wolfSSL Inc.
www.wolfssl.jp, info@wolfssl.jp

wolfTPM (TPM 2.0)

ユーザ・ガイド

日本語版ドラフト v1.0

2018 年 8 月 13 日

wolfSSL Inc.
www.wolfssl.jp, info@wolfssl.jp

改版履歴

wolfTPM Release 1.3 (07/20/2018)

- TIS TPM_BASE_ADDRESS がチップの仕様に適合していることを確認するよう修正
- 静的コード分析の警告に対する修正
- コンパイラの違いで出るマイナーなビルド警告に対する修正
- ソフトウェアベースの RSA を使用して、RSA 指数が 7 未満の TPM 障害を修正
- TPM ベンチマークのサポートを追加
- 公開鍵を wolfSSL 形式でインポート/エクスポートする機能を追加
- TPM による署名/検証を示す PKCS7 のサンプルコードを追加
- TPM キーに基づいて証明書要求を生成する CSR の例を追加
- CSR 署名スクリプト./certs/certreq.sh を追加して、自己署名 CA を使用して証明書を作成
- クライアント証明書に TPM ベースの鍵を使用する TLS クライアントのサンプルコードを追加
- wolfSSL WOLF_CRYPT_DEV コールバックのサポートを追加し、TPM ベースの ECC および RSA 秘密鍵を有効化
- ./examples/wrap/wrap_test 1 を使用して TPM をクリア/リセットする機能を追加
- サンプルコードの設定のいくつかを./examples/tpm_io.h に移動
-

wolfTPM Release 1.1 (03/09/2018)

- ラッパーは鍵生成、RSA 暗号/復号、ECC 署名、検証、ECDH および NV を追加
- TPM2 ラッパーのプログラム例を追加
- Raspberry Pi で動作する Linux SPI サポートを追加
- TPM2 コマンドと応答のアセンブリーと解析を修正
- コマンドと応答の認証サポートを修正
- 暗号化、復号化進捗状況のサポートパラメータ
- TIS と Packet 層を整理、新しいファイルに
- `wolfTPM2_GetRCString` のエラーコード、文字列の修正と改善

- 新規に `TPM2_Cleanup` 関数の追加
- TPM2 ネイティブ API の強化(テストカバレッジ約 75%)

wolfTPM Release 1.0 (02/06/2018)

- すべての TPM2 ネイティブ API を TIS と SPI IO コールバックでサポート
- TPM の返却文字列を得る ヘルパー関数 `TPM2_GetRCString` の追加
- STM32 CubeMX SPI で動作する TPM 2.0 のデモプログラム
  （ `examples/tpm/tpm2_demo.c` ）

wolfSSL Inc.
www.wolfssl.jp, info@wolfssl.jp

wolfTPM (TPM 2.0)

wolfTPM は組込み利用のための TPM2.0 ポータビリティを実現する製品です。この資料は、wolfTPM の TPM2.0 サポートに関して説明します。

特徴

- 本製品は、TPM2.0 の仕様に準拠した API を提供します。
- ラッパーは鍵生成、RSA 暗号/復号、ECC 署名、検証、ECDH および NV を提供します。
- テストは Infineon OPTIGA SLB9670 および LetsTrust TPM にて行いました。
- SPI 上の TPM Interface Specification (TIS)を使用します。
- プラットフォームとしては、Raspberry Pi と STM32 (CubeMX) を使用します。
- 他のプラットフォームへのポーティングが容易にできるよう設計されています。
    - すべて C コードをベースに組み込み向けに設計
    - ハードウェア SPI インターフェースのための一つの I/O コールバック
    - 外部の他のコンポーネントへ依存しない
    - コンパクトなコード設計と最小のメモリー所要量
- 以下の使用例を提供しています。
    - TPM2 ネイティブな API
    - TPM2 ラッパー
    - PKCS 7 機能
    - 証明書署名要求 (CSR)
    - TLS クライアント

wolfSSL Inc.
www.wolfssl.jp, info@wolfssl.jp

wolfTPM の概要

階層構造

```
プラットフォーム     TPM_RH_PLATFORM
所有者         TPM_RH_OWNER
裏書き(Endorsement) TPM_RH_ENDORSEMENT
```

それぞれの階層にて自身の製造工程で生成されるシードを持つ。

TPM2_CreateまたはTPM2_CreatePrimaryで使用される引数は、使用される鍵ベースの階層を生成するために KDF に渡されるテンプレートを作成します。生成される鍵は毎回同じです。再起動しても新しい RSA 2048 ビットの鍵の生成には約 15 秒かかります。通常、これらは TPM2_EvictControl で作成され NV に格納されます。各 TPM は、シードに基づいて一意に独自の鍵を生成します。 一時的な鍵を作成するために使用できる一時的な階層（TPM_RH_NULL）もあります。

