

```
https://sudir.mos.ru/blitz/oauth/ae?scope=openid+profile
&response_type=code&access_type=offline
&redirect_uri=https%3A%2F%2Fais.mos.ru%2Fre
&state=342a2c0c-d9ef-4cd6-b328-b67d9baf6a7f
&client_id=ais
```

Пример ответа со значением авторизационного кода (*code*) и параметром *state*:

```
https://ais.mos.ru/re?code=f954nEzQ08DXju4wxGbSSfCX7TkZ1GvXUR7TzVus8fG
nu4AUl-YIosgax-BLXMeQQAlasD6CN2qG_0KXK5NIjARoKykhuR9IpbuzqeFxS0
```

```
&state=342a2c0c-d9ef-4cd6-b328-b67d9baf6a7f
```

Возможные ошибки при вызове /oauth/ae соответствуют RFC 6749 и описаны по ссылке https://tools.ietf.org/html/rfc6749#section-4.1.2.1.

Пример запроса на получение авторизационного кода, при котором СУДИР не должен открыть страницу входа в случае, если пользователь еще не проходил идентификацию/аутентификацию в текущем браузере:

```
https://sudir.mos.ru/blitz/oauth/ae?scope=openid+profile
&response_type=code&access_type=offline
&redirect_uri=https%3A%2F%2Fais.mos.ru%2Fre
&state=342a2c0c-d9ef-4cd6-b328-b67d9baf6a7f
&client_id=ais&prompt=none
```

Пример ответа с ошибкой, если для получения авторизационного кода пользователь должен явно пройти идентификацию/аутентификацию на странице входа СУДИР, а запрос был выполнен с параметром prompt=none

```
https://ais.mos.ru/re?error=login_required&error_description=The+Authoriza
tion+Server+requires+End-User+authentication.+This+error+MAY+be+returned+
when+the+prompt+parameter+value+in+the+Authentication+Request+is+none%2C+b
ut+the+Authentication+Request+cannot+be+completed+without+displaying+a+use
r+interface+for+End-User+authentication.&state=342a2c0c-d9ef-4cd6-b328-
b67d9baf6a7f
```

3.1.3.3. Получение маркеров

В целях проведения аутентификации и получения данных программный интерфейс СУДИР предполагает использование маркеров. Перечень маркеров и их предназначение приведены в таблице 5.

Таблица 5 – Используемые в СУДИР маркеры

№	Название	Обозначение	Предназначение
1.	Маркер доступа	access_token	Получение доступа к защищенному ресурсу, например, к данным пользователя. Маркер действителен 3600 секунд.
2.	Маркер обновления	refresh_token	Обновление маркера доступа. Маркер refresh_token предоставляется, только если для приложения при регистрации была указана необходимость получения refresh_token, или если в запросе на получение авторизационного кода был указан параметр access_type=offline. Маркер действителен до момента использования, но не дольше 365 дней.
3.	Маркер идентификации	id_token	Получение идентификационной информации, например, идентификатора пользователя. Маркер действителен 3 часа.

После получения авторизационного кода приложение должно обменять его на маркеры. Для этого приложение должно сформировать запрос методом POST на URL для

получения маркера. Запрос должен содержать заголовок Authorization со значением Basic {secret}, где secret – это "client_id:client_secret" (например, ais:topsecret) в формате base64.

Примечание: Сервис получения маркеров должен обязательно вызываться с серверов подключенного к СУДИР приложения. Нельзя вызывать сервис из выполняемого на стороне браузера программного кода (например, нельзя вызывать сервис /oauth/te из JavaScript-кода веб-страницы).

Пример заголовка:

```
Authorization: Basic YWlzOmNsaWVudF9zZWNyZXQ=
```

Тело запроса должно содержать следующие параметры:

- <code> – значение авторизационного кода, который был ранее получен;
- <grant_type> – принимает значение “authorization_code”, если авторизационный код обменивается на маркер доступа;
- <redirect_uri> – ссылка, по которой должен быть направлен пользователь после того, как даст разрешение на доступ (то же самое значение, которое было указано в запросе на получение авторизационного кода).

