



Дата публикации: 25.01.2022

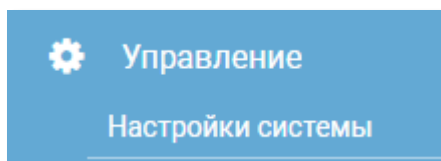
Аутентификация при разработке интеграции ПО BioTime со сторонними приложениями. Изменения от 01.2022

В предыдущих версиях программы для работы сторонних приложений с ПО BioTime через API требовалось указать только пару логин/пароль пользователя BioTime, имеющего права на чтение и запись необходимых данных. Ключ доступа был один для всех приложений.

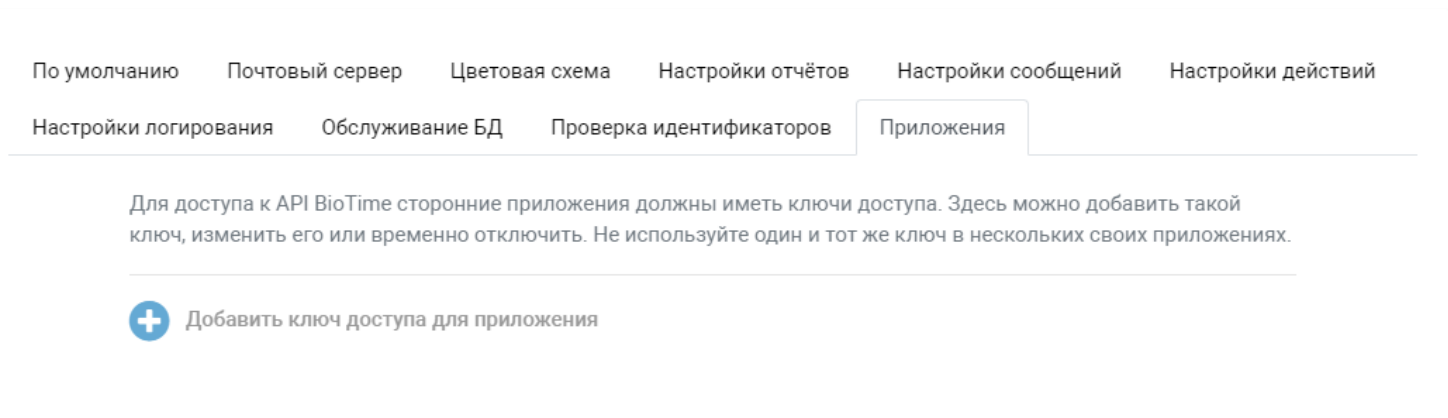
Теперь для подключения сторонних приложений к API BioTime требуется указывать не только пару логин/пароль, но индивидуальный секретный ключ доступа, сгенерированный ПО BioTime.

Получить ключ можно в разделе "Управление / Настройки системы / Приложения", создав соответствующую запись.

Добавление нового ключа



(перейдите в меню "Настройки системы")



(На вкладке "Приложения" добавьте новый ключ)

Название приложения	<input type="text" value="Название приложения"/>
Описание	<input type="text" value="Описание"/>
Идентификатор (ClientId)	<input type="text" value="Идентификатор (ClientId)"/> ?
Секретный ключ (ClientSecret)	<input type="text" value="e50bc22ca2214819a9e3a4cf5bc9b63d62a7c040238b435f94bffd0ed8fdfac"/>
Состояние	<input type="text" value="Подключение разрешено"/>

(Введите данные нового приложения и нажмите «Сохранить»)

Название приложения	<input type="text" value="Наша программа"/>
Описание	<input type="text" value="Интеграция с приложением"/>
Идентификатор (ClientId)	<input type="text" value="MyApp"/>
Секретный ключ (ClientSecret)	<input type="text" value="e50bc22ca2214819a9e3a4cf5bc9b63d62a7c040238b435f94bffd0ed8fdfac"/>
Состояние	<input type="text" value="Подключение разрешено"/>

(Пример корректно заполненных данных)

Если используется типовая внешняя обработка 1С (с закрытым кодом), ничего дополнительно создавать не требуется. Будет использован встроенный в ПО BioTime ключ.

