MQTT Client プロバイダ

(Publisher)

Version 1.3.1

ユーザーズ ガイド

August 30 2021

【備考】

【改版履歴】

バージョン	日付	内容
1.0.0	2016-12-15	初版.
1.1.0	2017-07-19	Publish するメッセージのデータ型を文字列(VT_BSTR)と BYTE 配列(VT_UI1 | VT_ARRAY)から選択するように変更.
1.2.0	2018-05-29	Mosquitto ライブラリの初期化・終了部分に排他制御追加. SSL 通信対応.
1.2.1	2018-07-31	AddController バグ修正.
1.2.2	2019-07-24	トピックとメッセージのコードページ対応
1.3.0	2020-05-21	セキュア通信時のクライアント認証を任意に変更.
1.3.0	2020-08-31	エラーコード 0x8010000F の原因&対処方法を追記
1.3.1	2021-08-30	内部コード最適化.
	2021-11-09	誤字修正.

【対応機器】

機種	バージョン	注意事項

目次

1. はじめに

本書は，MQTT Broker に対し MQTT Client として Publish を行う CAO プロバイダのユーザーズガイドです.

本書で扱う CAO プロバイダ(CaoProvMQTT_PUBLISHER.dll)を PUBLISHER プロバイダと呼びます.

第 2 章に PUBLISHER プロバイダの概要，変数の詳細を記載しています.

PUBLISHER プロバイダでは，通信に使用する MQTT プロトコルを利用するためにオープンソースの Mosquitto ライブラリを使用しています.

MQTT プロトコルや Mosquitto ライブラリの詳細に関しては，それぞれ以下のサイトを参照してください.

MQTTClient の SUBSCRIBER としての動作は，本プロバイダとは別に SUBSCRIBER プロバイダ（CaoProvMQTT_SUBSCRIBER.dll）として実装されています.

SUBSCRIBER プロバイダの詳細については，SUBSCRIBER プロバイダのユーザーズガイドを参照してください.

このプロバイダを使用するためには，「Visual Studio 2015 の Visual C++ 再頒布可能パッケージ」が必要です.

[関連サイトリンク]

http://mqtt.org/

http://mosquitto.org/

2. プロバイダの概要

2.1. 概要

PUBLISHER プロバイダは, Mosquitto ライブラリを用いて MQTT Broker に対し Publish を行う CAO プロバイダです. そのファイル形式は DLL(Dynamic Link Library)であり, CAO エンジンから使用時に動的にロードされます. PUBLISHER プロバイダを使用するにあたっては ORiN2SDK をインストールするか, 下表を参照して手作業でレジストリ登録を行う必要があります.

表 2-1 PUBLISHER プロバイダ

ファイル名	CaoProvMQTT_PUBLISHER.dll
ProgID	CaoProv.MQTT_PUBLISHER
レジストリ登録	regsvr32 CaoProvMQTT_PUBLISHER.dll
レジストリ登録の抹消	regsvr32 /u CaoProvMQTT_PUBLISHER.dll

2.2. メソッド・プロパティ

2.2.1. CaoWorkspace::AddController メソッド

PUBLISHER プロバイダは AddController 時に通信用の接続パラメータを参照し，MQTT Broker との接続を行います．(MQTT Client として動作します)

書式　AddController(<bstrCtrlName:BSTR>,<bstrProvName:BSTR>,
<bstrPCName:BSTR>,<bstrOption:BSTR>))

bstrCtrlName : [in] コントローラ名
bstrProvName : [in] プロバイダ名．固定値 = ” CaoProv.MQTT_PUBLISHER”
bstrPcName : [in] プロバイダの実行マシン名
bstrOption : [in] オプション文字列

以下にオプション文字列に指定するリストを示します．

表 2-2 CaoWorkspace::AddController のオプション文字列

オプション [1]	説明
Conn=<host>[:<port>[:<keepalive>]]]	必須．接続パラメータ． (参照：2.2.1.1)
ID[=<ID 文字列>]	クライアントの ID 文字列． (省略時は GUID を元にユニークな ID を生成)
User[=<ユーザ名>[:<パスワード>]]	ユーザ/パスワード認証設定． (省略時は指定なし)
MqttProtocol[=<version>]	通信に用いる MQTT プロトコルバージョン． (参照：2.2.1.2)
Will[=<topic>:<payload>[:<qos>[:<retain>]]]	Will 設定． (参照：2.2.1.3)
ConnTimeout[=<タイムアウト時間>]	接続時のタイムアウト時間．(ミリ秒) （デフォルト：500）
Timeout[=<タイムアウト時間>]	要求送信時のタイムアウト時間．(ミリ秒) （デフォルト：500）
CA[=<証明書機関（CA）証明書ファイル名>][2]	証明機関（CA）の証明書ファイル名を指定します．（デフォルト：空文字）
Certificate[=<証明書ファイル名>][2]	証明書ファイル名を指定します．（デフォルト：空

[1] 角括弧(“[ ]”)内は省略可能を示します．また，各パラメータの解説中の下線部はオプションを指定しなかったときのデフォルト値になります．
[2] セキュア通信は CA オプションを指定した場合に行います．クライアント認証も行う場合は Certificate，PrivateKey，PKPassword を指定してください．

	文字)
PrivateKey[=<秘密鍵ファイル名>][2]	秘密鍵ファイル名を指定します.(デフォルト:空文字)
PKPassword[=<秘密鍵パスワード>][2]	秘密鍵のパスワードを指定します.(デフォルト:空文字)
Insecure[=<True/False>]	セキュア通信時にサーバの証明書とサーバの整合性チェックの省略設定を指定します. True:チェックを省略する. False:チェックを省略しない(デフォルト)

2.2.1.1. Conn オプション

以下に Conn オプションのパラメータ文字列を示します.