Пример запроса:

```
POST /blitz/oauth/te HTTP/1.1
Host: sudir.mos.ru
Authorization: Basic cG9ydGFsLmlhc2l1LmxvY2FsOnBvcnRhbC5pYXNpdS5sb2NhbA==
Content-Type: application/x-www-form-urlencoded
Cache-Control: no-cache

grant_type=authorization_code&code=FLZHSMMqXTxU8EFW3bse7qOIYqarfbdbxGaDbVf
lffFENxBltpREKs7dEkN33dFvNyUggDb2XpP4nyTlUIZUMf4xBDreSmKrOkrVV7qK8GU
&redirect_uri=https%3A%2F%2Fais.mos.ru%2Fre
```

В ответ возвращается маркер доступа, маркер обновления и маркер идентификации:

```
{
    "id_token":
"eyJhbGciOiJSUzI1NiJ9.eyJub25jZSI6Ijk4NzY1NDMyMSIsImV4cCI6MTQ0NTAwNDc3Nywi
aWF0IjoxNDQ0OTk0MjEyLCJzdWIiOiJzemF5dHNldkBkZXYtYmxpdHotaWRwLmxvYyIsImF1ZC
I6WyJsb2NhbGhvc3QvZGVtbzIiXSwiaXNzIjoiaHR0cHM6Ly9kZXYtYmxpdHoucmVheG9mdC5s
b2MifQ.Ckt_dr9J4k514MIuQEDY88A026NWzdCcIIIDxOXaZxue52eQlNVxaDXIQ8F0IyG2T-
2KmeSeVG7UK4RoYWhYWT7Skn5NfF-gFJIfXB4NfUGwt5iic5UGQkp-xGqKzTxCKduuWrp-
oVb69Bd8fFCeFYTVhB-
iWR_V1yZhYTipQt6rLVKaqEFPvRv6iN_cGGIr7k0EJtEvF6Y71ktf6ERnhCXsLn78mPeJv0H0j
k6dWWl9JJxpZC6RvUEqv4Q8q92xXh7RjoMQUErNk03J74QLIdn3xYke0ch20fOauMPLYXOc0-
cPwo5kjF5zK6-c1chrwfk2FFMCi1bGYpgcICEssQ",
    "access_token": "dO-xymwduYR8uFqvYYK1ghpk-tqantG5PstfomddlO5d2-
BVzVVaNdHuJYdWxpL__c8MsLWB8IwmiUIEep53tnZR2mflAnYiguE0UyUZNBE",
    "expires_in": 3600,
    "scope": "openid profile",
```

```
    "refresh_token":
"1lEWX7IE7SESIXRJNJeN66oPdzoFSfOGDLB3foAgXUDfuzYf1aXS1FuJX5JM50mmDqHuX75t-
DrtvM0ISa82vs_t0NI9O1AcAilduRzZ8Iw",
    "token_type": "bearer"
}
```


Если авторизационный код был уже использован или истек срок его действия, то в качестве ответа будет возвращена ошибка. Пример ответа с ошибкой:


```
{
    "error": "invalid_grant",
    "error_description": "The provided authorization grant (e.g.,
authorization code, resource owner credentials) or refresh token is
invalid, expired, revoked, does not match the redirection URI used in the
authorization request, or was issued to another client."
}
```


Возможные ошибки при вызове /oauth/te соответствуют RFC 6749 и описаны по ссылке https://tools.ietf.org/html/rfc6749#section-5.2.

Для обновления маркера доступа приложение должно сформировать запрос методом POST на URL для получения и обновления маркера. Запрос должен содержать аналогичный заголовок Authorization, а также следующие параметры в теле запроса:

– <refresh_token> – значение маркера обновления;
– <grant_type> – принимает значение “refresh_token”, если маркер обновления обменивается на маркер доступа.