プラットフォーム構成レジスタ(PCRs)

インデックス 0-23 の SHA-1、 SHA-256 によるハッシュダイジェストを保持します。これらのハッシュダイジェストは、一つのブートシーケンス(secure boot) の一貫性を提供するために拡張することができます。

用語

本製品では、”marshall”に対して“追加(append)”、”unmarshall” に対して“解析(parse)” を使用します。

wolfSSL Inc.
www.wolfssl.jp, info@wolfssl.jp

サポート・プラットフォーム

本製品に含まれている使用例は、Raspberry Pi® 3 および STM32 (CubeMX HAL 使用)でテストされています。Raspberry 3 ではネイティブの spi_dev インタフェースと/dev/spidev0.1 のデフォルトが使用されています。Infineon パッチを使用すると、spi_tis_dev でカーネルの SPI インタフェースを上書きし、現時点ではデモが失敗する原因となります。これらは Rasbian 4.4.x.でのみ動作確認しています。

SPI IO コールバック

ハードウェアプラットフォームとのインタフェースの使用方法に関しては tpm2_demo.c のコールバック関数 TPM2_IoCb.を参照してください。TPM デモのために、自分用の IO コールバック関数に修正することができます。

テスト・ハードウェア

以下の環境でテストされています:

- Infineon OPTIGA (TM) Trusted Platform Module 2.0 SLB 9670
- LetsTrust: http://letstrust.de (https://buyzero.de/collections/andere-platinen/products/letstrust-hardware-tpm-trusted-platform-module). Compact Raspberry Pi TPM 2.0 board（Infineon SLB 9670 ベース）.

wolfSSL Inc.
www.wolfssl.jp, info@wolfssl.jp

ビルド方法

ビルドは、wolfSSL のビルド、wolfTMP のビルドの順に行います。

1） wolfSSL のビルド:

```
以下のコマンドでビルドします。
./autogen.sh
./configure --enable-ecc --enable-sha512
make
make check
sudo make install
sudo ldconfig
```

2）wolfTPM のビルド:

```
以下のコマンドでビルドします。
./autogen.sh
./configure
make
./examples/wrap/wrap_test
./examples/native/native_test
./examples/bench/bench
./examples/csr/csr
./examples/pkcs7/pkcs7
./examples/tls/tls_client
```

プログラム例

wolfTPM には以下のプログラム例が含まれています。これらのプロブラムはビルド時の make コマンドで一括してビルドされます。

TPM2 ラッパーのテスト

```
./examples/wrap/wrap_test
TPM2 Demo for Wrapper API's
RSA Encrypt Test Passed
ECC Sign/Verify Passed
ECC DH Generation Passed
NV Test on index 0x1800200 with 1024 bytes passed
```

TPM2 ベンチマーク

Infineon OPTIGA SLB9670 で動作の場合:

```
./examples/bench/bench
TPM2 Benchmark using Wrapper API's
RSA     2048 Public        65 ops took 1.005 sec, avg 15.466 ms, 64.657 ops/sec
RSA     2048 Private        3 ops took 1.343 sec, avg 447.759 ms, 2.233 ops/sec
RSA     2048 Pub  OAEP     12 ops took 1.040 sec, avg 86.657 ms, 11.540 ops/sec
RSA     2048 Priv OAEP      2 ops took 1.032 sec, avg 515.885 ms, 1.938 ops/sec
ECDSA    256 sign          14 ops took 1.037 sec, avg 74.101 ms, 13.495 ops/sec
ECDSA    256 verify         8 ops took 1.027 sec, avg 128.417 ms, 7.787 ops/sec
ECDHE    256 agree          8 ops took 1.040 sec, avg 130.003 ms, 7.692 ops/sec
```

