

Создание приложений с SSO-аутентификацией через ADVANTA

# Оглавление

| Что такое SSO   | 3  |
|---|----|
| Порядок SSO-аутентификации в ADVANTA                      | 3  |
| Регистрация идентификаторов приложений в ADVANTA          | 4  |
| Особенности передачи токена для приложений и веб-сервисов | 4  |
| Удаленная передача токена                                 | 4  |
| Локальная передача токена                                 | 5  |
| Примеры использования SSO-авторизации в приложениях       | 6  |
| Пример приложения с локальной передачей токена            | 7  |
| Процесс работы примера                                    | 8  |
| Описание исходного кода примера                           | 11 |
| Пример приложения с удалённой передачей токена            | 12 |
| Процесс работы примера                                    | 13 |
| Описание исходного кода примера                           | 17 |

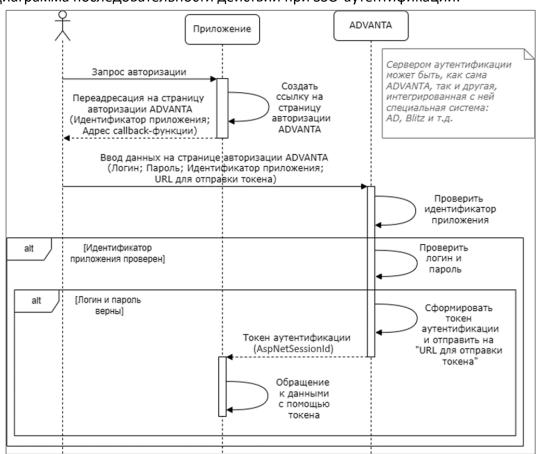
### Что такое SSO

Single Sign-On (SSO) — это метод получения доступа, который позволяет пользователям с помощью одной учётной записи входить в разные приложения, связанные с единым сервером авторизации. Этот сервер управляет проверкой учетных данных и автоматически выдает связанным с ним приложениям токены доступа.

Система ADVANTA может являться сервером SSO-аутентификации начиная с версии 3.29. Создание и хранение учётных записей может производиться, как в собственной базе данных ADVANTA, так и в специализированной внешней системе, например, <u>Active Directory</u>, <u>Blitz Identity Provider</u> и подобных.

# Порядок SSO-аутентификации в ADVANTA

Диаграмма последовательности действий при SSO-аутентификации:



Последовательность действий при SSO-аутентификации:

- 1. Пользователь инициирует процесс авторизации в клиентском приложении.
- 2. Приложение формирует ссылку на страницу авторизации ADVANTA. Ссылка содержит два параметра:
  - идентификатор приложения;
  - URL-адрес для отправки токена.
- 3. Приложение либо автоматически переадресует пользователя на страницу авторизации в браузере, либо предлагает ему перейти по созданной ссылке самостоятельно.

- 4. При входе пользователя на страницу авторизации в браузере, ADVANTA проверяет, авторизован ли он:
- если пользователь недавно уже выполнял авторизацию на этом сервере, ADVANTA автоматически применяет его учётные данные;
  - если пользователь не был авторизован просит его ввести логин и пароль.
  - 5. ADVANTA проверяет зарегистрирован ли идентификатор приложения на этом стенде.
- 6. Если все проверки прошли успешно, ADVANTA отправляет токен аутентификации AspNetSessionId приложению на указанный URL-адрес.
- 7. Приложение, получив токен, использует его для выполнения запросов к APIфункциям ADVANTA с уровнем доступа авторизовавшегося пользователя.

Перечень API-функций, в которых можно использовать токен (в качестве идентификатора AspNetSessionId), можно найти на странице Wiki «Описание методов API».

# Регистрация идентификаторов приложений в ADVANTA

Для каждого Приложения, в котором будет использован механизм SSO-авторизации, в конфигурационном файле ADVANTA client.config должен быть зарегистрирован идентификатор. Он вводится в параметре AllowedExternalApplicationClientIds. Если предполагается использовать SSO-авторизацию в нескольких приложениях, их идентификаторы указываются в этом параметре через точку с запятой.

В примере ниже будет использован идентификатор AuthDemold. В конфигурационном файле он может выглядеть, например, так:

<add key="AllowedExternalApplicationClientIds" value="application1; AuthDemoId; application3"/>

## Особенности передачи токена для приложений и веб-сервисов

В случае успешной авторизации пользователя, ADVANTA возвращает информационной системе токен доступа на адрес, указанный в параметре ReturnUrl. Инициатором передачи является именно система аутентификации.