Пример запроса:


```
POST /blitz/oauth/te HTTP/1.1
Host: sudir.mos.ru
Authorization: Basic cG9ydGFsLmlhc2l1LmxvY2FsOnBvcnRhbC5pYXNpdS5sb2NhbA==
Content-Type: application/x-www-form-urlencoded
Cache-Control: no-cache

grant_type=refresh_token&refresh_token=jj2DAYsPi-
aXMeSrQ7WiucRQKtz4_swxXdJDqVpC9laxIgQ_LhPmxiA8tPI7TocUkkgktXFqp28P0nC6af6S
THP46TvvO1gBTx1-aW9rPbQ
```


3.1.3.4. Получение данных и идентификационной информации

Данные пользователя могут быть получены двумя способами:

– Путем вызова REST-сервиса (рекомендуемый способ). Данные получаются в виде JSON в результате вызова специального REST-сервиса в СУДИР. См. главу 3.1.3.4.1.
– Из состава маркера идентификации (id_token). См. главу 3.1.3.4.2.

3.1.3.4.1. Получение данных пользователя через REST-сервис

Для запроса данных о пользователе необходимо выполнить запрос методом GET по URL-адресу получения данных пользователя. В запрос должен быть добавлен следующий

заголовок:

```
Authorization: Bearer <access token>
```

В заголовке <access token> – это маркер доступа, полученный от СУДИР на предыдущем шаге.

Пример запроса:

```
GET /blitz/oauth/me HTTP/1.1
Host: sudir.mos.ru
Authorization: Bearer
NINxnizbgYYQg94vEd6MjkTPxR3r2SZ3GO0HY0yEKLRXIDKsQ_0fZ-s9IAHBO92AszgTIqItY
Cache-Control: no-cache
```

В ответе будут отображены только те данные, которые определены в scope, на который получен маркер доступа. Пример ответа:

```
{
    "firstname": "Иван",
    "surname": "Иванов",
    "middlename": "Иванович",
    "email": "ivanov@it.mos.ru",
    "logonname": "ivanov.ivan"
}
```

3.1.3.4.2. Маркер идентификации. Получение данных пользователя и проверка маркера

Для получения данных об идентификации и аутентификации приложению необходимо самостоятельно анализировать содержание маркера идентификации (id_token). Маркер идентификации состоит из трех частей:

- заголовок (header), в котором содержится общая информация о типе маркера, в том числе об использованных в ходе его формирования криптографических операциях;
- набор утверждений (payload / claim set) с содержательными сведениями о маркере;
- подпись (signature), которая удостоверяет, что маркер «выдан» СУДИР и не был изменен при передаче.

Части маркера разделены точкой, так что он имеет вид:

```
HEADER.PAYLOAD.SIGNATURE
```

Маркер передается в виде строки в формате Base64url.

Заголовок (header) содержит описание алгоритма шифрования (параметр “alg”); в настоящее время в СУДИР поддерживается алгоритм электронной подписи RSA SHA-256, рекомендуемый спецификацией (соответствует значению “RS256”).

Набор утверждений включает следующие атрибуты:

- время прекращения действия (“exp”), указывается в секундах с 1 января 1970 г. 00:00:00 GMT;
- время выдачи (“iat”), указывается в секундах с 1 января 1970 г. 00:00:00 GMT;
- идентификатор субъекта (“sub”), в качестве значения указывается уникальный идентификатор пользователя (атрибут uid пользователя, совпадает с userPrincipalName в ЕСК);
- адресат маркера (“aud”), указывается client_id приложения, направившего запрос на аутентификацию;
- организация, выпустившая маркер (“iss”), указывается URL СУДИР;
- дополнительные атрибуты в соответствии с заявкой на подключение приложения к СУДИР (возможные атрибуты для включения в id_token приведены в таблице 3).

Пример набора утверждений:

```
{
  "exp": 1445004777,
  "iat": 1444994212,
  "sub": "ivanov.ivan",
  "aud": [
    "ais"
  ],
  "iss": "https://sudir.mos.ru"
}
```

Подпись (signature) маркера осуществляется по алгоритму, который указывается в параметре “alg” маркера. Подпись вычисляется от двух предыдущих частей маркера (HEADER.PAYLOAD). Сертификат открытого ключа СУДИР, необходимый для проверки подписи, можно загрузить по следующим ссылкам (находится в атрибуте x5c):

- https://sudir-test.mos.ru/blitz/oauth/.well-known/jwks (тестовая среда)
- https://sudir.mos.ru/blitz/oauth/.well-known/jwks (продуктивная среда)

После получения маркера идентификации приложению рекомендуется произвести валидацию маркера идентификации, которая включает в себя следующие проверки:

1. Получение идентификатора СУДИР, содержащегося в маркере идентификации, и получение иных необходимых приложению дополнительных атрибутов пользователя.
2. Проверка идентификатора приложения, т.е. именно приложение (client_id) должно быть указано в качестве адресата маркера идентификации.
3. Проверка подписи маркера идентификации (с использованием указанного в маркере алгоритма).
4. Проверка, что текущее время должно быть не позднее, чем время прекращения срока действия маркера идентификации.