TPM ネイティブのテスト

```
./examples/native/native_test
TPM2 Demo using Native API's
TPM2: Caps 0x30000697, Did 0x001b, Vid 0x15d1, Rid 0x10
TPM2_Startup pass
TPM2_SelfTest pass
TPM2_GetTestResult: Size 10, Rc 0x0
TPM2_IncrementalSelfTest: Rc 0x0, Alg 0x1 (Todo 0)
TPM2_GetCapability: Property FamilyIndicator 0x322e3000
TPM2_GetCapability: Property PCR Count 24
TPM2_GetRandom: Got 32 bytes
TPM2_StirRandom: success
TPM2_PCR_Read: Index 0, Digest Sz 32, Update Counter 21
TPM2_PCR_Read: Index 1, Digest Sz 32, Update Counter 21
TPM2_PCR_Read: Index 2, Digest Sz 32, Update Counter 21
TPM2_PCR_Read: Index 3, Digest Sz 32, Update Counter 21
TPM2_PCR_Read: Index 4, Digest Sz 32, Update Counter 21
TPM2_PCR_Read: Index 5, Digest Sz 32, Update Counter 21
TPM2_PCR_Read: Index 6, Digest Sz 32, Update Counter 21
TPM2_PCR_Read: Index 7, Digest Sz 32, Update Counter 21
TPM2_PCR_Read: Index 8, Digest Sz 32, Update Counter 21
TPM2_PCR_Read: Index 9, Digest Sz 32, Update Counter 21
TPM2_PCR_Read: Index 10, Digest Sz 32, Update Counter 21
TPM2_PCR_Read: Index 11, Digest Sz 32, Update Counter 21
TPM2_PCR_Read: Index 12, Digest Sz 32, Update Counter 21
TPM2_PCR_Read: Index 13, Digest Sz 32, Update Counter 21
TPM2_PCR_Read: Index 14, Digest Sz 32, Update Counter 21
TPM2_PCR_Read: Index 15, Digest Sz 32, Update Counter 21
TPM2_PCR_Read: Index 16, Digest Sz 32, Update Counter 21
TPM2_PCR_Read: Index 17, Digest Sz 32, Update Counter 21
TPM2_PCR_Read: Index 18, Digest Sz 32, Update Counter 21
```

```
TPM2_PCR_Read: Index 19, Digest Sz 32, Update Counter 21
TPM2_PCR_Read: Index 20, Digest Sz 32, Update Counter 21
TPM2_PCR_Read: Index 21, Digest Sz 32, Update Counter 21
TPM2_PCR_Read: Index 22, Digest Sz 32, Update Counter 21
TPM2_PCR_Read: Index 23, Digest Sz 32, Update Counter 21
TPM2_PCR_Extend success
TPM2_PCR_Read: Index 0, Digest Sz 32, Update Counter 22
TPM2_StartAuthSession: sessionHandle 0x3000000
TPM2_PolicyGetDigest: size 32
TPM2_PCR_Read: Index 0, Digest Sz 20, Update Counter 22
wc_Hash of PCR[0]: size 32
TPM2_PolicyPCR failed 0x1c4: TPM_RC_AUTHSIZE
TPM2_PolicyPCR: Updated
TPM2_PolicyRestart: Done
TPM2_HashSequenceStart: sequenceHandle 0x80000000
Hash SHA256 test success
TPM2_CreatePrimary: Endorsement 0x80000000 (314 bytes)
TPM2_CreatePrimary: Storage 0x80000001 (282 bytes)
TPM2_LoadExternal: 0x80000002
TPM2_MakeCredential: credentialBlob 68, secret 256
TPM2_ReadPublic Handle 0x80000002: pub 314, name 34, qualifiedName 34
Create HMAC-SHA256 Key success, public 48, Private 141
TPM2_Load New HMAC Key Handle 0x80000002
TPM2_PolicyCommandCode: success
TPM2_ObjectChangeAuth failed 0x9a2: TPM_RC_BAD_AUTH
TPM2_ObjectChangeAuth: private 2
TPM2_ECC_Parameters: CurveID 3, sz 256, p 32, a 32, b 32, gX 32, gY 32, n 32, h 1
TPM2_Create: New ECDSA Key: pub 88, priv 126
TPM2_Load ECDSA Key Handle 0x80000002
TPM2_Sign: ECC S 32, R 32
TPM2_VerifySignature: Tag 32802
TPM2_Create: New ECDH Key: pub 88, priv 126
```

```
TPM2_Load ECDH Key Handle 0x80000002

TPM2_ECDH_KeyGen: zPt 68, pubPt 68

TPM2_Create: New RSA Key: pub 278, priv 222

TPM2_Load RSA Key Handle 0x80000002

TPM2_RSA_Encrypt: 256

TPM2_RSA_Decrypt: 32

RSA Encrypt/Decrypt test passed

TPM2_NV_ReadPublic failed 0x18b: TPM_RC_HANDLE

TPM2_NV_ReadPublic failed 0x18b: TPM_RC_HANDLE

TPM2_NV_ReadPublic failed 0x18b: TPM_RC_HANDLE

TPM2_NV_ReadPublic failed 0x18b: TPM_RC_HANDLE

TPM2_NV_ReadPublic failed 0x18b: TPM_RC_HANDLE

TPM2_NV_ReadPublic failed 0x18b: TPM_RC_HANDLE

TPM2_NV_ReadPublic failed 0x18b: TPM_RC_HANDLE

TPM2_NV_ReadPublic failed 0x18b: TPM_RC_HANDLE

TPM2_NV_ReadPublic failed 0x18b: TPM_RC_HANDLE

TPM2_NV_ReadPublic failed 0x18b: TPM_RC_HANDLE

TPM2_NV_DefineSpace failed 0x184: TPM_RC_VALUE
```