В зависимости от типа информационной системы, принимающей токен, возможны два способа передачи:

- удаленно через вызов АРІ-функции приложения;
- локально через localhost.

### Удаленная передача токена

Если информационная система является веб-приложением, она должна предоставить для приёма токена API-функцию, вызов которой возможен через Интернет. В случае успешной аутентификации, сервер системы ADVANTA выполнит запрос этой API-функции, отправив ей сообщение методом POST. В теле сообщения он передаст в формате JSON токен доступа (в параметре tokenValue) или текст ошибки (в параметре errorMessage). Пример сообщения:

```
POST
{
    "tokenValue":"...",
    "errorMessage":"..."
}
```

## Пример ссылки переадресации пользователя при удалённой передаче токена:

https://advantaservername.ru/Pages/Login.aspx?ClientId=AuthDemoId&ReturnUrl=https://clientservername.ru/api/vl/authcallback/

#### где:

- <a href="https://advantaservername.ru">https://advantaservername.ru</a> сервер ADVANTA, выполняющий аутентификацию;
- /Pages/Login.aspx? стандартный путь к странице авторизации ADVANTA;
- AuthDemoId идентификатор клиентского приложения, зарегистрированный в файле client.config;
- <a href="https://clientservername.ru">https://clientservername.ru</a> сервер клиентской информационной системы, принимающей токен;
- /api/v1/authcallback/ путь к API-функции на сервере клиентского приложения, которая получит и обработает токен доступа.

### Локальная передача токена

▲

**Примечание:** Способ локальной передачи токена доступен начиная с версии ADVANTA 3.30.

Если авторизация выполняется в обычной программе на компьютере пользователя, токен передаётся путём Get-запроса на локальный сетевой адрес localhost. Удалённый вызов API-функции приложения обычно невозможен из-за ограниченных прав пользователя, настроек брандмауэра или сетевой архитектуры. Передачу сообщения с токеном в этом случае выполняет страница авторизации, открытая пользователем в браузере.

Для реализации данного способа приложение должно включить в переменную ReturnUrl adpec localhost или «127.0.0.1», а также номер порта. До переадресации пользователя на страницу авторизации ADVANTA программе необходимо начать «прослушивание» указанного порта.

### Пример ссылки для получения токена через localhost:

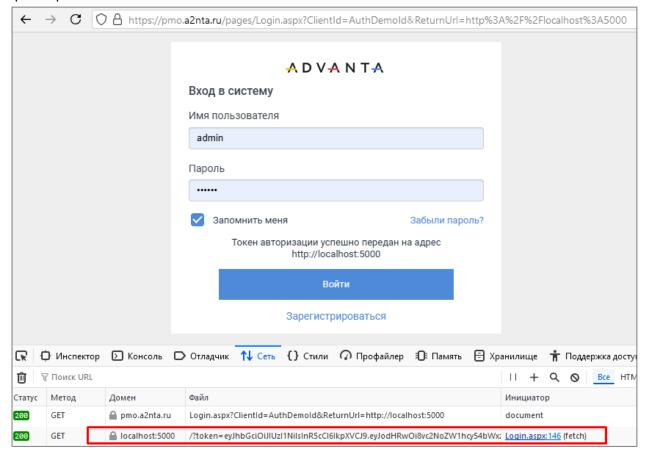
https://advantaservername.ru/Pages/Login.aspx?ClientId=AuthDemoId&ReturnUrl=http://localhost:5000/

#### где:

- <a href="https://advantaservername.ru">https://advantaservername.ru</a> сервер ADVANTA, выполняющий проверку учётной записи;
  - /Pages/Login.aspx? стандартный путь к странице авторизации ADVANTA;
- AuthDemoId идентификатор клиентского приложения, зарегистрированный в файле client.config;
- <a href="http://localhost:5000/">http://localhost:5000/</a> адрес localhost и номер порта, на который будет отправлен запрос с токеном.

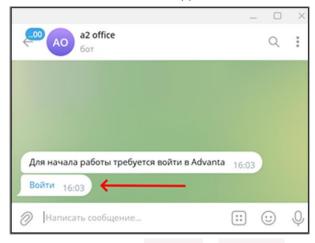
Порт лучше выбрать в диапазоне от 1024 до 49151. На прослушивание портов в этом диапазоне права администратора пользователю обычно не требуются.

