

SMAIL/ Gmail SMTP OAuth2 認証の利用・設定

2020/11/01 初版

2021/06/22 第2版

— 目次 —

1. 概要	
1) 準備	3
2) 設定	3
3) 動作確認	3
2. 詳細	
①Google API Console で OAuth 2.0 クライアント用の[クライアント ID]および [クライアントシークレット]の取得	4
1) 新しいプロジェクトの作成	4
2) OAuth 同意画面を[作成]	5
3) OAuth クライアント ID を作成	6
4) デスクトップ のクライアント ID の JSON ファイルのダウンロード	7
②Google OAuth API で[リフレッシュトークン]を取得	8
1) SmailRefreshToken.exe を起動させます	8
2) ブラウザーでログイン許可設定をします	9
③SmailRefreshToken.exe で正しくリフレッシュトークンが取得できれば、 実行フォルダーに、smail_refreshtoken.ini が生成されます	10
④smail_refreshtoken.ini を smail.exe と同じフォルダーにコピーし、smail.exe を起動して、 smtp 認証に OAUTH-XOAUTH2 を選択すれば利用できます	10
3. 動作確認	12

1. 概要

1) 準備

Google アカウントが必要です。(本参考例は Gmail 無料アカウントです。G Suite アカウントもほぼ同じです)

※実際の設定方法は2. 詳細を参照して下さい。

- ① Google API Console で OAuth 2.0 クライアント用の[クライアント ID]と[クライアントシークレット]の取得をブラウザアクセスで行います。
- ② Google OAuth API で[リフレッシュトークン]を取得
付属の SmailRefreshToken.exe で行います。認証はデフォルトのブラウザが起動します。

ここまでで、SMAIL で OAuth2 認証を利用するための各種情報が取得できます。

この操作は一度のみです。ただし、Google アカウントのパスワード変更等を行った場合はリフレッシュトークンの再取得 (SmailRefreshToken.exe) が必要です。

2) 設定

- ③ SmailRefreshToken.exe で正しくリフレッシュトークンが取得できれば、実行フォルダーに、smail_refreshtoken.ini が生成されます。
- ④ smail_refreshtoken.ini を smail.exe と同じフォルダーにコピーし、smail.exe を起動して、smtp 認証に OAUTH-XOAUTH2 を選択すれば利用できます。
※SMTP ユーザ ID,SMTP パスワードを設定していても無視されます。(OAUTH-XOAUTH2 では必要が無いため)

3) 動作確認

以降、SMAIL は、リフレッシュトークンを利用してアクセストークンを自動で取得し、OAuth 2.0 認証でメールを送信できるようになります。

OAuth 2.0 認証の動作 OS:

OS	Windows Server 2008 sp2	Windows Server 2008 R2	Windows Server 2012	Windows xp	Windows 7 pro	Windows 8.1	Windows 10 pro
OAuth 2.0 認証	△(*1)	○	○	×	○	○	○

(*1)

KB4019276 パッチを利用し、TLS1.2 を利用可にすれば動作可能です。(Windows Server 2008 sp2 64BIT 版検証済み)

<https://support.microsoft.com/ja-jp/help/4074621/add-rds-support-for-tls-1-1-and-tls-1-2-in-windows-server-2008-sp2>