"Conn=<host>[:<port>[:<keepalive>]]]"

<host>	:	接続先ホスト名又は IP アドレス. 例:"localhost", "192.168.0.1"
<port>	:	接続ポート番号. 1883, 8883, …任意指定可能.(デフォルト:1883)
<keepalive>	:	MQTT Broker に設定するキープアライブ値. (秒) (デフォルト:60)

2.2.1.2. MqttProtocol オプション

以下に MqttProtocol オプションのパラメータ文字列を示します.

"MqttProtocol[=<version>]"

<version>	:	通信に用いる MQTT プロトコルバージョン. V3.1 と V3.1.1 が選択可能. v31, v311.(デフォルト:v311)

2.2.1.3. Will オプション

以下に Will オプションのパラメータ文字列を示します.

Will[=<topic>:<payload>[:<qos>[:<retain>]]]

<topic>	:	トピック.
<payload>	:	Will 本文.
<qos>	:	QoS. 0, 1, 2.(デフォルト:0)
<retain>	:	Retain. True, False.(デフォルト:False)

2.2.2. CaoController::AddVariable メソッド

CaoController クラスの AddVariable メソッドは，それぞれのプロバイダで Publish を行うための変数オブジェクトを作成するためのメソッドです．

Publish を実行する際には変数名をトピックとして使用します．(参照：2.2.2.1)

書式　AddVariable(<bstrVariableName:VT_BSTR>[,<bstrOption:VT_BSTR>])

<bstrVariableName>　：　[in] 変数名

<bstrOption>　：　[in] オプション文字列

以下にオプション文字列に指定するリストを示します．

表 2-3 CaoController::AddVariable のオプション文字列

オプション [3]	説明
QoS[=<0/1/2>]	Publish する際の QoS． 0, 1, 2
Retain[=<True/False>]	Publish する際の Retain． True, False
Type[=<0/1>]	Publish するメッセージのデータ型． 0 (VT_BSTR) ，1 (VT_UI1 | VT_ARRAY)
CodePage[=<コードページ>]	Publish するメッセージのコードページ. 0～（参照：2.2.2.2）

[3] 角括弧(“[ ]”)内は省略可能を示します．また，各パラメータの解説中の下線部はオプションを指定しなかったときのデフォルト値になります．

2.2.2.1. 変数名によるトピック指定

PUBLISHER プロバイダは変数名をトピックとして使用します.

トピックは“/”(スラッシュ)を区切りとした階層形式で, “#”(シャープ)や“+”(プラス)を用いたワイルドカード指定を行うことができます. (組み合わせて使用することもできます)

/ : 区切りの階層構造

: その階層以下の全てのトピック

+ : その階層はワイルドカードとして扱う

例） Office/# … “Office”以下の階層のトピック全てが該当

マッチする : Office/Tokyo

マッチする : Office/Osaka

マッチする : Office/Nagoya/4F

マッチしない : House/Tokyo/1F

例） Office/+/3F … 1 階層目と 3 階層目が一致していれば 2 階層目は問わない

マッチする : Office/Tokyo/3F

マッチする : Office/Osaka/3F

マッチしない : Office/Nagoya/4F

マッチしない : Office/Tokyo/1F

2.2.2.2. コードページ指定

コードページまたは定数を指定します.

2.2.2.2.1. コードページ

例)

932	:	日本語(シフト JIS)
1200	:	Unicode
65001	:	Unicode(UTF-8)

2.2.2.2.2. 定数

0	:	システムデフォルトの Windows ANSI コードページ
1	:	現在のシステムの OEM コードページ
2	:	現在のシステムの Macintosh コードページ
3	:	現在のスレッドの Windows ANSI コードページ
42	:	シンボルコードページ

2.2.3. CaoVariable:put_Value プロパティ

変数名をトピック, Value として指定された文字列(VT_BSTR), または BYTE 配列(VT_UI1 | VT_ARRAY) を Payload として Publish を実行します.

2.2.4. CaoVariable:get_Value プロパティ

最後に Publish を実行した値(VT_BSTR, または(VT_UI1 | VT_ARRAY))を返します.

2.2.5. CaoController::OnMessage イベント

以下の契機で CaoController クラスの OnMessage イベントが発生します.

表 2-4 メッセージ種別

メッセージ種別		発生契機
1	断線検出	何らかの要因により MQTT Broker から切断された際に発生します.

2.2.5.1. 断線検出

断線検出メッセージで得られるデータ形式を以下に示します.