В случае успешной аутентификации страница авторизации отправит токен в запросе, в параметре «token = ...»:



## Примеры использования SSO-авторизации в приложениях

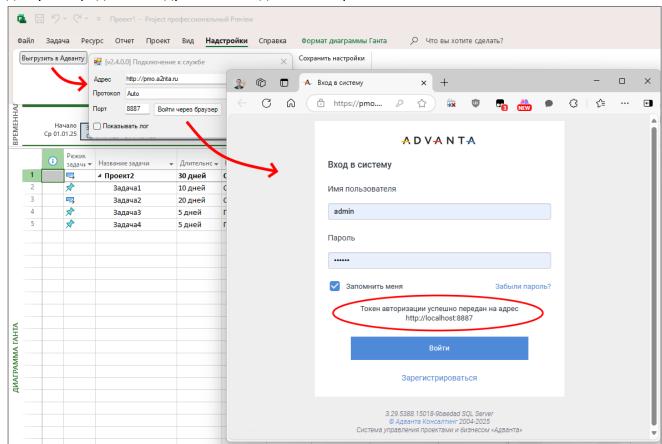
SSO-авторизация с передачей токена через API-функцию используется, например, в чатботах, создаваемых для работы с ADVANTA через мессенджеры. При первом подключении пользователя к такому чат-боту он выведет пользователю сообщение с просьбой перейти на страницу авторизации ADVANTA. Оно может выглядеть так:



Ссылка «Войти» содержит параметры ClientId и ReturnUrl, как указано выше.

API-функция, указанная в параметре ReturnUrl, должна быть реализована в серверной части чат-бота.

Локальная SSO-авторизация, через localhost, реализована в модуле импорта составов проектов из Microsoft Project. Если он установлен на компьютере пользователя, в окне MS Project, на закладке «Надстройки» появляются кнопки для обмена данными с ADVANTA. В диалоге «Подключение к службе», который появляется при нажатии кнопки «Выгрузить в Адванту», есть кнопка «Войти через браузер». По нажатию этой кнопки происходит переадресация пользователя на страницу SSO-авторизации ADVANTA. После авторизации токен доступа передаётся модулю обмена данными через localhost.



# Пример приложения с локальной передачей токена

В прилагаемом архиве «A2-SSO-Demo.zip» приведён исходный код консольного приложения на языке С#, которое демонстрирует работу SSO-аутентификации в ADVANTA. Код создан в бесплатной среде разработки Visual Studio Code. Для его запуска требуется, чтобы на ПК был установлен набор библиотек «.net SDK» версии не ниже 8.

Найти ссылку на загрузку .net SDK можно на сайте Microsoft: <a href="https://dotnet.microsoft.com/ru-ru/download/dotnet">https://dotnet.microsoft.com/ru-ru/download/dotnet</a>.

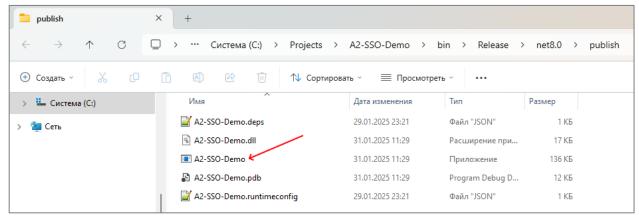
Visual Studio Code – бесплатный кроссплатформенный редактор кода, созданный Microsoft. Существуют версии для Windows, Linux и MacOS. Скачать можно с официального сайта: <a href="https://code.visualstudio.com">https://code.visualstudio.com</a>.

Архив приложения «A2-SSO-Demo.zip» нужно скачать, распаковать и поместить содержащуюся в нём папку «A2-SSO-Demo», например, в каталог «C:\Projects».

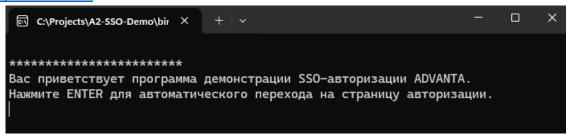
## Процесс работы примера

Прежде, чем запускать приложение, откройте в браузере страницу демонстрационного стенда ADVANTA: <a href="http://pmo.a2nta.ru">http://pmo.a2nta.ru</a>. Нужно убедиться, что он доступен из вашего сетевого окружения.

