

Documento original em:

<http://tdn.totvs.com/pages/viewpage.action?pagelId=268570110>

Configuração do OAuth 1.0A

Para maiores informações sobre a especificação do OAuth1.0 A acesse o link abaixo:

<https://oauth.net/core/1.0a/>

Configuração do Appserver.ini e das Políticas

1 - No arquivo appserver.ini inclua as configurações necessárias como mostra o passo a passo no link abaixo:

[Configuração arquivo Appserver.ini](#)

2. Acessar SIGACFG > Usuário > Senhas > Política

3 .Abra a aba integrações

4. Preencha os campos Nome, Endereço IP HTTPS e porta HTTPS, de acordo com o que foi configurado no arquivo appserver.ini.

TOTVS | Configurador

TOTVS Srea T Manufatura MSSQL P12_rea Administrador 20/10/2015 Teste / Status

Regras de senha - Poltica de Segurana

Outras Altes Cancelar Salvar

Poltica de Segurana Conexes Outros Integraes

Client OAuth Integra SAML Client Pblica SCM Client Privada

teste

Informaes OAuth

Client* Chave Pblica*

teste

Chave Privada* URL de requisio de token

URL de autorizao URL de acesso

URL de requisio de token (REST) URL de autorizao (REST)

URL de autorizao (REST)

Configuraes do HTTPS

Endereo IP HTTPS* Porta HTTPS*

localhost 8082

Enviar E-mail Reset de Acesso Gerar Chaves

5 .Em seguida selecione a opo Gerar Chaves.

TOTVS | Configurador

TOTVS Srea T Manufatura MSSQL P12_rea Administrador 20/10/2015 Teste / Status

Regras de senha - Poltica de Segurana

Outras Altes Cancelar Salvar

Poltica de Segurana Conexes Outros Integraes

Client OAuth Integra SAML Client Pblica SCM Client Privada

teste

Informaes OAuth

Client* Chave Pblica*

teste

Chave Privada* URL de requisio de token

URL de autorizao URL de acesso

URL de requisio de token (REST) URL de autorizao (REST)

URL de autorizao (REST)

Configuraes do HTTPS

Endereo IP HTTPS* Porta HTTPS*

localhost 8082

Enviar E-mail Reset de Acesso Gerar Chaves

salve a configurao.

ATENÇÃO: existem alguns aplicativos disponíveis para teste de APIs, procure mais informações nos sites dos desenvolvedores.

Ex: Postman , Advanced Rest Client

O Exemplo a seguir foi realizado utilizando o app do Google Chrome: Advanced REST Client App

1. No Advanced REST Client temos a possibilidade de testar um serviço de REST passando uma URL e o cabeçalho do HTTP.

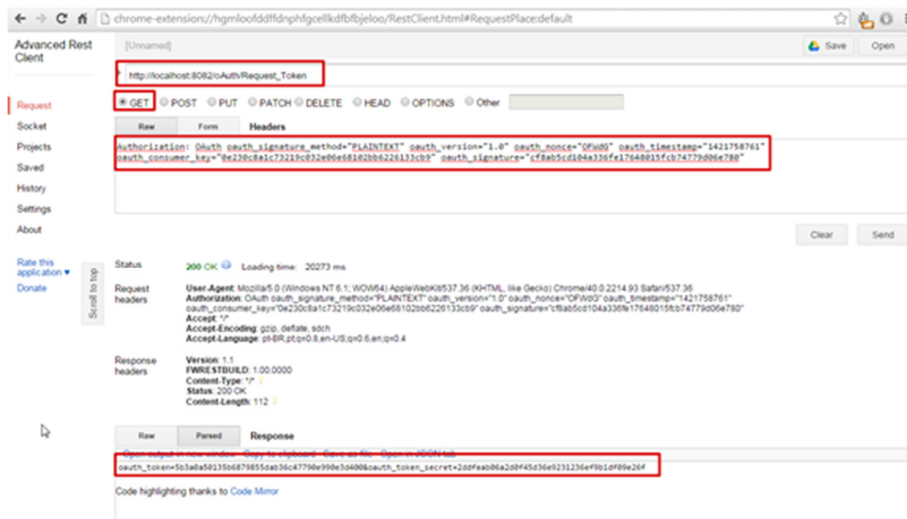
Como primeiro passo vamos passar:

- URL = http://localhost:8082/oauth/Request_Token
- Header = Authorization: OAuth oauth_signature_method="PLAINTEXT" oauth_version="1.0" oauth_nonce="OFWdG" oauth_timestamp="1421758761" oauth_consumer_key="<CHAVE PÚBLICA*>" oauth_signature="<CHAVE PRIVADA***>"

* Chave Pública gerada nas Políticas

**** Chave Privada gerada nas Políticas**

Esta chamada nos retornará o Token de Requisição, utilizado posteriormente para fazer o login.