2. 詳細（実際の取得方法）

① Google API Console で OAuth 2.0 クライアント用の[クライアント ID]および[クライアントシークレット]の取得

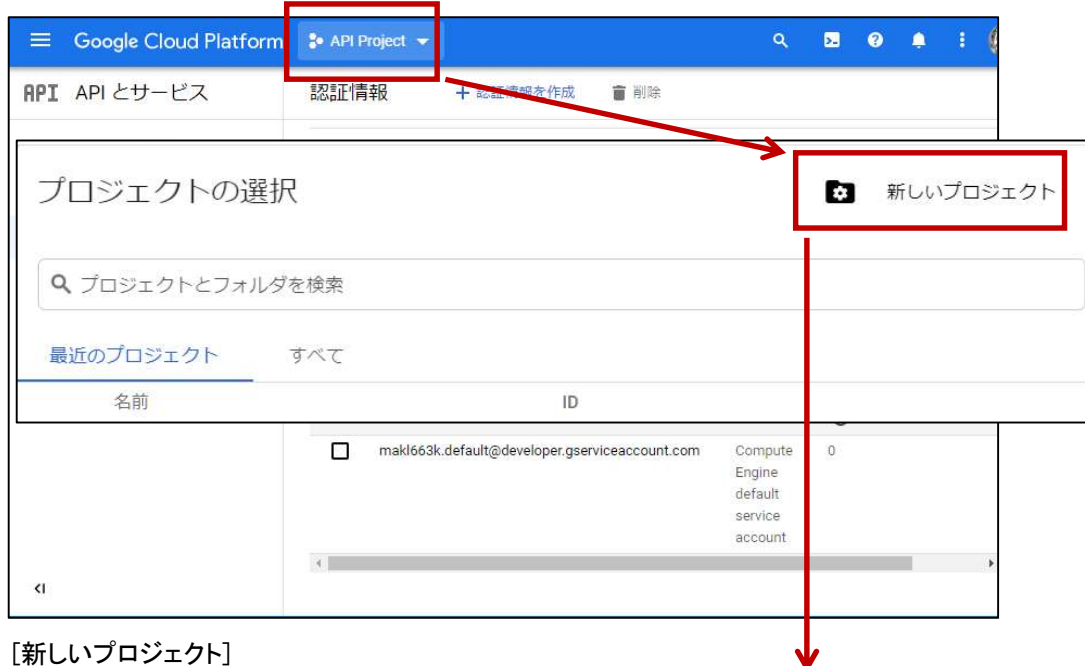
Google API Console (<https://console.cloud.google.com/apis/>)にアクセスする。

Google アカウントでログインします。

（本参考例は Gmail 無料アカウントです。G Suite アカウントもほぼ同じです。この処理は一度だけです。）

1.) 新しいプロジェクトの作成

プロジェクト名は任意の名前です。



[新しいプロジェクト]



2) OAuth 同意画面を[作成]。

- ・外部を選択。G Suite であれば [User Type]、内部が選択できます。

Google Cloud Platform API Project

API API とサービス

OAuth 同意画面

アプリをどのように構成および登録するか（ターゲットユーザーを含む）を選択します。プロジェクトに関連付けることができるアプリは 1 つだけです。

User Type

☐ 内部

組織内のユーザーのみが使用できます。検証を受けるためにアプリを送信する必要はありません。

☒ 外部

Google アカウントを持つすべてのユーザーが使用できます。

作成

- ・アプリケーション名

smail 連携：名前は任意の名前です。

Google Cloud Platform API Project

API API とサービス

OAuth 同意画面

ユーザーが認証を行う前に、この同意画面では、プライベートデータへのアクセス権を付与するかどうか、また利用規約とプライバシーポリシーへのリンクを付与するかどうかをユーザーが選択できます。このページでは、このプロジェクトのすべてのアプリケーションの同意画面を構成します。

確認ステータス
非公開

アプリケーション名
同意を求めるアプリの名前
smail

アプリケーションのロゴ
ユーザーがアプリを認識しやすいよう同意画面に表示される画像
アップロードするローカルファイル 参照

サポートメール
ユーザーサポートに関する同意画面に表示されます

Google API のスコープ
スコープを設定すると、アプリケーションはユーザーのプライベートデータにアクセスできるようになります。詳細
カレンダーやドライブへの完全アクセス権を付与するスコープなど、プライベートデータにかかわるスコープを追加する場合は、スコープの公開前に Google が同意画面を確認します。

email
profile
openid

スコープを追加

承認済みドメイン
デベロッパーとユーザーを保護するために、Google では、OAuth を使用して認証するアプリケーションのみに承認済みドメインの使用を許可しています。アプリケーションのリンクは承認済みドメインでホストされている必要があります。詳細
example.com
ドメインを入力し、Enter キーを押して追加します