Пример изменения кода запроса на С#

В старых версиях:

```
string baseUrl = "http://localhost:9000/";

//по протоколу OAuth 2.0 описываем что за приложение обращается к API
string APIClientId = "WebManager";
string APIClientSecret = "thisRandomStringUsedToSecureCompanyAPI";

//получаем токен доступа (он не вечный, перестанет действовать через сутки (или как настроено в API))
var token = GetAccessToken(baseUrl, APIClientId, APIClientSecret, "admin", "1234").GetAwaiter().GetResult();
Console.WriteLine(token.AccessToken);
```

В новых версиях:

```
string baseUrl = "http://localhost:9000/";

//по протоколу OAuth 2.0 описываем что за приложение обращается к API
string APIClientId = "Идентификатор вашего приложения (ClientId), который
добавляется в BioTime в разделе <Управление-Настройки системы-Приложения> ";
string APIClientSecret = "Секретный ключ вашего приложения (ClientSecret), который
добавляется в BioTime в разделе <Управление-Настройки системы-Приложения>";

//получаем токен доступа (он не вечный, перестанет действовать через сутки (или как
настроено в API))
var token = GetAccessToken(baseUrl, APIClientId, APIClientSecret, "admin",
"1234").GetAwaiter().GetResult();
Console.WriteLine(token.AccessToken)
```

Метода для получения токена на C#

```
public static async Task<AccessTokenResponse> GetAccessToken( string baseUrl,
string clientId, string clientSecret, string login, string passwordHash )
{
    HttpClient client = new HttpClient();
    using (HttpRequestMessage request = new
HttpRequestMessage(HttpMethod.Post, baseUrl + "oauth/token"))
    {
        var values = new List<KeyValuePair<string, string>> {
            new KeyValuePair<string, string>(GrantType, Password),
            new KeyValuePair<string, string>(Username, login),
            new KeyValuePair<string, string>(Password, passwordHash),
            // new KeyValuePair<string, string>(Constants.Parameters.Scope,
"Read Write")
        };
        // client_id and client_secret:
http://tools.ietf.org/html/rfc6749#section-2.3.1
        request.Headers.Authorization = new
AuthenticationHeaderValue("Basic", EncodeToBase64(string.Format("{0}:{1}",
clientId, clientSecret)));
        request.Content = new FormUrlEncodedContent(values);
        HttpResponseMessage response = await client.SendAsync(request);
        response.EnsureSuccessStatusCode();
        var resultJSON = await response.Content.ReadAsStringAsync();
        return
Newtonsoft.Json.JsonConvert.DeserializeObject<AccessTokenResponse>(resultJSON);
    }
}
```

Пример для внешней обработки 1С

```
Соединение = Новый HTTPСоединение(АдресБТ, ПортБТ);
Запрос = Новый HTTPЗапрос();
    Запрос.АдресРесурса = "oauth/token";
    Запрос.Заголовки.Вставить("Authorization", "Basic
V2ViTWFuYWdlcjp0aGZlUmFuZG9tU3RyaW5nVXNlZFRvU2VjdXJlQ29tcGFueUFQSQ==");
    Запрос.Заголовки.Вставить("Content-Type", "application/x-www-form-
urlencoded");
    Стр =
"grant_type=password&username="+СокрЛП(ИмяПользователяБТ)+"&password="+СокрЛП(Парол
ьПользователяБТ);
    Запрос.УстановитьТелоИзСтроки(Стр);
    ОтветHTTP = Соединение.ВызватьHTTPМетод("POST", Запрос);
```

В заголовке Authorization после Basic – это закодированные в base64 ClientID и ClientSecret.

Ранее использовались:

```
ClientID = WebManager
ClientSecret = thisRandomStringUsedToSecureCompanyAPI
```

Теперь требуется задание индивидуальных ClientID и ClientSecret в интерфейсе BioTime на странице “Управление / Настройки системы / Приложения”.