Запустите файл «A2-SSO-Demo.exe» из каталога «C:\Projects\A2-SSO-Demo\bin\Release\net8.0\publish».



Появится консоль Windows с предложением нажать ENTER для автоматического перехода на страницу SSO-авторизации ADVANTA на демонстрационном стенде <a href="http://pmo.a2nta.ru">http://pmo.a2nta.ru</a>.



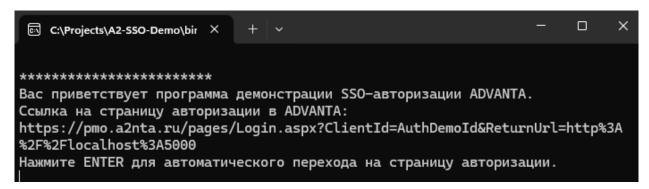
После нажатия ENTER, программа запустит браузер и откроет страницу авторизации по адресу https://pmo.a2nta.ru/pages/Login.aspx с двумя параметрами:

- ClientId = AuthDemoId;
- ReturnUrl = http%3A%2F%2Flocalhost%3A5000.

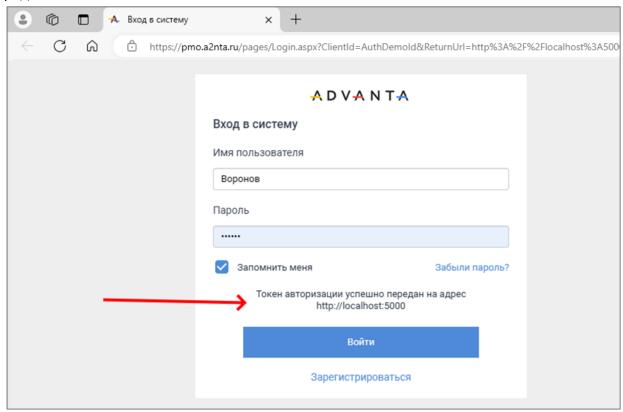
#### Где:

- ClientId идентификатор приложения, зарегистрированного в файле client.config на сервере ADVANTA, в строке AllowedExternalApplicationClientIds;
- ReturnUrl адрес, на который страница авторизации должна будет переслать токен авторизации в случае успешного ввода логина и пароля пользователем на странице ADVANTA.

В консоли приложения появятся сообщение о переходе по ссылке и о начале прослушивания порта 5000 в ожидании сообщения с токеном.

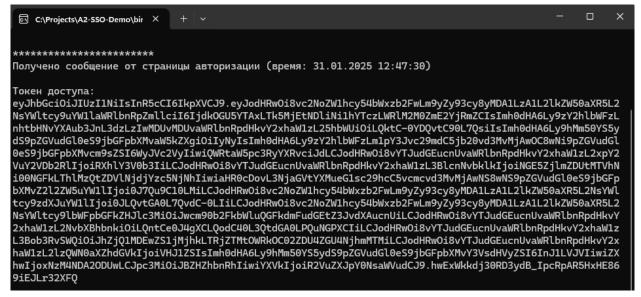


Если к моменту запуска браузера Вы уже были авторизованы на стенде <a href="https://pmo.a2nta.ru">https://pmo.a2nta.ru</a>, то страница авторизации сразу же получит токен доступа и отправит на адрес <a href="http://localhost:5000">http://localhost:5000</a>. При этом появится сообщение, что «Токен авторизации успешно передан...».



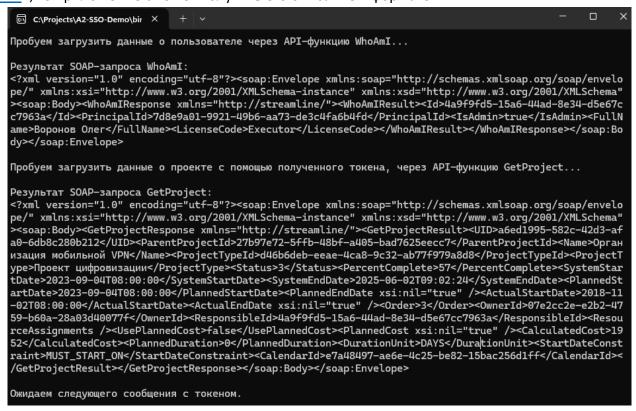
Если Вы ранее не были авторизованы на этом стенде, потребуется ввести логин и пароль, после чего токен будет отправлен и появится сообщение об успешной отправке.