[アプリケーション ホームページ] リンク
同意画面に表示されます。承認済みドメインでホストされている必要があります。
https:// または http://

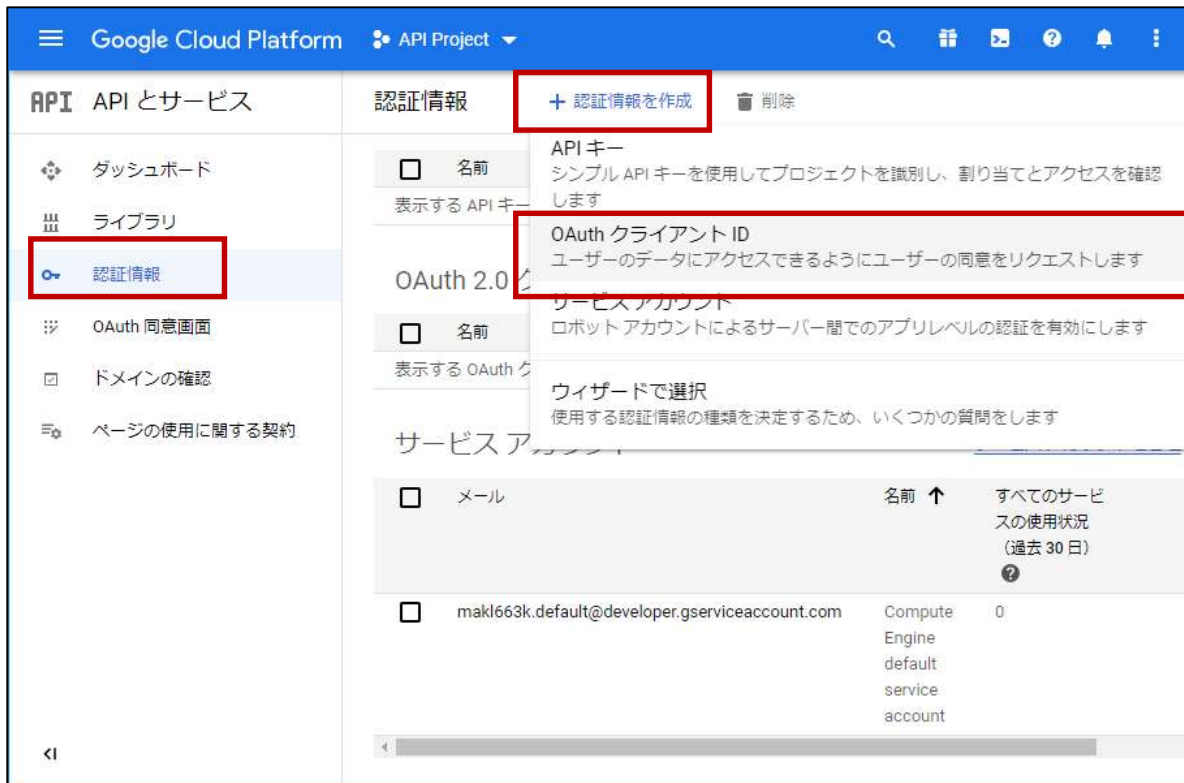
[アプリケーション プライバシーポリシー] リンク
同意画面に表示されます。承認済みドメインでホストされている必要があります。
https:// または http://

[アプリケーション 利用規約] リンク (省略可)
同意画面に表示されます。承認済みドメインでホストされている必要があります。
https:// または http://