После валидации маркера идентификации приложение может считать пользователя аутентифицированным.

Для анализа содержания маркера идентификации, а также для упрощения разработки модулей по его проверке можно воспользоваться доступными онлайн-декодерами и библиотеками[11].

3.1.3.5. Проверка маркера доступа

Данные о маркере доступа (access_token) необходимо проверять в следующих случаях:

- приложению требуется отслеживать срок действия маркера, чтобы оперативно менять его на новый;
- приложение является поставщиком ресурсов и предоставляет доступ к этим ресурсам по предъявлению маркера доступа, выданного СУДИР приложению, запрашивающему ресурс.

Для запроса данных о маркере доступа необходимо выполнить запрос методом POST по адресу сервиса интроспекции маркера доступа[12]. В запрос для аутентификации системы-клиента должен быть добавлен описанный ранее заголовок Authorization: Basic. Также должен быть добавлен заголовок “Content-Type”, принимающий значение “application/x-www-form-urlencoded”.

Сервис интроспекции может быть вызван любой системой, зарегистрированной в СУДИР, для проверки любого маркера доступа (система может проверить маркер, выданный другой системе).

В теле запроса могут быть указаны параметры:

- token – маркер доступа, данные о котором требуется просмотреть (обязательный параметр);
- token_type_hint – тип маркера доступа (например, access_token), предназначен для ускорения поиска (опциональный параметр).

Пример запроса:

```
POST /blitz/oauth/introspect HTTP/1.1
Host: sudir.mos.ru
Authorization: Basic cG9ydGFsLmlhc2l1LmxvY2FsOnBvcnRhbC5pYXNpdS5sb2NhbA==
Content-Type: application/x-www-form-urlencoded
Cache-Control: no-cache

token=MkvRff63ASFvC0-w6D3fiDxQgaYqmkrnzeDDcz374NDIbux6DA8ByDdB9-
oMfaIklyoaNfrk87a4Ep3XzSxbF9m0wA2Z7LIUOaCGuSaKDNo
```

[11] См. http://jwt.io/ и http://kjur.github.io/jsjws/mobile/tool_jwt.html#verifier

[12] Создан в соответствие с RFC 7662: https://tools.ietf.org/html/rfc7662

В ответе будут переданы следующие данные о маркере доступа:

- active – признак действительности маркера доступа, принимает значения true/false. Маркер действителен, если он выдан сервисом авторизации СУДИР, не был отозван и срок его действия не истек;
- scope – область доступа, на которую выдан маркер доступа. Передается в виде перечня разрешений (scope);
- client_id – идентификатор системы-клиента, которая получила данный маркер доступа;
- username – идентификатор пользователя (владельца ресурса, предоставившего доступ к своим данным), определенный как базовый идентификатор в СУДИР. Значение параметра возвращается только в том случае, если он может быть передан в рамках scope по предъявленному маркеру доступа;
- jti – идентификатор маркера доступа (в виде строки);
- token_type – тип предъявленного маркера доступа.

Пример ответа:

```
{
    "username": "ivanov.ivan",
    "scope": "openid profile",
    "jti": "10jdlNohfHzuv3xoFurvWSPheEJEC7KHdHr-dcaVyYYvV3h0l2sh",
    "token_type": "Bearer",
    "client_id": "ais",
    "active": true
}
```

3.1.3.6. Обеспечение логаута при использовании OpenID Connect

Если приложение предоставляет пользователю возможность инициировать выход из приложения (логаут), то приложению для обеспечения выхода недостаточно завершить локальную сессию. Необходимо также вызвать в СУДИР операцию логаута. Если этого не сделать, то может возникнуть ситуация, что пользователь в приложении нажал кнопку *Выход*, после чего сразу попробовал нажать кнопку *Вход*, и вместо ожидаемого запроса идентификации и аутентификации сработал Single Sign-On, и пользователь сразу автоматически оказался авторизованным.