Number : メッセージ種別 (1)

Value : 切断要因のレスポンスコード (VT_I4)

DateTime : タイムスタンプ

Destination : Null

Source : 接続先ホスト名又は IP アドレス

Description : レスポンスコードの説明文字列

2.3. 変数一覧

2.3.1. CaoController クラス

表 2-5 CaoController クラス ユーザ変数一覧

変数名	データ型	説明	属性	
			get	put
任意 (トピックとして使用)	VT_BSTR	put_Value で Publish を実行.	○	○

表 2-6 CaoController クラス システム変数一覧

変数名	データ型	説明	属性	
			get	put
@VERSION	VT_BSTR	プロバイダバージョン情報.	○	－
@MQTT_LIB_VERSION	VT_BSTR	Mosquitto ライブラリ バージョン情報	○	－

2.4. エラーコード

PUBLISHER プロバイダでは，以下の固有エラーコードが定義されています．

ORiN2 共通エラーについては，「ORiN2 プログラミングガイド」のエラーコードの章を参照してください．

表 2-7 固有エラーコード

エラー名	エラー番号	説明
E_MOSQ_CONN_PENDING	0x80100001	Mosquitto ライブラリのエラーコード MOSQ_ERR_CONN_PENDING が発生
E_MOSQ_NOMEM	0x80100002	Mosquitto ライブラリのエラーコード MOSQ_ERR_NOMEM が発生
E_MOSQ_PROTOCOL	0x80100003	Mosquitto ライブラリのエラーコード MOSQ_ERR_PROTOCOL が発生
E_MOSQ_INVAL	0x80100004	Mosquitto ライブラリのエラーコード MOSQ_ERR_INVAL が発生
E_MOSQ_NO_CONN	0x80100005	Mosquitto ライブラリのエラーコード MOSQ_ERR_NO_CONN が発生
E_MOSQ_CONN_REFUSED	0x80100006	Mosquitto ライブラリのエラーコード MOSQ_ERR_CONN_REFUSED が発生
E_MOSQ_NOT_FOUND	0x80100007	Mosquitto ライブラリのエラーコード MOSQ_ERR_NOT_FOUND が発生
E_MOSQ_CONN_LOST	0x80100008	Mosquitto ライブラリのエラーコード MOSQ_ERR_CONN_LOST が発生
E_MOSQ_TLS	0x80100009	Mosquitto ライブラリのエラーコード MOSQ_ERR_TLS が発生
E_MOSQ_PAYLOAD_SIZE	0x8010000A	Mosquitto ライブラリのエラーコード MOSQ_ERR_PAYLOAD_SIZE が発生
E_MOSQ_NOT_SUPPORTED	0x8010000B	Mosquitto ライブラリのエラーコード MOSQ_ERR_NOT_SUPPORTED が発生
E_MOSQ_AUTH	0x8010000C	Mosquitto ライブラリのエラーコード MOSQ_ERR_AUTH が発生
E_MOSQ_ACL_DENIED	0x8010000D	Mosquitto ライブラリのエラーコード MOSQ_ERR_ACL_DENIED が発生
E_MOSQ_UNKNOWN	0x8010000E	Mosquitto ライブラリのエラーコード MOSQ_ERR_UNKNOWN が発生

E_MOSQ_ERRNO	0x8010000F	Mosquitto ライブラリのエラーコード MOSQ_ERR_ERRNO が発生 接続先の MQTT Broker が起動していない可能性があります. MQTT Broker を起動してから再接続してください.
E_MOSQ_EAI	0x80100010	Mosquitto ライブラリのエラーコード MOSQ_ERR_EAI が発生
E_MOSQ_PROXY	0x80100011	Mosquitto ライブラリのエラーコード MOSQ_ERR_PROXY が発生

付録A. MQTT 用語集

以下に MQTT プロトコルを扱う上で登場する用語を記載します.