Переданный приложению токен будет выведен в консоль:



В момент получения токена доступа демонстрационное приложение вызывает две API-функции ADVANTA. Функция WhoAmI предоставит информацию о пользователе, токен которого получило приложение. Функция GetProject — информацию о тестовом проекте «Организация мобильной VPN» (UID «a6ed1995-582c-42d3-afa0-6db8c280b212»).

В консоль будет выведена информация о пользователе, которому выдан токен, а также – итог запроса данных о проекте. Если учётная запись, под которой вы прошли авторизацию на стенде <a href="https://pmo.a2nta.ru">https://pmo.a2nta.ru</a>, обладает правами на просмотр проекта «Организация мобильной VPN», то приложение сможет получить его описание в формате XML.



Если прав доступа к проекту будет недостаточно, вместо информации о нём появится сообщение об ошибке.

К примеру, на странице SSO-авторизации можно сменить учётную запись на «Шилкова» (пароль «123456»). В этом случае API <u>WhoAml</u> будет выполнена успешно, поскольку у каждого пользователя есть доступ к информации о себе. При этом API-функция <u>GetProject</u> вместо

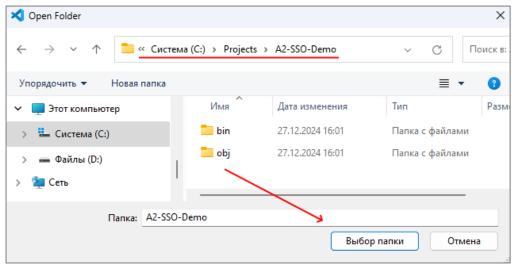
сведений о проекте получит сообщение о недостаточном уровне доступа, поскольку у пользователя «Шилкова» доступа именно к указанному проекту нет.

Пробуем загрузить данные о пользователе через API-функцию WhoAmI... Результат SOAP-запроса WhoAmI: <!xml version="1.0" encoding="utf-8"?><soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema"><soap:Body><WhoAmIResponse xmlns="http://streamline/"><WhoAmIResponse xmlns="http://streamline/"><WhoAmIResponse xmlns="http://streamline/"><WhoAmIResponse xmlns="http://streamline/"><WhoAmIResponse xmlns="http://streamline/"><WhoAmIResponse xmlns="http://streamline/"><WhoAmIResponse xmlns="http://streamline/"></whoAmIResponse xmlns="http://streamline/"></www.wax.accomplexaccompl 81329364cc8a</id><PrincipalId>74be20d0-c537-4e65-8936-d3ca529d9a08</PrincipalId><IsAdmin>false</IsAdm in><FullName>Шилкова Элина</FullName><LicenseCode>Executor</LicenseCode></WhoAmIResult></WhoAmIRespon se></soap:Body></soap:Envelope> Пробуем загрузить данные о проекте с помощью полученного токена, через API-функцию GetProject... Результат SOAP-запроса GetProject: <?xml version="1.0" encoding="utf-8"?><soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/enve</pre> lope/" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSche ols.SoapException: Серверу не удалось обработать запрос. ---> System.Exception: У Вас нет прав на просмотр проекта в SL.components.Services.APIProjects.GetProject(ProjectIdDataContract contract) в D:\jenkins\works pace\AdvantaBuild\_Commit\_OnDemand\_ByAgent1\SL\components\Services\APIProjects.asmx.cs:строка 690 Конец трассировки внутреннего стека исключений ---</faultstring><detail /></soap:Fault></soap: Body></soap:Envelope> Ожидаем следующего сообщения с токеном.

### Описание исходного кода примера

Рассмотрим основные моменты исходного кода примера.

Запустите Visual Studio Code и откройте каталог с кодом программы командой «File > Open Folder».



Основные моменты, которые могут быть полезны для разработки собственных приложений с SSO-аутентификацией:

- 1. Для получения токена приложение прослушивает порт с помощью объекта класса HttpListener. Если не открывать порт на брандмауэре, то принять токен можно будет только через localhost. Если потребуется принимать токен через внешний IP-адрес, потребуется открыть порт на доступ через брандмауэр.
- 2. При формировании ссылки на страницу авторизации требуется экранировать спецсимволы в строке параметра ReturnUrl через метод Uri. EscapeDataString().