Для инициирования логаута в СУДИР приложение после закрытия своей локальной сессии должно осуществить HTTP Redirect на следующий URL в СУДИР, передав в качестве query-параметра адрес возврата в приложение:

```
https://sudir.mos.ru/blitz/login/logout?post_logout_redirect_uri=https://redirect_uri
```

Если СУДИР успешно завершит логаут, то он перенаправит пользователя по

переданному URL обратно в приложение.

Допустимые префиксы страниц возврата должны быть зарегистрированы в настройках СУДИР, иначе при логауте будет выдана ошибка.

3.2. Подключение приложения по протоколу SAML

3.2.1. Данные для подключения по результатам рассмотрения заявки

По результатам исполнения заявки на подключение к СУДИР будет предоставлена следующая информация:

- идентификатор, присвоенный приложению в СУДИР (entity_id);
- уточненный файл метаданных поставщика услуг[13].

Приложение взаимодействует с сервисами СУДИР, используя следующие адреса:

- метаданные СУДИР:
  - https://sudir-test.mos.ru/blitz/saml/profile/Metadata/SAML (тестовая среда)
  - https://sudir.mos.ru/blitz/saml/profile/Metadata/SAML (продуктивная среда)
- URL для аутентификации:
  - https://sudir-test.mos.ru/blitz/saml/profile/SAML2/Redirect/SSO (тестовая среда)
  - https://sudir.mos.ru/blitz/saml/profile/SAML2/Redirect/SSO (продуктивная среда)
- URL для логаута:
  - https://sudir-test.mos.ru/blitz/saml/profile/SAML2/Redirect/SLO (тестовая среда)
  - https://sudir.mos.ru/blitz/saml/profile/SAML2/Redirect/SLO (продуктивная среда)
- URL издателя:
  - https://sudir-test.mos.ru/blitz/saml/ (тестовая среда)
  - https://sudir.mos.ru/blitz/saml/ (продуктивная среда)

Если приложение поддерживает протокол подключения SAML, то указанных данных должно быть достаточно для конфигурирования приложения. Если приложение не поддерживает протокол SAML, следует произвести его доработку согласно рекомендациям, изложенным в п. 3.2.2 и 3.2.3 настоящего документа.

Типичные вопросы о том, как настроить приложение для подключения к СУДИР по протоколу SAML, приведены в таблице ниже.

[13] Теоретически этот файл должен совпадать с тем, который был приложен к заявке. Однако администратор СУДИР может скорректировать файл метаданных и прислать его вместе с результатами рассмотрения заявки.

Таблица 6 – Типичные вопросы при подключении приложения по протоколу SAML

Вопрос	**Ответ**
Где найти метаданные поставщика идентификации СУДИР?	Чтобы загрузить метаданные, просто пройдите по ссылке вида https://sudir.mos.ru/blitz/saml/profile/Metadata/SAML и скопируйте открытый XML документ в приложение
Где найти сертификат SAML поставщика идентификации СУДИР?	Откройте XML документ с метаданными поставщика идентификации СУДИР. Найдите раздел <ds:X509Certificate></ds:X509Certificate> – в нем и располагается сертификат SAML поставщика идентификации. Пример:  Иногда для корректной загрузки в приложение перед строкой с сертификатом нужно вставить строку -----BEGIN CERTIFICATE----- , а после – -----END CERTIFICATE-----
Где найти адреса SAML-обработчиков СУДИР?	Запросы на идентификацию/аутентификацию приложение должно отправлять на следующие обработчики (SingleSignOnService) в ПРОД-среде: – https://sudir.mos.ru/blitz/saml/profile/SAML2/Redirect/SSO – для приема сжатых с помощью алгоритма Deflate запросов – стандартный SAML-обработчик. – https://sudir.mos.ru/blitz/saml/profile/SAML2/Redirect/Plain/SSO – для приема несжатых запросов – следует использовать только в случае, если подключаемое приложение не использует Deflate. Запросы на единый логаут приложение должно отправлять на следующие обработчики (SingleLogoutService) в ПРОД-среде: – https://sudir.mos.ru/blitz/saml/profile/SAML2/Redirect/SLO – для приема сжатых с помощью алгоритма Deflate запросов – стандартный SAML-обработчик. – https://sudir.mos.ru/blitz/saml/profile/SAML2/Redirect/Plain/SLO – для приема несжатых запросов – следует использовать только в случае, если подключаемое приложение не использует Deflate. В ТЕСТ-среде аналогичные адреса начинаются с https://sudir-test.mos.ru.
Какой EntityID у СУДИР?	СУДИР как поставщик идентификации имеет следующие EntityID: – Для ПРОД-среды – https://sudir.mos.ru/blitz/saml – Для ТЕСТ-среды – https://sudir-test.mos.ru/blitz/saml