wolfSSL Inc.
www.wolfssl.jp, info@wolfssl.jp

TPM2 ラッパーのテスト（デバッグメッセージ有効化）

wolfTPM ビルド時の configure コマンドで次のようなデバッグメッセージ有効化指定をします。

```
./configure --enable-debug
```

またはヘッダーファイル内に `#define DEBUG_WOLFTPM` を指定します。

```
./examples/wrap/wrap_test
TPM2 Demo for Wrapper API's
TPM2: Caps 0x30000697, Did 0x001b, Vid 0x15d1, Rid 0x10
Command: 12
        80 01 00 00 00 0c 00 00 01 44 00 00            | .........D..
Response: 10
        80 01 00 00 00 0a 00 00 01 00                  | ..........
TPM2_Startup pass
Command: 14
        80 01 00 00 00 0e 00 00 01 73 81 00 02 00      | .........s....
Response: 366
        80 01 00 00 01 6e 00 00 00 00 01 1a 00 01 00 0b | .....n..........
        00 03 04 72 00 00 00 06 00 80 00 43 00 10 08 00 | ...r.......C....
        00 00 00 00 01 00 af 17 14 f3 dd 71 f5 4b ce 09 | ...........q.K..
        04 40 18 30 25 18 97 4e 7d 97 e2 6a 99 7f 1c 79 | .@.0%..N}..j...y
        d8 f1 bc eb 97 f1 6a 63 6d 43 60 a1 30 5a fc 14 | ......jcmC`.0Z..
        b5 e3 d1 e0 b7 39 90 43 30 11 8c e7 01 09 e8 01 | .....9.C0.......
        a8 bd e9 60 08 6d 8c a3 c1 a0 a5 40 e6 56 dc 98 | ...`.m.....@.V..
        84 75 4a ca 69 17 d9 0e f7 66 70 ce 64 51 20 b8 | .uJ.i....fp.dQ .
        70 c0 d5 86 ad b4 81 ab bd f2 43 73 3f 8c 2a 39 | p.........Cs?.*9
        f1 3a 21 18 82 c3 1d d3 39 d6 73 84 51 90 d3 f0 | .:!.....9.s.Q...
        7a 08 cc a7 f4 1c 6a 6f 27 48 43 bb ed af 3c a7 | z.....jo'HC...<.
        9c 6e da b7 12 04 28 14 07 23 72 c3 01 e6 c6 c9 | .n....(..#r.....
        b6 ff 86 3d c2 4e dc 7c 66 1f 62 0b 88 32 26 19 | ...=.N.|f.b..2&.
        0c cd 72 63 9c aa 39 ef 87 6b d1 2f 2f 3a 03 96 | ..rc..9..k.//:..
        7e 34 b9 06 1a 4f 6e f2 f3 0f 9f 4c 33 37 35 e0 | ~4...On....L375.
```

```
        93 f6 be 31 5c 6f b3 83 50 88 57 71 31 9f 1d 57 | ...1¥o..P.Wq1..W
        fb 9b 4b 8d 5d c1 66 b1 be ea f6 5e 00 15 91 13 | ..K.].f....^....
        76 ab c8 6b e4 ad 86 a9 57 ad fb e2 2c 45 c5 90 | v..k....W...,E..
        11 cc 6f bd 5e f3 00 22 00 0b 23 24 f6 f5 3c 45 | ..o.^.."..#$..<E
        70 7c 3e 0a d7 78 3e bc 01 ae cb d9 73 0a 54 49 | p|>..x>.....s.TI
```