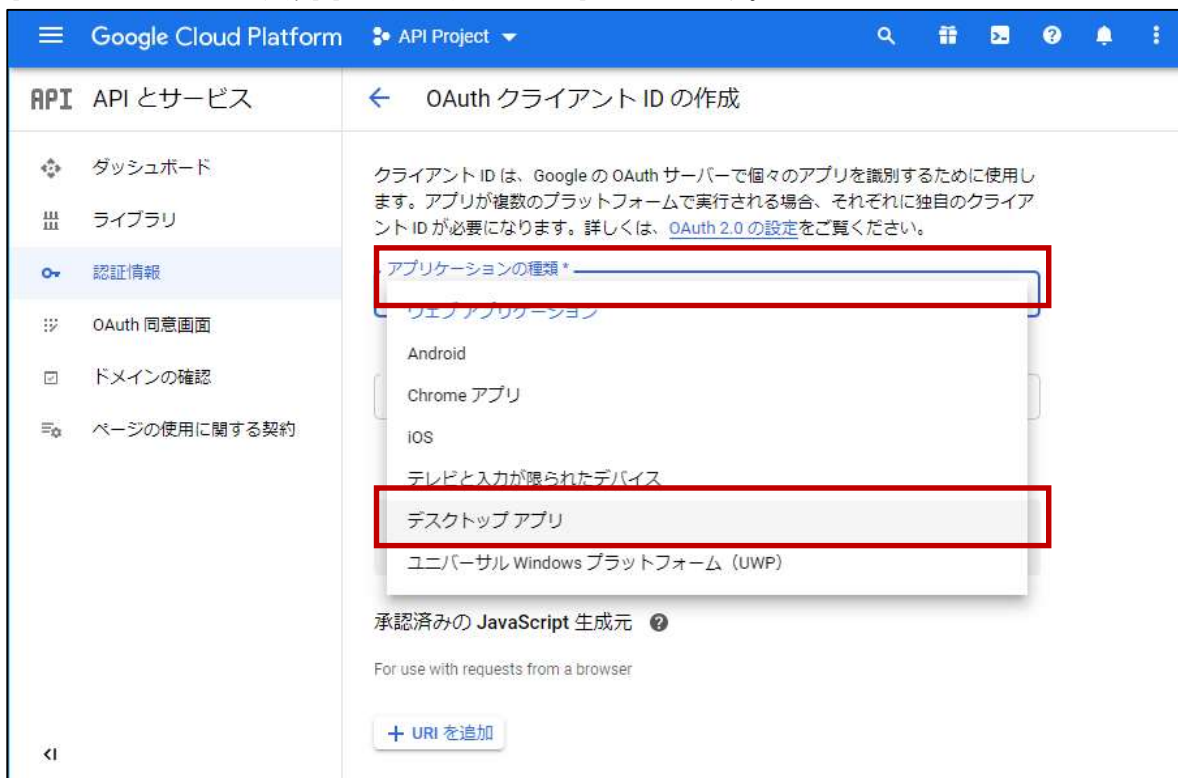
保存 確認のため送信 キャンセル

3) OAuth クライアント ID を作成

[認証情報]-[認証情報を作成]-[OAuth クライアント ID]をクリック。



[アプリケーションの種類]-[デスクトップアプリ]を選択します。



[名前]は、任意な名前を入力します。

Google Cloud Platform API Project

API API とサービス

OAuth クライアント ID の作成

クライアント ID は、Google の OAuth サーバーで個々のアプリを識別するために使用します。アプリが複数のプラットフォームで実行される場合、それぞれに独自のクライアント ID が必要になります。詳しくは、[OAuth 2.0 の設定](#)をご覧ください。

アプリケーションの種類 *
デスクトップアプリ

OAuth クライアントの種類については[こちら](#)をご覧ください

名前 *
smail クライアント

OAuth 2.0 クライアントの名前。この名前はコンソールでクライアントを識別するためにのみ使用され、エンドユーザーには表示されません。

作成 キャンセル

OAuth クライアントを作成しました

クライアント ID とシークレットには、常に API とサービスの認証情報からアクセスできます

OAuth 同意画面が確認されるまで、OAuth では**プライベートデータにかかわるスコープのログイン**が 100 回までに制限されます。公開には確認プロセスが必要になる場合があります。確認プロセスには数日を要する場合があります。