3.2.2. Готовые библиотеки

Так как самостоятельная разработка программного интерфейса клиента SAML является трудоемкой задачей, а ошибки в реализации могут быть чреваты угрозами безопасности, то рекомендуется при интеграции приложения по SAML использовать существующие популярные библиотеки SAML-клиентов: OIOSAML[14] (Java, .NET), OpenSAML[15] (Java), Spring Security SAML[16] (Java), SimpleSAMLphp[17] (PHP), ruby-saml[18] (Ruby on Rails). Далее приводятся ключевые сведения, необходимые для понимания процесса аутентификации по протоколу SAML.

3.2.3. Разработка взаимодействия приложения с СУДИР по SAML

3.2.3.1. Общие сведения

Для подключения к СУДИР в целях идентификации и аутентификации пользователей приложение может использовать стандарт SAML версий 1.0, 1.1, 2.0[19]. При этом процесс взаимодействия приложения и СУДИР должен быть построен в соответствии с профилем SAML Web Browser SSO Profile[20].

Стандарт SAML основан на XML и определяет способы обмена информацией об аутентификации пользователей и их идентификационных данных (атрибуты, полномочия).

Для возможности осуществлять взаимодействия поставщик услуг и поставщик идентификации предварительно должны обменяться настройками взаимодействия, описанными в форме XML-документов и называемых метаданными. Поставщик услуг должен получить настройки СУДИР, называемые метаданными поставщика идентификации (см. п. 2.2.1).

3.2.3.2. Процесс проведения идентификации и аутентификации

В процессе взаимодействия приложение (поставщик услуг) посылает в СУДИР SAML-запрос на идентификацию пользователя (SAML Request). Запрос представляет собой оформленный в соответствии со стандартом SAML XML-документ. В запросе присутствует идентификатор запрашивающего идентификацию приложения, называемый EntityID, а также дополнительная служебная информация. Сам запрос передается подписанным электронной подписью приложения. В качестве транспортного протокола для передачи сообщения используется протокол HTTPS, вызов поставщика идентификации осуществляется через

[14] http://digitaliser.dk/group/42063/resources

[15] https://wiki.shibboleth.net/confluence/display/OS30/Home

[16] http://projects.spring.io/spring-security-saml/

[17] https://simplesamlphp.org/

[18] https://rubygems.org/gems/ruby-saml/

[19] http://saml.xml.org/saml-specifications

[20] http://docs.oasis-open.org/security/saml/v2.0/saml-profiles-2.0-os.pdf

HTTP Redirect. Это означает, что запрос от приложения к СУДИР осуществляется опосредованно, через браузер пользователя, и прямое сетевое взаимодействие между приложением и СУДИР при использовании SAML не требуется.

Получив SAML-запрос на идентификацию, СУДИР идентифицирует принадлежность запроса определенному приложению, после чего отображает пользователю веб-страницу единого входа для проведения идентификации и аутентификации пользователя. В случае успешной идентификации и аутентификации пользователя СУДИР передает приложению (поставщику услуг) SAML-ответ (SAML Response). В зависимости от заданных настроек взаимодействия запрос может быть подписанным и зашифрованным. Для формирования подписи и для шифрования используются стандарты XML Signature и XML Encryption. В качестве транспортного протокола для передачи сообщения с результатами идентификации используется протокол HTTPS, вызов поставщика услуг осуществляется через HTTP POST.