- 2 .Como segundo passo devemos passar:

- URL = <http://localhost:8082/oAuth/Login>

- Header = Authorization: OAuth oauth_signature_method="PLAINTEXT" oauth_version="1.0" oauth_nonce="OFWdG" oauth_timestamp="1421758761" oauth_consumer_key="<CHAVE PÚBLICA>" oauth_signature="<CHAVE PRIVADA>" user="<USUARIO>" password="<SENHA>" oauth_token="<PRIMEIRO RETORNO DO REST*>" oauth_token_secret="<SEGUNDO RETORNO DO REST*>"

* No nosso teste o primeiro retorno é:

Open output in new window Copy to clipboard Save as file
 oauth_token=5b3a0a50135b6879855dab36c47790e990e3d4008c

** No nosso teste o segundo retorno é:

as file Open in JSON tab
 400 oauth_token_secret=2ddf06a2d0f45d36e9231236ef9b1df09e26f

Esta chamada nos retornará o mesmo Token, apenas confirmando que o login foi feito com sucesso:

The screenshot shows the Advanced Rest Client interface. The URL is `http://localhost:8082/oAuth/Login`. The request method is GET. The response status is 200 OK, and the response body contains the OAuth token and secret.

```

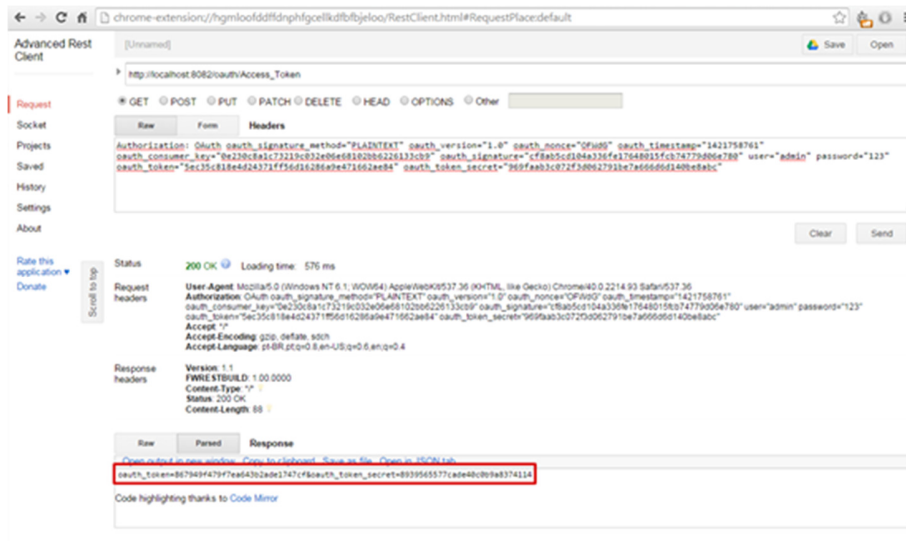
Raw
{
  "oauth_token": "5b3a0a50135b6879855dab36c47790e990e3d4008c",
  "oauth_token_secret": "2ddf06a2d0f45d36e9231236ef9b1df09e26f"
}

```

3. Para o passo seguinte deveremos passar:

- URL = http://localhost:8082/oAuth/Access_Token
- Header = Manter o mesmo utilizado no Login

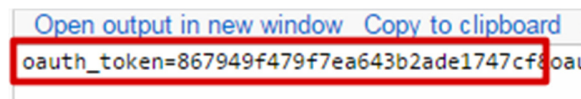
Esta chamada nos retornará o Token de acesso:



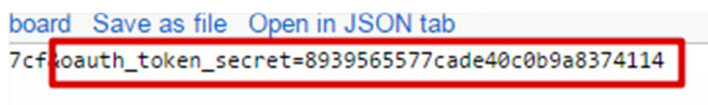
4. Para o passo seguinte deveremos passar:

- URL = <http://localhost:8082/Ping>
- Header = Authorization: OAuth oauth_signature_method="PLAINTEXT" oauth_version="1.0" oauth_nonce="OFWdG" oauth_timestamp="1421758761" oauth_consumer_key="<CHAVE PÚBLICA>" oauth_signature="<CHAVE PRIVADA>" user="<USUARIO>" password="<SENHA>" oauth_token="<PRIMEIRO RETORNO DO ULTIMO REST*>" oauth_token_secret="<SEGUNDO RETORNO DO ULTIMO REST*>"

* No nosso teste o primeiro retorno é:



** No nosso teste o segundo retorno é:



A resposta desse ultimo Get deverá ser:

chrome-extension://hgmloofddfnghhgoellkbfbjeloq/RestClient.html#RequestPlacedefault

Advanced Rest Client

[Unnamed]

Save Open

Request

Socket

Projects

Saved

History

Settings

About

Rate this application

Donate

Send to log

Raw Form Headers

Authorization: OAuth oauth_signature_method="PLAINTEXT" oauth_version="1.0" oauth_nonce="0fa09" oauth_timestamp="1421758761" oauth_consumer_key="fe2208e8c72232c032e6e681820a2226333c8" oauth_signature="cf8a5c3d04a326f4376488355c3a7736956780" user="admin" password="123" oauth_token="867849f4707fa643b2ade1747cf" oauth_token_secret="8939565577cade68c09a8374114"

Clear Send

Status 200 OK Loading time: 897 ms

Request headers

User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64; AppleWebKit/537.36 (KHTML, like Gecko) Chrome/40.0.2214.93 Safari/537.36

Authorization: OAuth oauth_signature_method="PLAINTEXT" oauth_version="1.0" oauth_nonce="0fa09" oauth_timestamp="1421758761" oauth_consumer_key="fe2208e8c72232c032e6e681820a2226333c8" oauth_signature="cf8a5c3d04a326f4376488355c3a7736956780" user="admin" password="123" oauth_token="867849f4707fa643b2ade1747cf" oauth_token_secret="8939565577cade68c09a8374114"

Accept: */*

Accept-Encoding: gzip, deflate, sdch

Accept-Language: zh-CN,zh;q=0.8,en-US;q=0.6,en;q=0.4

Response headers

Version: 1.1

FWRETSBUILD: 1.00.0000

Content-Type: application/json

Status: 200 OK

Content-Length: 50

Raw JSON Response

Copy to clipboard Save as file

```
{
  serverStatus: "UNAVAILABLE",
  syncStatus: "NONE"
}
```