クライアント ID

クライアントシークレット

OK

4) デスクトップ のクライアント ID の JSON ファイルのダウンロード

※クライアント ID、クライアント シークレット等の必要な情報が記載されています。

Google Cloud Platform API Project

API API とサービス

デスクトップのクライアント ID

JSON をダウンロード シークレットをリセット 削除

名前 *
smail クライアント

OAuth 2.0 クライアントの名前。この名前はコンソールでクライアントを識別するためにのみ使用され、エンドユーザーには表示されません。

クライアント ID

クライアントシークレット

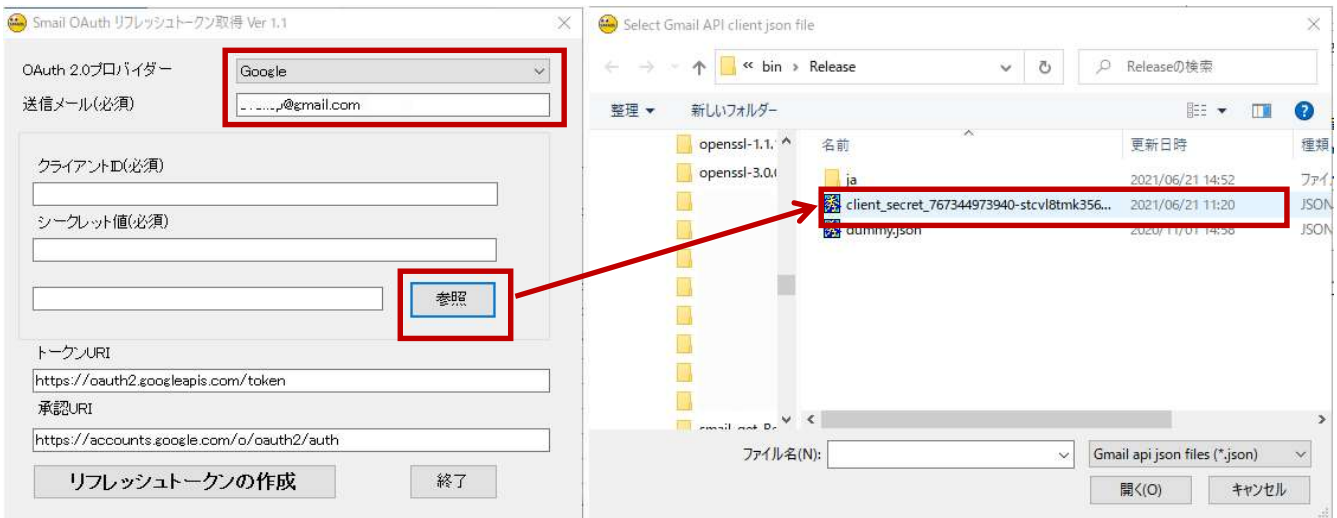
client_secret_xxxxxxxxxxgoogleusercontent.com.json

後でリフレッシュトークンを取得する際に必要なファイルです。

② Google OAuth API で[リフレッシュトークン]を取得

1) SmailRefreshToken.exe を起動させます。

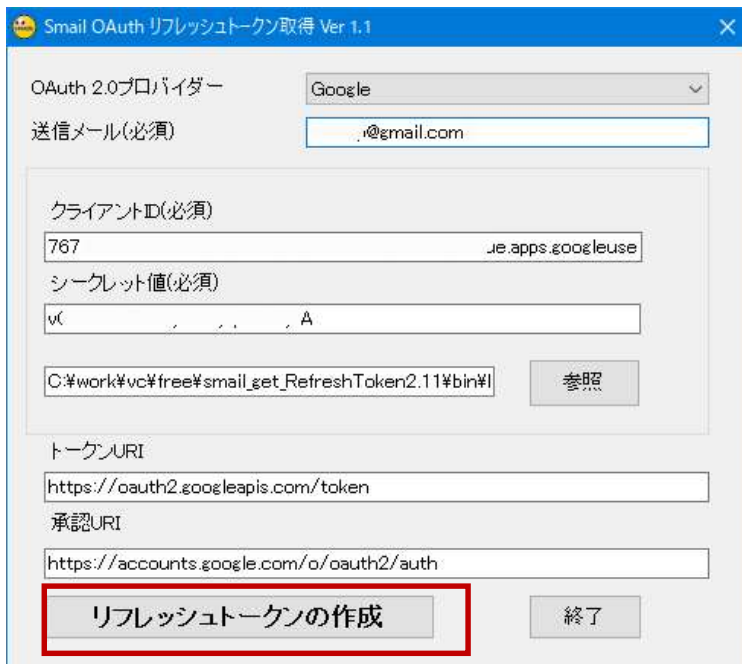
① で使用したメールアドレスと取得した設定用 J S O N ファイルを取り込みます。



正しく、J S O N ファイルが取り込めると、クライアント ID、シークレット値が表示されます。

トークン URI、承認 URI も自動的に更新されます。(通常は初期表示の内容と同じで、見た目は変わりません)