Получив от СУДИР SAML-ответ, приложение проверяет его подпись, выполняет расшифровку, после чего извлекает из SAML-утверждений (SAML Assertions) идентификационные данные пользователя (идентификаторы, атрибуты, полномочия).

Процесс взаимодействия приложения и СУДИР с использованием SAML приведен на рисунке 4.

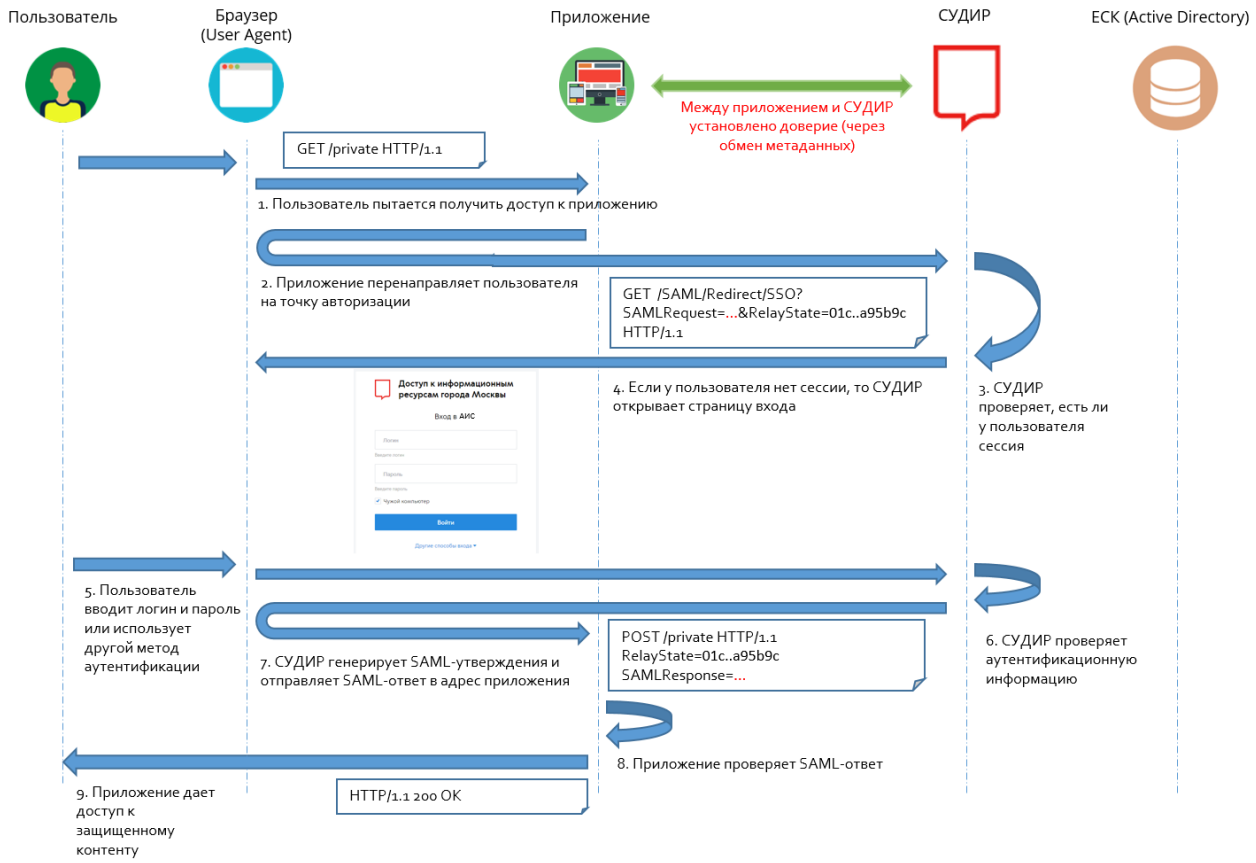

Рисунок 4 – Схема взаимодействия приложения с СУДИР по SAML

3.2.3.3. Обеспечение логаута при использовании SAML

Подключенное по SAML к СУДИР приложение также может предусматривать возможность реализации единого выхода (логаута). Для этих целей СУДИР поддерживает SAML Single Logout Profile[21]. Приложение может направить в СУДИР SAML-запрос <LogoutRequest> и в случае успешного завершения единого логаута получить от СУДИР SAML-ответ <LogoutResponse>. Если приложение должно быть задействовано в едином логауте, инициированным другим приложением, подключенным к СУДИР, то оно также должно предусматривать возможность обработки запросов <LogoutRequest>, поступивших к приложению от СУДИР. В случае успешного завершения локальной сессии приложение должно уведомлять СУДИР путем отправки ему SAML-ответа <LogoutResponse>.

Если приложению достаточно при инициировании пользователем логаута только завершить собственную сессию приложения и глобальную сессию в СУДИР, то вместо SAML Single Logout Profile можно для простого логаута инициировать HTTP Redirect на следующий URL в СУДИР, передав в качестве query-параметра адрес возврата в приложение: https://sudir.mos.ru/blitz/login/logout?post_logout_redirect_uri=https://redirect_uri. Если СУДИР успешно завершит логаут, то он перенаправит пользователя по переданному URL.

[21] https://docs.oasis-open.org/security/saml/v2.0/saml-profiles-2.0-os.pdf

Приложение 1. IP-адреса серверов СУДИР

Сервера sudir-test.mos.ru и sudir.mos.ru должны быть доступны для обращений по https (443 порт) с устройств пользователей и с тех серверов подключаемой системы, которые обращаются к сервисам СУДИР[22]. Ниже в таблице приведены IP-адреса хостов СУДИР:

Таблица 7 – IP-адреса СУДИР

Нахождение источника запроса к СУДИР	**Тестовая среда СУДИР (sudir-test.mos.ru)**	**Промышленная среда СУДИР (sudir.mos.ru)**