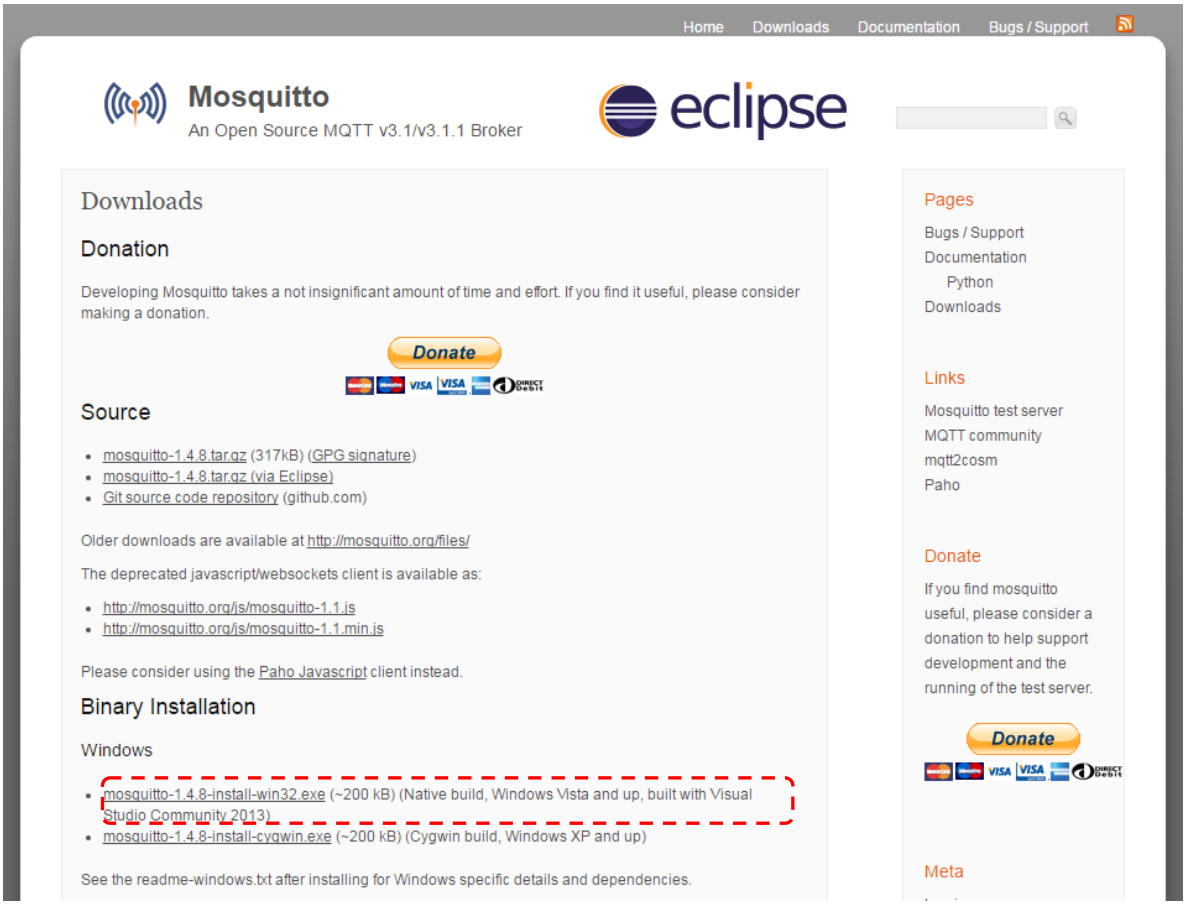
- **Broker(ブローカ)**
  MQTT プロトコル通信におけるサーバ.
  Publisher から送信されたトピックのメッセージを該当トピックの購読をしている Subscriber に配信する.
- **Subscriber(サブスクライバ)**
  MQTT プロトコル通信におけるクライアント(メッセージ受信側).
  購読したいトピックを Broker に登録(Subscribe)し, メッセージを受信する.
- **Publisher(パブリッシャ)**
  MQTT プロトコル通信におけるクライアント(メッセージ送信側).
  Subscriber を意識することなく, トピックを指定して Broker へメッセージを送信(Publish)する.
- **QoS**
  メッセージ毎の到達保障.
  QoS0 ： At most once (最高 1 回配信)
  QoS1 ： At least once (最低 1 回配信)
  QoS2 ： Exactly once (正確に 1 回配信)
- **Will**
  Will は「遺言」を意味し, Client が Broker へ接続する際に設定できる.
  Broker と Client の接続が切断された際に Will で指定されたトピックとメッセージを Broker に接続している Subscriber に送信する.
- **Retain**
  トピック毎に最後に Publish されたメッセージを Broker が保持しておき, 新たに接続してきた Subscriber へそのメッセージを送信する機能.

付録B. MQTT Broker 導入例

MQTT Broker の導入方法をオープンソースの Mosquitto を例に紹介します.

