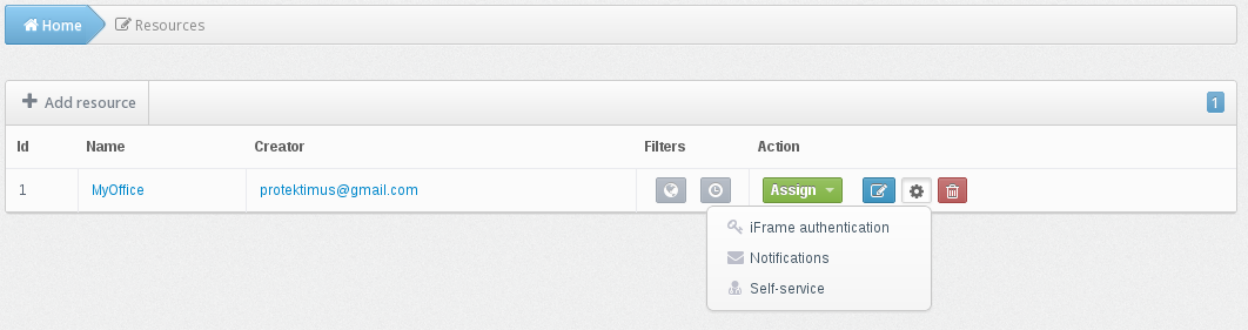
IFrame Widget Setup for Easy Integration

Version 1.00
26 May 2014

IFrame Widget Setup for Easy Integration

You can use the IFrame widget to authenticate users. When using the IFrame widget, a user will be prompted to enter the authentication details in the format that we developed. After the details are entered, we will perform authentication and send a notification on its results to the address that you specified.

To enable the mechanism for your resource, go to the IFrame Authentication section, which you can access by clicking on the gearwheel icon button in the resource list.



A page will open where you will need to enter the following parameters:

- Success URL a notification of successful user authentication will be sent to this address.
- Fail URL a notification will be sent to this address if a user enters the details incorrectly, exceeding the maximum number of failed attempts for the current resource.
- Password Protectimus will use this password to confirm notifications sent to the Success URL or Fail URL.
- Password Confirmation the password above is used to confirm that the details are entered correctly.
- Activity it allows or prohibits the operation of the mechanism on this resource.

When the setup is complete, place IFrame for user login on your page. It should use this address:

https://api.protectimus.com/plugins/authentication

and transmit the following parameters: client_id, auth_type, resource_id resource_name user_id user_login token_id.

Some parameters are required, and some are not.

client_id a required parameter that represents your company's ID provided in the Personal Information tab on the profile page: http://service.protectimus.com/profile

auth_type a required parameter that specifies the user authentication type that you want to use. The following values are acceptable for this parameter:

0 token authentication, i.e. verification of the provided OTP's validity. It corresponds to the authenticate/token method in the API. When this type is used, the token must be assigned to the resource.

1 - user authentication with a password. It corresponds to the authenticate/userpassword method in the API. When this type is used, the user must be assigned to the resource, and a password must be set for the user.

2 user authentication with OTP. It corresponds to the authenticate/usertoken method in the API. When this type is used, the token must be assigned to the resource, together with the user.

3 user authentication with a static password and OTP. It corresponds to the authenticate/userpasswordtoken method in the API. If you want to use this method, the user must be assigned to the resource, together with the token.

resource_id and resource_name the resource's identifier and name. It is required to specify one of these parameters so that we would be able to identify on which resource user authentication needs to be performed.

The **user_login** or **user_id** parameter identifies the user for which authentication will be performed. If one of these parameters is not provided, a user will be prompted to enter the login independently.

token_id the identifier of the token from which the OTP needs to be verified. This parameter is only required for authenticating a token that is not connected with the user on the resource (authentication type: **auth_type**=0).

iFrame will generate a set of fields that a user will need to fill out to be authenticated on your resource.

If authentication is successful, we will send you a POST request to the address that you provided as the Success URL in the settings for the resource iFrame authentication.

If a user enters the authentication details incorrectly, exceeding the maximum number of failed attempts for the current resource, this user will be blocked, and a POST request will be sent to the address that you provided as the Fail URL.

Thus, a user will be redirected to the addresses that you provided, depending on the authentication results.

The POST request will include all the parameters that you provided for iFrame, and the following parameters will be added:

datetime - the time when authentication was performed in the following format: "yyyyMMdd HH:mm:ss";

auth_user_id the identifier of the user for which authentication was performed;

auth_user_login the login of the user for which authentication was performed;

auth_token_id the identifier of the token for which authentication was performed ;

hash_source the string that will be converted with the HMAC algorithm to confirm the consistency of our message. This string consists of a set of the remaining parameters separated by semicolons;

hash the result of the conversion of the hash_source string with the HMAC algorithm. This parameter is provided in the HEX format. See more detailed explanation on how hashing is performed below.

The auth_user_id and auth_token_login parameters will not be included if only a token is involved in authentication. If authentication was performed without using a token, the auth_token_id parameter will not be included.

To confirm the consistency of our messages, we create a hash_source string consisting of a set of all the parameters that you provided divided by semicolons. These parameters are included in the string in the following order:

```
client_id;auth_user_id;auth_user_login;auth_token_id;resource_id;resource_name;user_id;user_login;toke
n_id;custom params;datetime
```

We convert the resulting hash_source string with the HMAC algorithm, using SHA1 as the hashing functions and the password that you provided in the settings for the iFrame authentication of the current resource as the password. The result provided by the algorithm is converted into the HEX format and transmitted to you in the hash parameter.

Let’s look at it in more detail using this example:

You have created a user with a password, assigned a token to this user and assigned this token together with this user to a resource named “MyOffice”. Your company’s ID provided in the profile is “1”. To authenticate this user with a password and OTP on this resource, you only need to set up iFrame authentication on the resource and insert the following code on the login page:

```
<iframe
src="https://api.protectimus.com/plugins/authentication?client_id=1&resource_name=MyOffice&auth_type
=3">
  Your browser doesn't support the iFrame
</iframe>
```

The user will see a window in which the user will need to enter their login and password; if the password is entered correctly, the user will be prompted to enter the OTP.

The user enters “protector” as the login, their password, and the OTP generated by their token.

After that, we will send you a POST request that will contain the following parameters:

- auth_token_id = 5
- auth_user_id = 5
- auth_user_login = protector
- client_id = 1
- datetime = 20140514 18:00:47
- hash = 98548B070F5A4A3D2719FE3FE39146C2174060E6
- hash_source = 1;5;protector;5;MyOffice;20140514 18:00:47
- resource_name = MyOffice

In the iFrame authentication settings, the “pass” password is provided; as a result of the conversion of the following string: 1;5;protector;5;MyOffice;20140514 18:00:47 with the HMAC algorithm based on SHA1 using the “pass” password, we get the following result in the HEX format:

```
98548B070F5A4A3D2719FE3FE39146C2174060E6
```