3. В обработчике запросов, поступающих на указанный порт, исключаем все типы, кроме GET, поскольку через localhost страница авторизации ADVANTA передаёт их именно этим методом.

```
75
             // Находит токен в поступившем на указанный порт сообщении
76
             private static async Task HandleRequest(HttpListenerContext context)
77
78
                 // Проверка на тип запроса
79
                 if (context.Request.HttpMethod != "GET")
80
81
                     // Если метод не GET, возвращаем 405 Method Not Allowed
                     context.Response.StatusCode = (int)HttpStatusCode.MethodNotAllowed;
22
                     string responseTextNoGet = "Метод не разрешен. Используйте GET.";
83
84
                     byte[] bufferNoGet = Encoding.UTF8.GetBytes(responseTextNoGet);
                     context.Response.ContentType = "text/plain; charset=utf-8";
85
86
                     context.Response.ContentLength64 = bufferNoGet.Length;
                     await context.Response.OutputStream.WriteAsync(bufferNoGet, 0, bufferNoGet.Length);
87
                     context.Response.Close(); // Заканчиваем обработку запроса
88
89
                     return; // Прерываем дальнейшую обработку
90
91
```

- 4. Получение данных через API-функции ADVANTA мы выполняем в методах MakeSoapRequestWhoAmI() и MakeSoapRequestGetProject(). Они принимают в качестве входного параметра токен доступа, полученный от страницы авторизации.
- 5. В методе MakeSoapRequestWhoAmI() реализован код для получения данных через API-функции WhoAmI. В теле запроса токен доступа подставляется в параметр ASPNETSessionId.

```
130
                   // Устанавливаем формат запроса, как указано в описании API WhoAmI
131
                   string url = $"{serverName}/components/services/login.asmx";
                   string soapXml = $@"<?xml version=""1.0"" encoding=""utf-8""?>
132
                       <soap:Envelope xmlns:soap=""http://schemas.xmlsoap.org/soap/envelope/"" xmlns:xsi=""htt</pre>
133
134
                           <soap:Body>
                               <WhoAmI xmlns=""http://streamline/"">
135
                                   <ASPNETSessionId>{sessionId}</ASPNETSessionId>
136
137
                               </WhoAmI>
138
                           </soap:Body>
                       </soap:Envelope>";
139
```

6. Аналогично, в методе MakeSoapRequestGetProject() выполняется запрос к API-функции GetProject. Здесь также токен подставляется в параметр ASPNETSessionId, а в параметр ProjectId записывается идентификатор проекта.

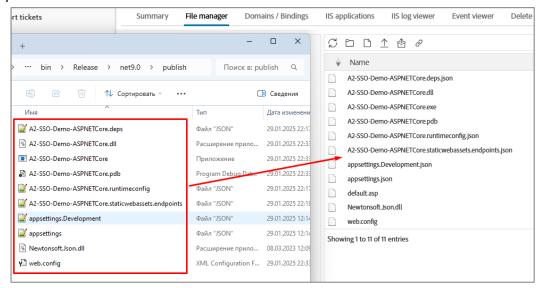
# Пример приложения с удалённой передачей токена

Приложение «A2-SSO-Demo-ASPNETCore» демонстрирует работу SSO-аутентификации для платформы ASP.NET Core. На веб-странице приложения будут отображаться ссылка на страницу авторизации ADVANTA, извлечённый токен и результаты вызова API-функций. Серверная часть приложения будет принимать POST-запросы от сервера авторизации как через внешний IP, так и через localhost.

▲ Примечание: Способ передачи токена через localhost доступен начиная с версии ADVANTA 3.30.

Для запуска приложения необходим веб-сервер с платформой ASP.NET Core 9, который доступен из Интернет. Для тестирования примера можно использовать любой ASP.NET-хостинг, например, бесплатный <a href="https://somee.com">https://somee.com</a>.

В прилагаемом архиве «A2-SSO-Demo-ASPNETCore.zip» найти каталог «\bin\Release\net9.0\publish» и загрузить файлы из него в директорию вашего тестового сайта на сервере.