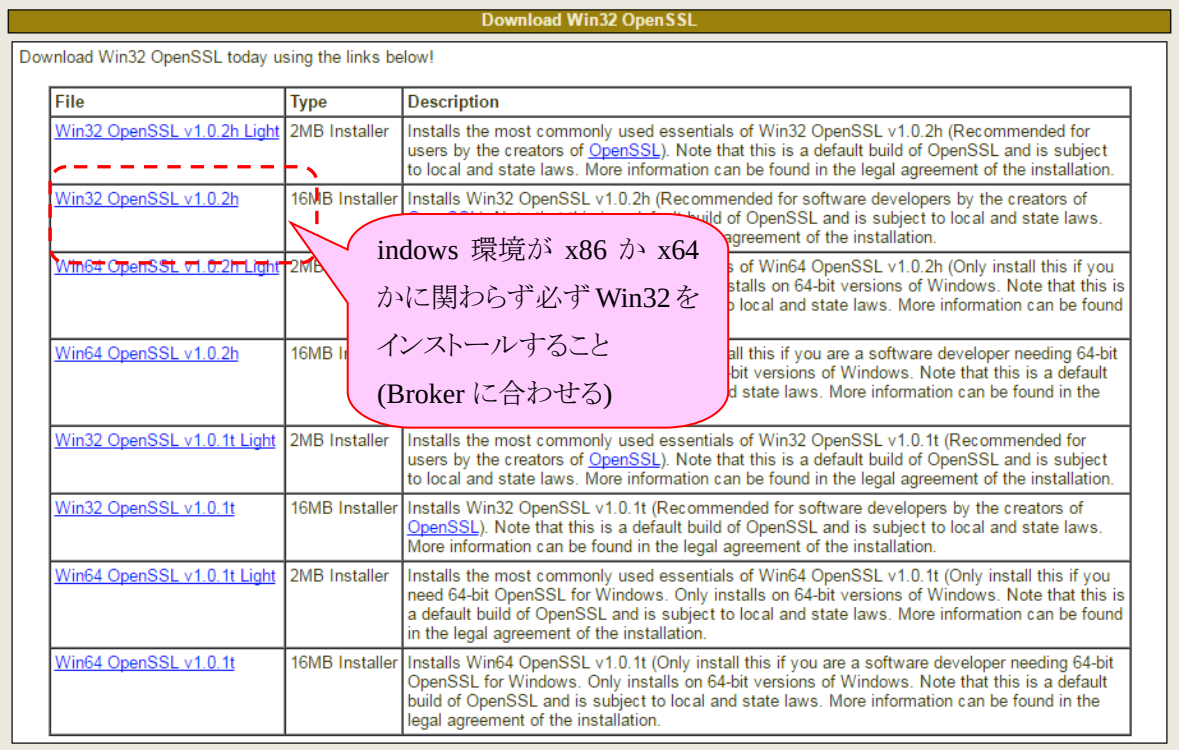
【Windows 編】

1. http://mosquitto.org/download/ からインストーラをダウンロード.



2. ダウンロードしたインストーラを実行しインストール開始
3. OpenSSL(libeay32.dll, ssleay32.dll)と pthreads(pthreadVC2.dll)を手動でインストールする必要がある旨が表示されるが, 現段階では無視して進める. (詳細は後述)
4. コンポーネントはデフォルトのままで進める.
5. インストール先はデフォルトの[C:¥Program Files (x86)¥mosquitto]から[C:¥mosquitto]等へ変更する. (パスにスペースを含んでいると設定等を変更する際に分かりにくいため)
6. インストールを完了させる.

7. http://slproweb.com/products/Win32OpenSSL.html からコンパイル済の OpenSSL の Win32 インストーラ(最新の版を推奨)をダウンロードしインストールする. (全てデフォルトで OK)



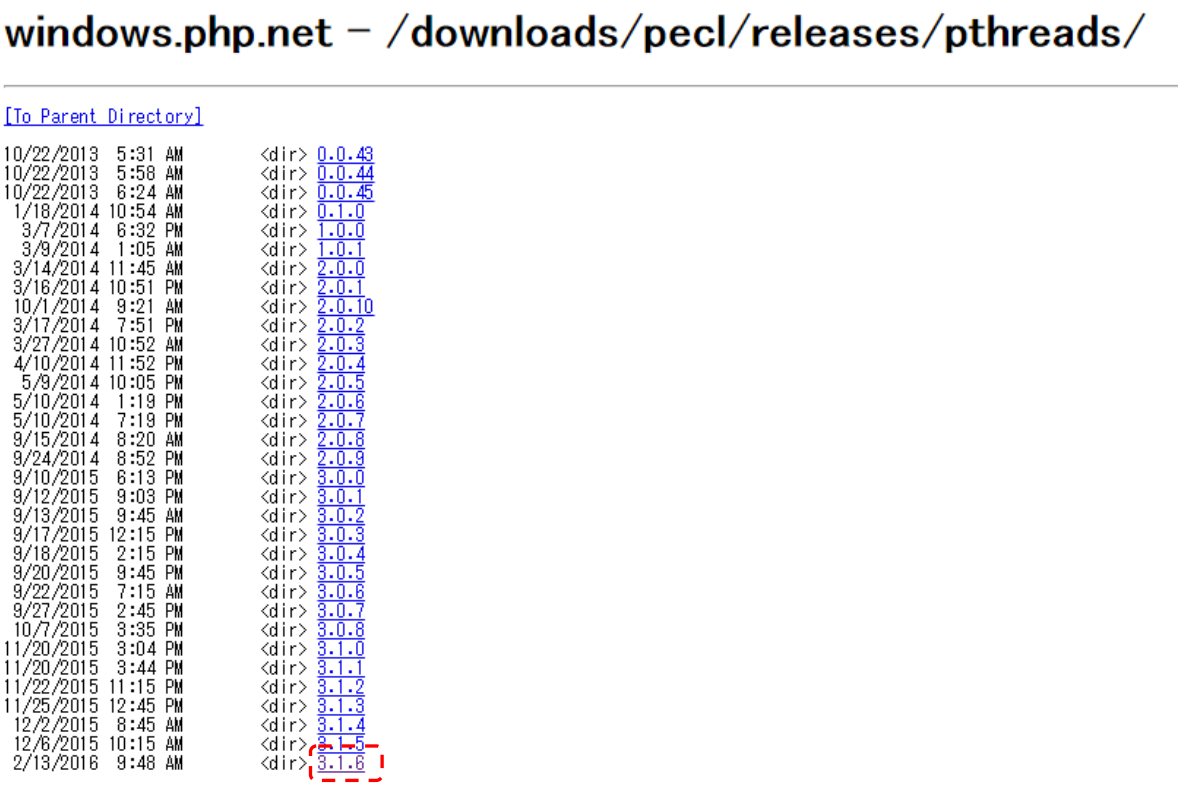
Download Win32 OpenSSL

Download Win32 OpenSSL today using the links below!