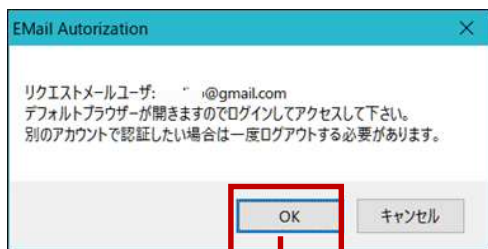
(※設定用 J S O N ファイルを使用せず、クライアント ID、シークレット値を直接入力しても問題ありません)



[リフレッシュトークンの作成] をクリックします。

(このボタンを押下することで、Google API サーバへ承認接続し、リフレッシュトークンを取得します。)

確認ダイアログが表示されますので、[OK] ボタンをクリックして下さい。（リクエストのメールアドレスは正しいか、再確認して下さい）



デフォルトのブラウザが起動します。

「このアプリは確認されていません」
と表示された場合は、[詳細]をクリック



[smail 連携（安全ではないページ）
に移動をクリック

2) ブラウザーでログイン許可設定をします。





アカウントが複数ある場合は①で使用したアカウントを選択して下さい。

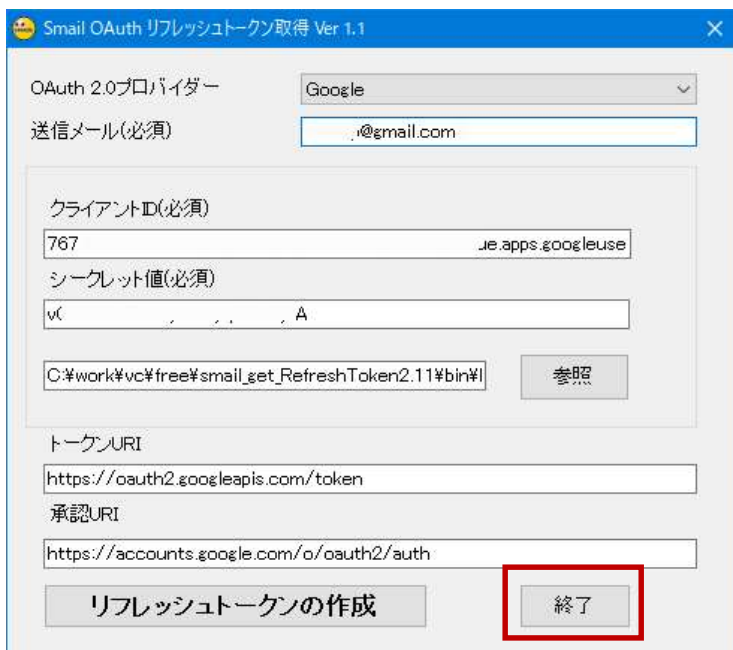


③ SmailRefreshToken.exe で正しくリフレッシュトークンが取得できれば、実行フォルダーに、smail_refreshtoken.ini が生成されます。