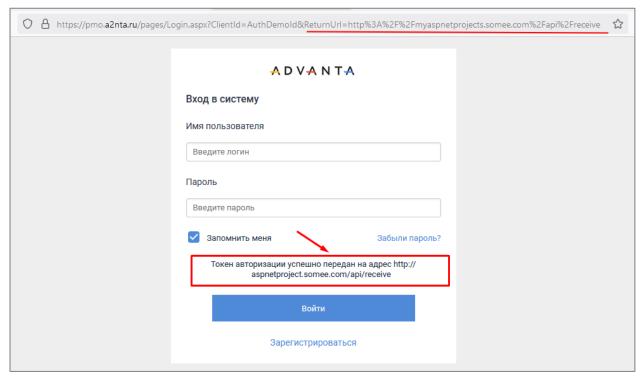
Сразу после загрузки приложение должно быть доступно по адресу вашего сайта. В браузере обязательно должен быть включен JavaScript.

## Процесс работы примера

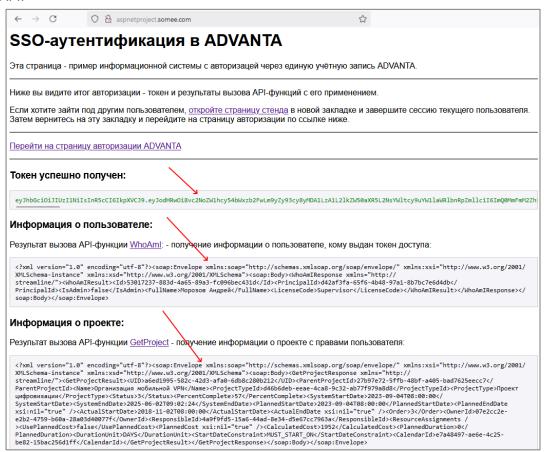
Перейдите на страницу вашего AspNet-сайта в браузере, вы увидите страницу демонстрационного приложения с предложением перейти по ссылке на страницу авторизации демонстрационного стенда ADVANTA.



При переходе по ссылке в новой вкладке браузера будет открыта страница SSO-аутентификации ADVANTA. При этом страница автоматически отправит серверу демонстрационного приложения POST-запрос с токеном доступа на адрес API-функции, который тот передал в параметре ReturnURL.



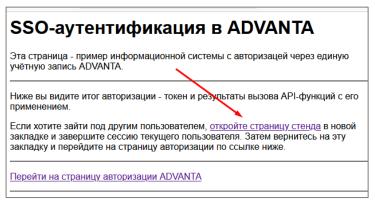
Сразу же после этого можно вернуться на страницу демонстрационного приложения. При этом страница обновится и загрузит полученный от ADVANTA токен доступа и результаты вызова API.



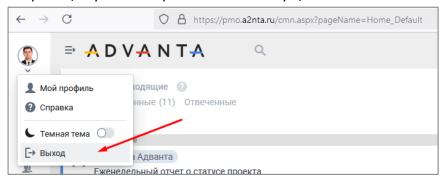
В момент получения токена доступа демонстрационное приложение вызывает две API-функции ADVANTA. Функция WhoAmI предоставит информацию о пользователе, токен которого получило демонстрационное приложение. Функция GetProject — информацию о тестовом проекте «Организация мобильной VPN» (UID «a6ed1995-582c-42d3-afa0-6db8c280b212»).

Если потребуется авторизоваться под другой учётной записью, то сделать это надо в три этапа:

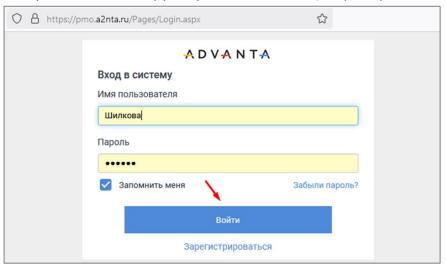
1. Сначала перейти на главную страницу демонстрационного стенда ADVANTA по ссылке на странице приложения:



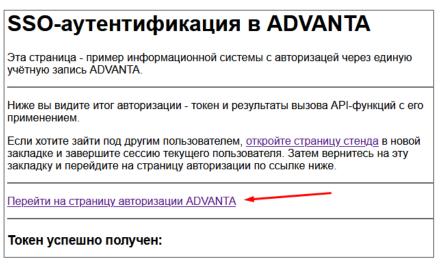
2. На этой странице нужно завершить сессию текущего пользователя:



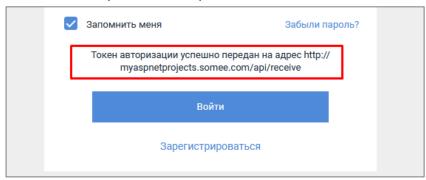
3. Затем авторизоваться под другой учётной записью (например, «Шилкова»):



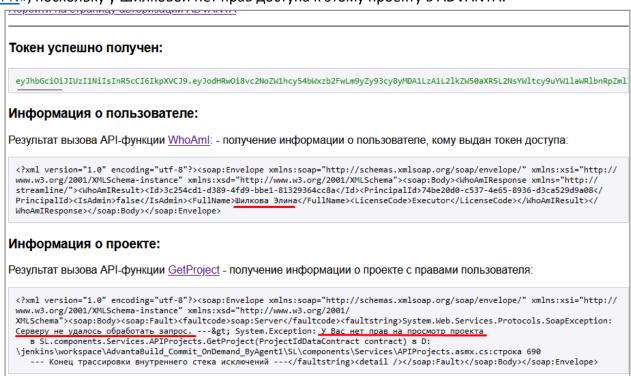
4. После этого нужно вернуться на страницу демонстрационного приложения и с неё перейти по ссылке на страницу SSO-авторизации ADVANTA:



Страница SSO-авторизации отправит демонстрационному приложению токен текущего авторизованного пользователя («Шилковой»).

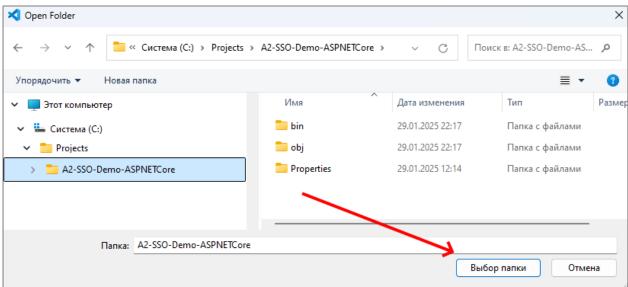


После этого нужно вернуться на страницу демонстрационного приложения. Через несколько секунд она обновится и будет отображена информация о пользователе «Шилкова». При этом приложение не сможет получить информацию о проекте «Организация мобильной VPN», поскольку у Шилковой нет прав доступа к этому проекту в ADVANTA.



### Описание исходного кода примера

Запустите Visual Studio Code и откройте каталог с кодом программы командой «File > Open Folder».



Проект создан на базе простого web-шаблона ASP.NET Core, весь код — в файле Program.cs. Основные моменты, которые могут быть полезны для разработки собственных приложений с SSO-аутентификацией:

- 1. В принципе, ASP.NET-приложение может принимать токены доступа из ADVANTA как по открытому IP-адресу, так и через localhost. В коде для примера реализованы оба варианта. Код для приёма GET-запроса с query-параметром token в функции app.MapGet, а для POST-запроса, который передаёт данные в теле, в JSON-формате, в элементе tokenValue в функции app.MapPost.
- 2. Режим передачи токена (локально или удалённо) зависит от содержания переменной ReturnUrl. Если она содержит localhost или «127.0.0.1» токен будет отправлен GET-запросом. Иначе через POST-запрос.
- 3. Значение переменной ReturnUrl устанавливается автоматически, в зависимости от информации в «контексте» приложения. Если приложение запущено на веб-сервере в интернете, в переменную будет подставлен IP-адрес. Если на клиентском ПК − то localhost.

```
30
31 // Получаем хост из текущего запроса
32 var host = context.Request.Host.Host;
33 // Выбираем значение clientId в зависимости от хоста
34 string clientId = host.Contains("localhost") ? clientId_loc : clientId_srv;
35
```

- 4. В случае передачи токена через Интернет используется идентификатор «AuthDemold». В реальных приложениях вам следует использовать свой идентификатор, который вы зарегистрируете в файле client.config вашего сервера ADVANTA.
- 5. В случае, если ADVANTA по каким-то причинам не смогла сформировать токен, она передаст сообщение об ошибке. При GET-запросе в query-параметре error, а при POST-запросе в элементе errorMessage JSON-данных, в теле сообщения.

6. Вызов API-функции <u>WhoAmI</u> реализован в методе GetUserInfo, а функции <u>GetProject</u> – в методе GetProject. Им передаётся полученный от ADVANTA токен доступа, который подставляется в SOAP-запрос в качестве параметра <u><aspnersesionId></u>.