File	Type	Description
Win32 OpenSSL v1.0.2h Light	2MB Installer	Installs the most commonly used essentials of Win32 OpenSSL v1.0.2h (Recommended for users by the creators of OpenSSL). Note that this is a default build of OpenSSL and is subject to local and state laws. More information can be found in the legal agreement of the installation.
Win32 OpenSSL v1.0.2h	16MB Installer	Installs Win32 OpenSSL v1.0.2h (Recommended for software developers by the creators of OpenSSL). Note that this is a default build of OpenSSL and is subject to local and state laws. More information can be found in the legal agreement of the installation.
Win64 OpenSSL v1.0.2h Light	2MB	s of Win64 OpenSSL v1.0.2h (Only install this if you stalls on 64-bit versions of Windows. Note that this is o local and state laws. More information can be found
Win64 OpenSSL v1.0.2h	16MB I	all this if you are a software developer needing 64-bit -bit versions of Windows. Note that this is a default d state laws. More information can be found in the
Win32 OpenSSL v1.0.1t Light	2MB Installer	Installs the most commonly used essentials of Win32 OpenSSL v1.0.1t (Recommended for users by the creators of OpenSSL). Note that this is a default build of OpenSSL and is subject to local and state laws. More information can be found in the legal agreement of the installation.
Win32 OpenSSL v1.0.1t	16MB Installer	Installs Win32 OpenSSL v1.0.1t (Recommended for software developers by the creators of OpenSSL). Note that this is a default build of OpenSSL and is subject to local and state laws. More information can be found in the legal agreement of the installation.
Win64 OpenSSL v1.0.1t Light	2MB Installer	Installs the most commonly used essentials of Win64 OpenSSL v1.0.1t (Only install this if you need 64-bit OpenSSL for Windows. Only installs on 64-bit versions of Windows. Note that this is a default build of OpenSSL and is subject to local and state laws. More information can be found in the legal agreement of the installation.
Win64 OpenSSL v1.0.1t	16MB Installer	Installs Win64 OpenSSL v1.0.1t (Only install this if you are a software developer needing 64-bit OpenSSL for Windows. Only installs on 64-bit versions of Windows. Note that this is a default build of OpenSSL and is subject to local and state laws. More information can be found in the legal agreement of the installation.

indows 環境が x86 か x64 かに関わらず必ず Win32 をインストールすること (Broker に合わせる)

8. http://windows.php.net/downloads/pecl/releases/pthreads/ から pthreads の圧縮ファイル(最新の版を推奨)をダウンロードし，任意の場所へ展開する.



9. OpenSSL と pthreads の場所にパスを通す.
10. PC を再起動する.

上記の手順で Mosquitto Broker を使用することができるようになります. Mosquitto Broker はデフォルトで PC 起動時にサービスとして起動しています. コマンドラインからオプションを指定して起動したり，設定を変更させた conf ファイルを読み込ませたりしたい場合は起動中のサービスを終了させる必要があります.

【Linux(Ubuntu)編】

1. Terminal を起動し，以下のコマンドを実行します．
2. パッケージリストの更新．

```
$ sudo apt-get update
```

3. Broker の導入．

```
$ sudo apt-get install mosquitto
```

4. Client の導入．

```
$ sudo apt-get install mosquitto-clients
```

OpenSSL 等依存のあるパッケージがインストールされていない場合は別途入れる必要があります．

パッケージリストに mosquitto が出てこない場合は以下のコマンドでリポジトリを追加してください．

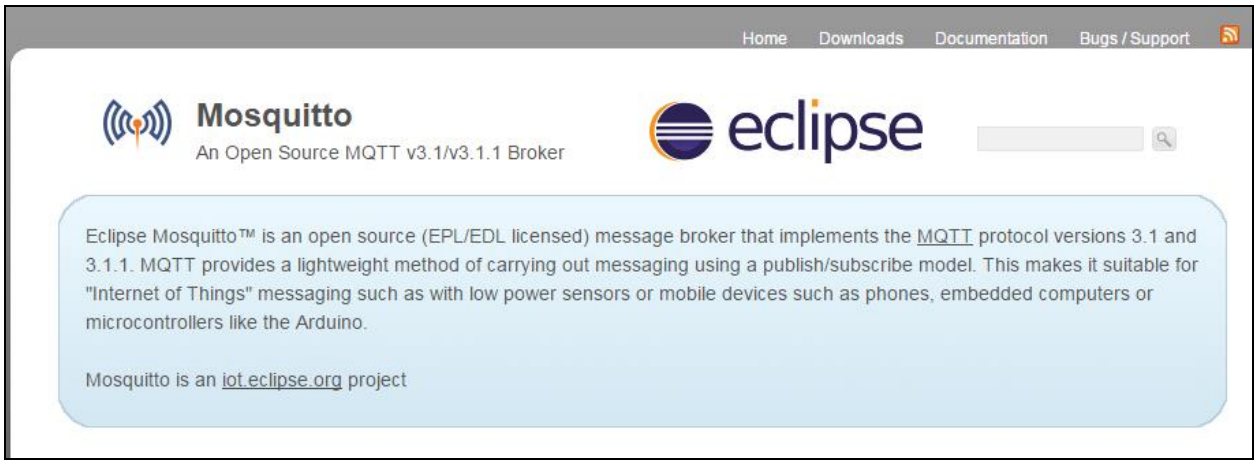
```
$ sudo add-apt-repository ppa:mosquitto-dev/mosquitto-ppa
```