IP-адреса, доступные из Интернет	212.11.152.249	212.11.152.248
IP-адреса, доступные из внутренних сетей правительства и ЦОД	10.15.21.8	10.15.21.7

[22] Сервера подключаемой системы взаимодействуют с СУДИР по сети в случае интеграции с помощью OIDC. В случае же использоваться для интеграции с СУДИР протокола SAML все взаимодействия системы с СУДИР осуществляются опосредовано через веб-браузер пользователя – прямых сетевых запросов с серверов к СУДИР при подключении по SAML нет.

Перечень терминов, сокращений и обозначений

Используемые в настоящем документе сокращения, определения и основные понятия области автоматизированных систем определены в ГОСТ 34.003-90 «Информационная технология. Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Термины и определения». Также в тексте настоящего документа введены специальные термины на русском и английском языках:

Термин, сокращение, обозначение	Полная форма
HTTP	Протокол прикладного уровня передачи данных (HyperText Transfer Protocol)
HTTPS	Расширение протокола HTTP, поддерживающее шифрование
IP	Межсетевой протокол (Internet Protocol)
JSON	Текстовый формат обмена данных, основанный на JavaScript (JavaScript Object Notation)
OAuth	Открытый протокол авторизации
OIDC	Профиль, определяющий способ использования OAuth в процессе идентификации/аутентификации пользователя (OpenID Connect)
REST	Архитектурный стиль взаимодействия компонентов распределенного приложения (Representational State Transfer)
RFC	Техническая спецификация (Request for Comments)
SAML	Стандарт обмена данными для осуществления аутентификации (Security Assertion Markup Language)
SSO	Единый вход (Single Sign-On)
URL	Унифицированный указатель ресурса (Uniform Resource Locator)
XML	Расширяемый язык разметки (eXtensible Markup Language)
ГОСТ	Государственный стандарт
ЕСК	Единая служба каталогов
ОИВ	Органы исполнительной власти
ПК	Персональный компьютер
ПО	Программное обеспечение
СУДИР	Автоматизированная информационная система «Система управления доступом к информационным системам и ресурсам города Москвы»
ЦОД	Центр обработки данных
ЭП	Электронная подпись