RESTFUL WEB API к сервисам Московской биржи

Данный документ описывает принципы подключения и аутентификации к тестовому окружению биржи для тестирования следующих сервисов:

- Онлайн регистрации клиентов
- Клирингового терминала

Принципы подключения и аутентификации для вышеуказанных сервисов МБ одинаковы, в целях упрощения процесса тестирования пользователь может использовать одну тестовую запись в moex passport и один тестовый сертификат для работы с несколькими сервисами.

Получение токена доступа:

Для работы с API первоначально необходимо получить токен доступа, для этого необходимо выполнить несколько шагов:

1. Создать тестовую учетную запись на https://passport-test.moex.com/registration. Получить MOEX Passport Token, выполнив GET запрос по адресу https://passport-test.moex.com/authenticate, используя Basic аутентификацию с учетными данными пользователя, от имени которого предполагается работа с API. Значение **Moex Passport Token** будет возвращено в куке MicexPassportCert;
2. Прислать на адрес help@moex.com письмо с заголовком «Тестирование online регистрации» или «Тестирование API к КТ», в котором указать:
   - Email, который использовался для регистрации в MOEX Passport в п. 1;
   - Идентификатор фирмы, с которой вы работаете на тестовых стендах фондового валютного или срочного рынков (если есть). Если такой фирмы нет, то она будет создана.
   - Желаемый тип сертификата для п. 3 (RSA или ГОСТ)

   В ответном письме будут высланы:
   - Тестовый сертификат пользователя для п. 3;
   - **client_id** и **client_secret** для п. 4;
   - Данные для подключения к тестовым стендам, если необходимы.

3. **Для ГОСТ-сертификата**: скачать дистрибутивы можно тут https://www.moex.com/s1292
   - Скачать и установить Дистрибутив АПК Клиент МБ (ПК "Справочник сертификатов", XCS) соответствующей разрядности (если не установлено/без поддержки компонент TLS, при необходимости добавить считыватель типа «Реестр»). Перезагрузить компьютер.
   - Скачать и установить Дистрибутив СКЗИ "Валидата CSP" соответствующей разрядности (если не установлено). Запустить.
   - Запустить Справочник сертификатов. При первичном запуске необходимо выполнить восстановление из резервной копии. В полученном в п. 2 архиве с тестовым сертификатом в папке Spr находится база локального справочника сертификатов. Эту папку нужно указать в качестве источника резервной копии. Также можно скопировать ее содержимое в каталог профиля C:\Users\<USER>\AppData\Roaming\Validata\xcs\. При следующем запуске указать тестовый сертификат. В дальнейшем при запуске Справочника будет выводиться сообщение с просьбой предоставить носитель с ключом, для этого необходимо выгрузить содержимое папки vdkeys из архива в корневой каталог съемного носителя (виртуальная дискета или USB-флэш накопитель).

- Скачать утилиту Утилита командной строки для использования в СЭД (прямая ссылка http://fs.moex.com/cdp/po/xCertUtil.zip). Для ГОСТ необходимо использовать утилиту xpki1utl* соответствующей разрядности. Краткое описание параметров запуска находится в архиве, файлы RunUtil.txt и xpki1utl.txt. Создать отделенную ЭЦП, пример команды:

  xpki1utl.exe -profile <User> -sign -detached -data <token> -out token.p7d

Для RSA-сертификата: скачать дистрибутивы можно тут https://www.moex.com/s1293

- Скачать и установить Дистрибутив ПКЗИ СЭД МБ (ПК "Справочник сертификатов", RCS) версии не ниже 6.0.
- Запустить Справочник сертификатов. При первичном запуске необходимо выполнить восстановление из резервной копии. В полученном в п. 2 архиве с тестовым сертификатом в папке Spr находится база локального справочника сертификатов. Эту папку нужно указать в качестве источника резервной копии. Также можно скопировать ее содержимое в каталог профиля C:\Users\<USER>\AppData\Roaming\Validata\rcs\. При следующем запуске указать ключ UserOrg.rsa.
- Скачать утилиту Утилита командной строки для использования в СЭД (прямая ссылка http://fs.moex.com/cdp/po/rpkiutlv6.zip). Для RSA необходимо использовать утилиту rpki1utl* соответствующей разрядности. Краткое описание параметров запуска находится в архиве, файлы RunUtil.txt и rpki1utl.txt. Создать отделенную ЭЦП, пример команды:

  rpki1utl.exe -profile <User> -sign -detached -data <token> -out token.p7d

4. Далее полученный токен необходимо закодировать в Base64 с помощью любого инструмента или функции используемого языка программирования. Пример использования стандартной команды в Windows:

   certutil -encode token.p7d token.sig

   Полученный файл подписи не должен содержать переносов строк! Соответственно, для команд, не имеющих параметра для создания файла без переносов (включая приведённый пример с certutils), необходимо их удалить из закодированного в Base64 файла.

5. Выполнить POST запрос по адресу https://play-api.moex.com/auth/oauth/v2/token, использовав следующие параметры (параметры должны передаваться с использованием метода "application/x-www-form-urlencoded"):
   - **grant_type** – passport
   - **scope** – запрашиваемые права доступа (в случае онлайн-регистрации значение равно *client_registration*)
   - **client_id** – идентификатор приложения, полученный в п. 2
   - **client_secret** – ключ безопасности, полученный в п. 2
   - **certificate** - MOEX Passport Token, полученный в п. 1
   - **algorithm** – значение GOST или RSA, в зависимости от типа подписи, использованной при формировании электронной подписи MOEX Passport Token
   - **signature** – электронная подпись MOEX Passport Token, сформированная в п. 4, в Base64 кодировке

Общий алгоритм 2-х факторной аутентификации:

MicexPassportCert = Аутентифицироваться_MOEX_Passport (login, password) ;

Signature = Валидата.создать_ЭЦП (**MicexPassportCert** , CLIENT_CERTIFICATE);

access_token = HTTP.post (url=https://play-api.moex.com/auth/oauth/v2/token, параметры= {

```
                grant_type= “passport”, scope “client_registration”,
        client_id=Client_id, client_secret=Client_secret,
                certificate= MicexPassportCert,
            algorithm=”GOST”|”RSA”, signature=Signature
        }
    );
```

Если запрос выполнится успешно, вы получите JSON объект со следующими полями:

- **access_token** – токен доступа, который должен передаваться при каждом вызове API
- **expires_int** – время жизни токена доступа в секундах
- **refresh_tokent** – токен обновления, токен который необходимо использовать при обновлении текущего токена доступа
- **token_type** – всегда имеет значение *bearer*

В случае же, если переданные данные не являются валидными (например, приложение с таким client_id отсутствует, client_secret не соответствует client_id или же переданная электронная подпись не соответствует переданному MOEX Passport токену) результатом буден HTTP Response Code 403

Использование токена доступа

Теперь, когда у вас есть токен доступа, необходимо использовать его для подписания запросов, отправленных в API.

Для этого к вашим запросам добавляется следующий заголовок:

Authorization: Bearer <access_token>

В случае, если используемый Access Token (токен доступа) не является валидным или время его жизни истекло, в ответ вы получите HTTP Response Code 401.
При получении ответа с данным кодом ошибки, вы можете повторно запросить токен доступа так, как это описано ранее.

Системные требования для клиентской части:

- При использовании СКЗИ Валидата:
  - Операционная система Windows 7 и новее.
  - В настройках безопасности браузера необходимо отключить протокол tls ver 1.0 и выставить поддержку протокола tls ver 1.1 и/или tls ver 1.2
- Должна устанавливаться telnet-сессия на адреса https://passport-test.moex.com и https://play-api.moex.com на порт 443/tcp
- HTTPS запросы должны быть разрешены в вашей локальной сети.