…

```
Command: 31
        80 02 00 00 00 1f 00 00 01 22 40 00 00 01 01 80 | ........."@.....
        02 00 00 00 00 09 40 00 00 09 00 00 01 00 00    | ......@........
Response: 19
        80 02 00 00 00 13 00 00 00 00 00 00 00 00 00 00 | ................
        01 00 00                                        | ...
TPM2_NV_UndefineSpace: Auth 0x40000001, Idx 0x1800200
NV Test on index 0x1800200 with 1024 bytes passed
Command: 12
        80 01 00 00 00 0c 00 00 01 45 00 00             | .........E..
Response: 10
        80 01 00 00 00 0a 00 00 00 00                   | ..........
```

TPM ネイティブのテスト（デバッグメッセージ有効化）:

wolfTPM ビルド時の configure コマンドで次のようなデバッグメッセージ有効化指定をします。

```
./configure --enable-debug
```

またはヘッダーファイル内に `#define DEBUG_WOLFTPM` を指定します。

```
./examples/native/native_test
TPM2 Demo using Native API's
TPM2: Caps 0x30000697, Did 0x001b, Vid 0x15d1, Rid 0x10
Command: 12
        80 01 00 00 00 0c 00 00 01 44 00 00           | .........D..
Response: 10
        80 01 00 00 00 0a 00 00 01 00                 | ..........
TPM2_Startup pass
Command: 11
        80 01 00 00 00 0b 00 00 01 43 01              | .........C.
Response: 10
        80 01 00 00 00 0a 00 00 00 00                 | ..........
TPM2_SelfTest pass
Command: 10
        80 01 00 00 00 0a 00 00 01 7c                 | .........|
Response: 26
        80 01 00 00 00 1a 00 00 00 00 00 0a 00 01 f9 db | ................
        00 00 00 00 00 00 00 00 00 00                 | ..........
TPM2_GetTestResult: Size 10, Rc 0x0
        00 01 f9 db 00 00 00 00 00 00                 | ..........
Command: 16
        80 01 00 00 00 10 00 00 01 42 00 00 00 01 00 01 | .........B......
Response: 14
        80 01 00 00 00 0e 00 00 00 00 00 00 00 00      | ..............
TPM2_IncrementalSelfTest: Rc 0x0, Alg 0x1 (Todo 0)
Command: 22
```

```
        80 01 00 00 00 16 00 00 01 7a 00 00 00 06 00 00 | .........z......
        01 00 00 00 00 01                               | ......
Response: 27
        80 01 00 00 00 1b 00 00 00 00 01 00 00 00 06 00 | ................
        00 00 01 00 00 01 00 32 2e 30 00                | .......2.0.
TPM2_GetCapability: Property FamilyIndicator 0x322e3000
Command: 22
        80 01 00 00 00 16 00 00 01 7a 00 00 00 06 00 00 | .........z......
        01 12 00 00 00 01                               | ......
Response: 27
        80 01 00 00 00 1b 00 00 00 00 01 00 00 00 06 00 | ................
        00 00 01 00 00 01 12 00 00 00 18                | ...........
TPM2_GetCapability: Property PCR Count 24
Command: 12
        80 01 00 00 00 0c 00 00 01 7b 00 20             | .........{.
Response: 44
        80 01 00 00 00 2c 00 00 00 00 00 20 fb 1c d0 d0 | .....,..... ....
        5d ee a2 49 f7 b2 5a 38 72 87 7b aa 10 e8 33 12 | ]..I..Z8r.{...3.
        dc b7 78 54 46 6f 61 d9 7d 1f b7 0e             | ..xTFoa.}...
TPM2_GetRandom: Got 32 bytes
        fb 1c d0 d0 5d ee a2 49 f7 b2 5a 38 72 87 7b aa | ....]..I..Z8r.{.
        10 e8 33 12 dc b7 78 54 46 6f 61 d9 7d 1f b7 0e | ..3...xTFoa.}...
Command: 44
```

...

```
        80 01 00 00 00 0e 00 00 01 65 80 00 00 00       | .........e....
Response: 10
        80 01 00 00 00 0a 00 00 00 00                   | ..........
```

```
Command: 14
        80 01 00 00 00 0e 00 00 01 65 80 00 00 01       | .........e....
Response: 10
        80 01 00 00 00 0a 00 00 00 00                   | ..........
Command: 12
        80 01 00 00 00 0c 00 00 01 45 00 00             | .........E..
Response: 10
        80 01 00 00 00 0a 00 00 00 00                   | ..........
```

wolfSSL Inc.
www.wolfssl.jp, info@wolfssl.jp

wolfSSL 製品、サービスに関しては
Email: info@wolfssl.jp(日本語)、info@wolfssl.com(英語) まで
お問い合わせください。