- ・成功するとデフォルトのブラウザに以下のように表示されます。



- [OK]を押して[終了]させてください。



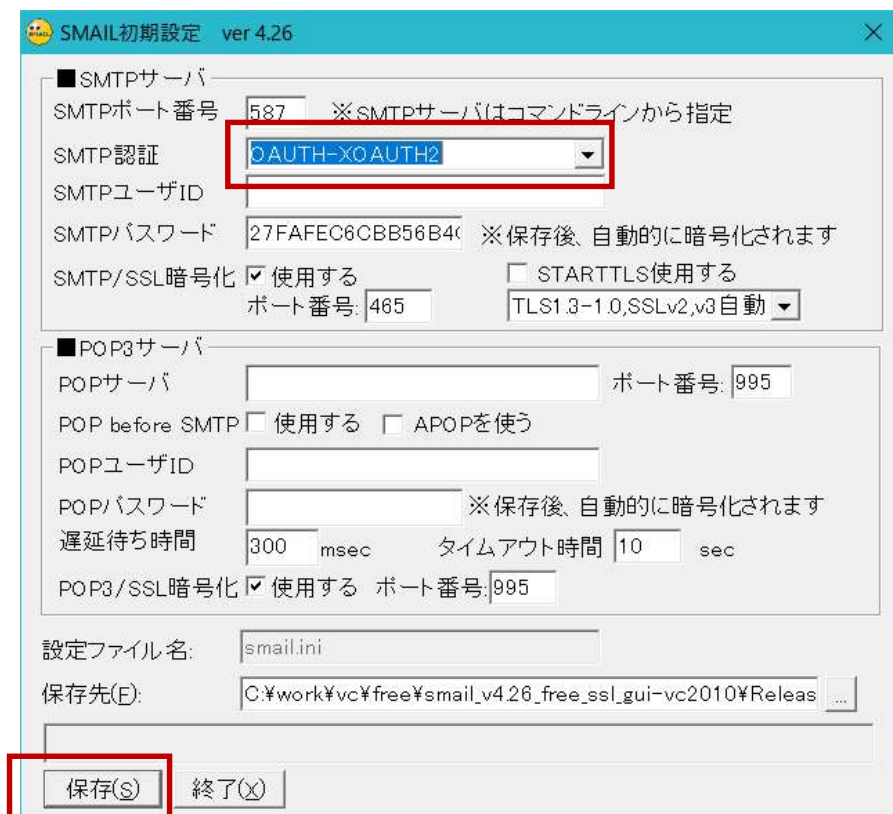
通常は一度だけ取得すれば OK ですが、Google アカウントのパスワード変更の時は再取得が必要となります。

④smail_refreshtoken.ini を smail.exe と同じフォルダーにコピーし、smail.exe を起動して、smtp 認証に OAUTH-XOAUTH2 を選択すれば利用できます。

- 以後、`smail refreshtoken.ini` のファイルが `SMAIL` を実行する PC に必要です。

```
[OAUTH2]
UserID=???????@gmail.com
ClientID=8888888888888-stcvxxxxxxxxxxxxxxxxxxapps.googleusercontent.com
ClientSecret=vXXXXXXXXXXXXXXXXXXXXXyq0VUOjLA
RefreshToken=1//0exxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
TokenURI=https://oauth2.googleapis.com/token
```

2) SMAIL.exe を起動し SMTP 認証 : OAUTH-XOAUTH2 を選択して保存して下さい。



[保存]して終了して下さい。

refresh token.ini が見つからない場合は以下の警告が出ます。正しく INI ファイルを作製して下さい。



smail.ini に設定が保存されます。(既存の smail.ini に引き継がれます)