Mosquitto のパッケージ導入後，Broker はサービスとして起動している状態になります．

Windows 環境と同様にオプションを指定して起動したり，設定を変更させた conf ファイルを読み込ませたりしたい場合は起動中のサービスを終了させる必要があります．

付録C. Mosquitto

URL: https://mosquitto.org/



EPL

Eclipse Public License - v 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and

b) in the case of each subsequent Contributor:

i) changes to the Program, and

ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.
b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.
c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.
d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and
b) its license agreement:

i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;
ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;
iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and
iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and
b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes

performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement , including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

EDL

Eclipse Distribution License - v 1.0

Copyright (c) 2007, Eclipse Foundation, Inc. and its licensors.

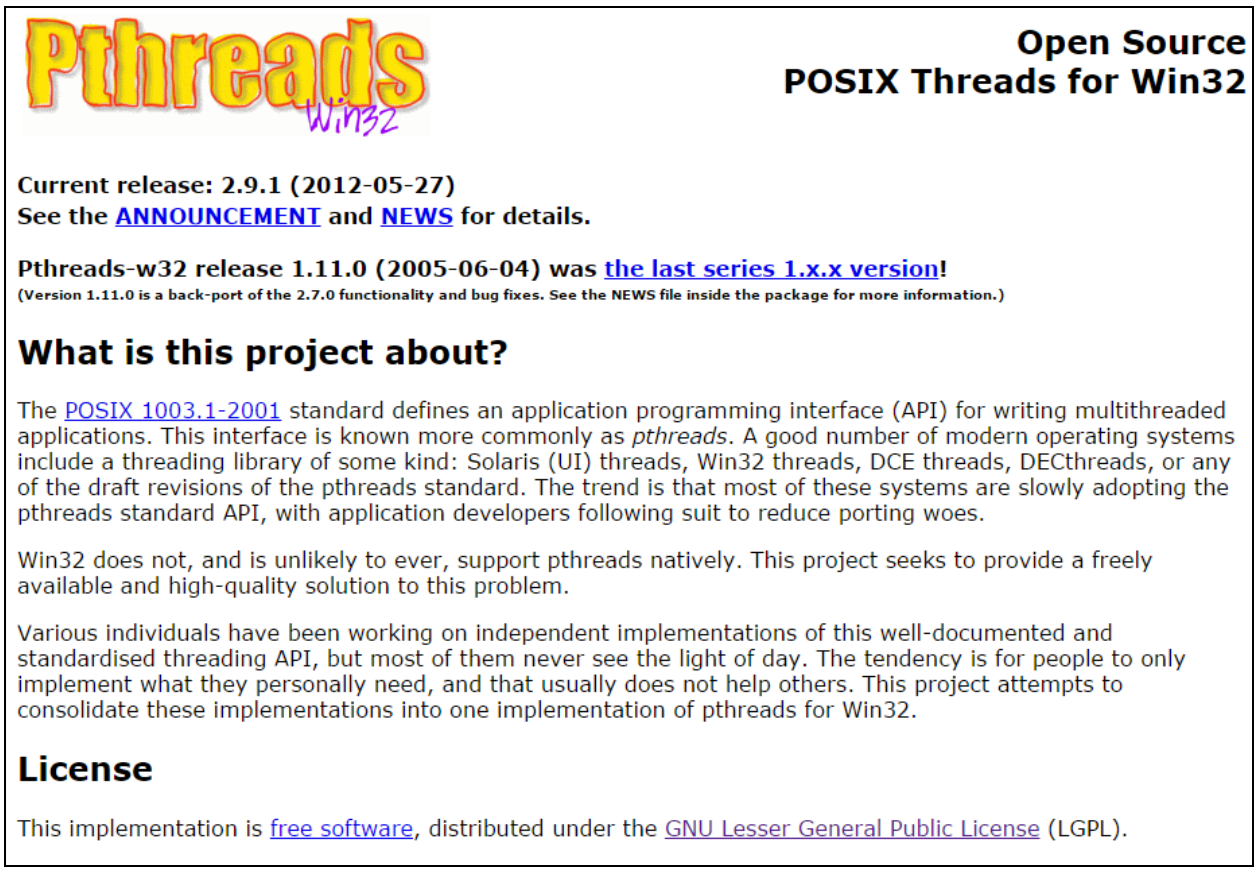
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

付録D. Pthread_win32

URL: http://sources.redhat.com/pthreads-win32



Pthreads Win32

Open Source POSIX Threads for Win32

Current release: 2.9.1 (2012-05-27)
See the ANNOUNCEMENT and NEWS for details.

Pthreads-w32 release 1.11.0 (2005-06-04) was the last series 1.x.x version!
(Version 1.11.0 is a back-port of the 2.7.0 functionality and bug fixes. See the NEWS file inside the package for more information.)

What is this project about?