3. 動作確認

例) c>smail -hsmtp.gmail.com -s"TEST MAIL" -fxxxxxx@gmail.com dummy@example.jp -d -i

```
#----- SMAIL 4.26 Information By Eva -----
#設定ファイル      : [C:\work\vc\free\smail_v4.26_free_ssl_gui-vc2010\Release\smail.ini]
#POP before SMTP   : 使わない
#APOP              : 使わない
#SMTP 認証         : 使う OAUTH XAUTH2 を利用
#設定ファイル (OAUTH 用) : [C:\work\vc\free\smail_v4.26_free_ssl_gui-vc2010\Release\smail_refresh token.ini]
#API アカウント    : [xxxxxx@gmail.com]
#SMTP アカウント   :
#SMTPS (SMTP over SSL) : 使う [OpenSSL 1.1.1h]
#SSL 方式          : TLS1.3-1.0, SSLv2, v3 自動
#STARTTLS          : 使わない
#本文ファイル      : なし
#添付ファイル      : なし
#メールサーバ      : smtp.gmail.com
```

```

#メール送り元      : xxxxx@gmail.com
#メール送り先      : dummy@example.jp
#サブジェクト内容   : TEST MAIL
#キャラクターセット : ISO-2022-JP
#エンコーディング   : 7bit
SSL connection using TLS_AES_256_GCM_SHA384 - TLSv1.3
X509 subject_name: /C=US/ST=California/L=Mountain View/O=Google LLC/CN=smtp.gmail.com
X509 issuer_name: /C=US/O=Google Trust Services/CN=GTS CA 101
Recv:[52 bytes] 220 smtp.gmail.com ESMTP 10sm8435678pjt.50 - gsmt
Send: EHLO xxxxxx
Recv:[220 bytes] 250-smtp.gmail.com at your service, [xx.xx.xx.xx]
250-SIZE 35882577
250-8BITMIME
250-AUTH LOGIN PLAIN XOAUTH2 PLAIN-CLIENTTOKEN OAUTHBEARER XOAUTH
250-ENHANCEDSTATUSCODES
250-PIPELINING
250-CHUNKING
250 SMTPUTF8
[250 message ok!]
[Try get access token from https://oauth2.googleapis.com/token] ; 認証サーバにアクセストークン要求
[access_token=ya29.AOAfh6SMDzH_2_0....] ; アクセストークン取得
Send: AUTH XOAUTH2 ***** ; XOAUTH2 で接続要求
Recv:[20 bytes] 235 2.7.0 Accepted
Send: MAIL FROM:<XXXXXX@gmail.com>
Recv:[40 bytes] 250 2.1.0 OK 10sm8435678pjt.50 - gsmt
Send: RCPT TO:<xxxxx@xxxxxxxxxxxx.jp>
Recv:[40 bytes] 250 2.1.5 OK 10sm8435678pjt.50 - gsmt
Send: DATA
Recv:[41 bytes] 354 Go ahead 10sm8435678pjt.50 - gsmt
Send: From: XXXXXXX@gmail.com
To: dummy@example.jp
Subject: TEST MAIL
X-Mailer: SMAIL 4.26
Mime-Version: 1.0
Content-Type: text/plain; charset=ISO-2022-JP
Content-Transfer-Encoding: 7bit
Date: Sun, 01 Nov 2020 15:44:39 +0900
Send:
#//メールヘッダー送信完了 (1.609sec)....
Send:
.
Recv:[52 bytes] 250 2.0.0 OK 1604213082 10sm8435678pjt.50 - gsmt
Send: QUIT
Recv:[56 bytes] 221 2.0.0 closing connection 10sm8435678pjt.50 - gsmt

C:¥work¥vc¥free¥smail_v4.26_free_ssl_gui-vc2010¥Release>

```

・SMAILはSSL通信にOPENSSLライブラリーを使っていますが、OAUTHのアクセストークン取得部分のみ、googleへのhttpsプロトコルは、標準Windows関数を使用しております。従って、使用端末がtls1.2以降をサポートしている必要があります。

OAuth 2.0 認証の動作 OS:

OS	Windows Server 2008 sp2	Windows Server 2008 R2	Windows Server 2012	Windows xp	Windows 7 pro	Windows 8.1	Windows 10 pro
OAuth 2.0 認証	△(*1)	○	○	×	○	○	○

(*1)

KB4019276 パッチを利用し、TLS1.2 を利用可にすれば動作可能です。(Windows Server 2008 sp2 64BIT 版検証済み)

<https://support.microsoft.com/ja-jp/help/4074621/add-rds-support-for-tls-1-1-and-tls-1-2-in-windows-server-2008-sp2>

・プロキシ対応です。httpsアクセスに端末のプロキシ情報を利用しています。

OAUTH 関連エラー例

例) smail_refreshtoken.ini

```
[250 message ok!]  
[Try get access token from https://oauth2.googleapis.com/token]  
Error HttpSendRequest [12157]  
ERR-0028:OAUTH 認証サーバからアクセストークンが取得できませんでした。
```

12157 セキュリティ・チャネル・エラー

SSL ライブラリのロード中に、アプリケーションに内部エラーが発生しました

```
[250 message ok!]  
[Try get access token from https://oauth2.googleapis.com/token]  
Error HttpSendRequest [12045]  
ERR-0028:OAUTH 認証サーバからアクセストークンが取得できませんでした。
```

12045 無効な認証局です

サーバーSSL 証明書は、未知または無効な認証局から発行されたものです。

PC が TLS1.2 をサポートしていない場合のエラーです。

```
[250 message ok!]  
[Try get access token from https://oauth2.googleapis.com/token]  
Error HttpSendRequest [12029]  
ERR-0028:OAUTH 認証サーバからアクセストークンが取得できませんでした。
```

12029 サーバー接続の試みに失敗しました

プロキシ設定が間違っており接続できない場合もこのエラーが出ます。