The POSIX 1003.1-2001 standard defines an application programming interface (API) for writing multithreaded applications. This interface is known more commonly as *pthreads*. A good number of modern operating systems include a threading library of some kind: Solaris (UI) threads, Win32 threads, DCE threads, DECthreads, or any of the draft revisions of the pthreads standard. The trend is that most of these systems are slowly adopting the pthreads standard API, with application developers following suit to reduce porting woes.

Win32 does not, and is unlikely to ever, support pthreads natively. This project seeks to provide a freely available and high-quality solution to this problem.

Various individuals have been working on independent implementations of this well-documented and standardised threading API, but most of them never see the light of day. The tendency is for people to only implement what they personally need, and that usually does not help others. This project attempts to consolidate these implementations into one implementation of pthreads for Win32.

License

This implementation is free software, distributed under the GNU Lesser General Public License (LGPL).

LGPL

```
        pthreads-win32 - a POSIX threads library for Microsoft Windows


This file is Copyrighted
------------------------

    This file is covered under the following Copyright:

        Copyright (C) 2001,2006 Ross P. Johnson
        All rights reserved.
```

Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Pthreads-win32 is covered by the GNU Lesser General Public License
---

Pthreads-win32 is open software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation version 2.1 of the License.

Pthreads-win32 is several binary link libraries, several modules, associated interface definition files and scripts used to control its compilation and installation.

Pthreads-win32 is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

A copy of the GNU Lesser General Public License is distributed with pthreads-win32 under the filename:

COPYING.LIB

You should have received a copy of the version 2.1 GNU Lesser General Public License with pthreads-win32; if not, write to:

Free Software Foundation, Inc.
59 Temple Place
Suite 330
Boston, MA 02111-1307
USA

The contact addresses for pthreads-win32 is as follows:

Web: http://sources.redhat.com/pthreads-win32

```
      Email:  Ross Johnson
              Please use: Firstname.Lastname@homemail.com.au



Pthreads-win32 copyrights and exception files
---------------------------------------------

   With the exception of the files listed below, Pthreads-win32
   is covered under the following GNU Lesser General Public License
   Copyrights:

       Pthreads-win32 - POSIX Threads Library for Win32
       Copyright(C) 1998 John E. Bossom
       Copyright(C) 1999,2006 Pthreads-win32 contributors

       The current list of contributors is contained
       in the file CONTRIBUTORS included with the source
       code distribution. The current list of CONTRIBUTORS
       can also be seen at the following WWW location:
       http://sources.redhat.com/pthreads-win32/contributors.html

   Contact Email: Ross Johnson
                  Please use: Firstname.Lastname@homemail.com.au

   These files are not covered under one of the Copyrights listed above:

          COPYING
          COPYING.LIB
          tests/rwlock7.c

   This file, COPYING, is distributed under the Copyright found at the
   top of this file.  It is important to note that you may distribute
   verbatim copies of this file but you may not modify this file.

   The file COPYING.LIB, which contains a copy of the version 2.1
   GNU Lesser General Public License, is itself copyrighted by the
```

```
   Free Software Foundation, Inc.  Please note that the Free Software
   Foundation, Inc. does NOT have a copyright over Pthreads-win32,
   only the COPYING.LIB that is supplied with pthreads-win32.

   The file tests/rwlock7.c is derived from code written by
   Dave Butenhof for his book 'Programming With POSIX(R) Threads'.
   The original code was obtained by free download from his website
   http://home.earthlink.net/~anneart/family/Threads/source.html
   and did not contain a copyright or author notice. It is assumed to
   be freely distributable.

   In all cases one may use and distribute these exception files freely.
   And because one may freely distribute the LGPL covered files, the
   entire pthreads-win32 source may be freely used and distributed.



General Copyleft and License info
---------------------------------

   For general information on Copylefts, see:

       http://www.gnu.org/copyleft/

   For information on GNU Lesser General Public Licenses, see:

       http://www.gnu.org/copyleft/lesser.html
       http://www.gnu.org/copyleft/lesser.txt


Why pthreads-win32 did not use the GNU General Public License
-------------------------------------------------------------

   The goal of the pthreads-win32 project has been to
   provide a quality and complete implementation of the POSIX
   threads API for Microsoft Windows within the limits imposed
   by virtue of it being a stand-alone library and not
```

linked directly to other POSIX compliant libraries. For example, some functions and features, such as those based on POSIX signals, are missing.

Pthreads-win32 is a library, available in several different versions depending on supported compilers, and may be used as a dynamically linked module or a statically linked set of binary modules. It is not an application on it's own.

It was fully intended that pthreads-win32 be usable with commercial software not covered by either the GPL or the LGPL licenses. Pthreads-win32 has many contributors to it's code base, many of whom have done so because they have used the library in commercial or proprietry software projects.

Releasing pthreads-win32 under the LGPL ensures that the library can be used widely, while at the same time ensures that bug fixes and improvements to the pthreads-win32 code itself is returned to benefit all current and future users of the library.

Although pthreads-win32 makes it possible for applications that use POSIX threads to be ported to Win32 platforms, the broader goal of the project is to encourage the use of open standards, and in particular, to make it just a little easier for developers writing Win32 applications to consider widening the potential market for their products.