JCATT ファイルフォーマット仕様書

NIST SP800-56B に記載された KAS1

2018 年 8 月

独立行政法人情報処理推進機構

目次

1　はじめに

暗号アルゴリズム実装試験ツール (以下 JCATT と略記する) が使用する各種ファイルのフォーマット規則を記述する．JCATT が使用するファイルには次のようなものがある．

ファイルの種類

- パラメータファイル (*.par)
  試験項目の設定を記述する．JCATT を用いて作成する．
- リクエストファイル (*.req)
  暗号モジュール開発ベンダに対する要求を記述する．JCATT を用いて作成する．
- Facts ファイル (*.fax)
  テストベクタを記述する．JCATT を用いて作成する．
- レスポンスファイル (*.rsp)
  ベンダからの回答を記述する．リクエストファイルおよび本稿で指定するファイルフォーマットに基づいてベンダが作成する．
- 結果ファイル (*.out)
  試験結果を記述する．JCATT を用いて作成する．

これらのファイルの名前は，次の規則に従ってつけること．

ファイル名の規則

- 拡張子は，上記 ( ) 内に指定したものを使用すること．
- 拡張子以外の名前は，試験対象実装ごとに同じ名称をつけること．
  リクエストファイル (*.req) と Facts ファイル (*.fax) の生成時には，リクエストファイル (*.req) と Facts ファイル (*.fax) に対してパラメータファイル (*.par) と同じ名称を JCATT が自動的につける．
  試験実行時には，同じ名称のレスポンスファイル (*.rsp) と Facts ファイル (*.fax) に対して試験が行われる．また，試験実行時は，結果ファイル (*.out) に対して，Facts ファイル (*.fax) と同じ名称を JCATT が自動的につける．

ファイルフォーマット詳細は次章以降に記述する．各ファイルに共通の規則は次の通りである．

共通規則

- JCATT 互換ファイルフォーマットの選択時, [　　] で囲まれた “タグ” の次の行に値を記述する．
- CAVS 準互換ファイルフォーマットの選択時, 〈 タグ 〉=〈 値 〉 の形式で 1 行で記述する．
- ヘッダ部分については各行について [〈 タグ 〉=〈 値 〉] の形式で 1 行で記述する．
- レスポンスファイルにおいては，【出力】と記述したタグが，試験対象実装が出力するデータを記述する箇所である．
- 半角英数字を用いること．
- タグおよび値は大文字小文字の区別をするので，大文字小文字を含めて正確に記述すること．ただし，数値を 16 進数で記述する場合は，大文字小文字は区別しない．
- 一文字目が ♯(半角) で始まるコメント行を自由に書き込むことができる．
- 平文，暗号文，鍵などのデータの区切り文字は改行 (CR+LF または LF) とする．
- 平文，暗号文，署名，鍵などのデータは 16 進表記とする．
- ビット数，個数などの数値は 10 進表記とする．
- ACSII コードを使用すること．
- 各行には必ず改行を入れること (最後のデータと EOF との間にも改行を入れること)．

2　NIST SP800-56B に記載された KAS1

鍵確立手法 KAS1 in NIST SP 800-56B の暗号アルゴリズム実装試験のためのファイルフォーマットを記述する.

`Algorithm Name` は，KAS1_in_NIST_SP800_56B.

試験方法の詳細は，暗号アルゴリズム実装試験仕様書を参照のこと.

2.1　CAVS 準互換ファイルフォーマット

この章で取り扱うファイルフォーマットでは, 公開鍵指数の種別として, 表1に記載された表現, 鍵導出関数識別子として, 表2に記載された表現, Key Confirmation に使う MAC アルゴリズム識別子として, 表3に記載された表現を用いる.

表1　公開鍵指数の種別

識別子	対応する公開鍵 e
TYPE1	$e = 65,537$
TYPE2	e はランダム

表2　鍵導出関数識別子

鍵導出関数識別子	対応する鍵導出関数	
SP800_56B_5_5_1_KDFConcat_SHA1	NIST SP 800-56B	Concatenation based KDF with SHA-1
SP800_56B_5_5_1_KDFConcat_SHA224		Concatenation based KDF with SHA-224
SP800_56B_5_5_1_KDFConcat_SHA256		Concatenation based KDF with SHA-256
SP800_56B_5_5_1_KDFConcat_SHA384		Concatenation based KDF with SHA-384
SP800_56B_5_5_1_KDFConcat_SHA512		Concatenation based KDF with SHA-512
SP800_56B_5_5_1_KDFConcat_SHA512_224		Concatenation based KDF with SHA-512/224
SP800_56B_5_5_1_KDFConcat_SHA512_256		Concatenation based KDF with SHA-512/256
SP800_56B_5_5_1_KDFConcat_SHA3_256		Concatenation based KDF with SHA3-256
SP800_56B_5_5_1_KDFConcat_SHA3_384		Concatenation based KDF with SHA3-384
SP800_56B_5_5_1_KDFConcat_SHA3_512		Concatenation based KDF with SHA3-512
SP800_56B_5_5_1_KDFConcat_HMAC_SHA1		Concatenation based KDF with HMAC-SHA-1
SP800_56B_5_5_1_KDFConcat_HMAC_SHA224		Concatenation based KDF with HMAC-SHA-224
SP800_56B_5_5_1_KDFConcat_HMAC_SHA256		Concatenation based KDF with HMAC-SHA-256
SP800_56B_5_5_1_KDFConcat_HMAC_SHA384		Concatenation based KDF with HMAC-SHA-384
SP800_56B_5_5_1_KDFConcat_HMAC_SHA512		Concatenation based KDF with HMAC-SHA-512
SP800_56B_5_5_1_KDFConcat_HMAC_SHA512_224		Concatenation based KDF with HMAC-SHA-512/224
SP800_56B_5_5_1_KDFConcat_HMAC_SHA512_256		Concatenation based KDF with HMAC-SHA-512/256
SP800_56B_5_5_1_KDFConcat_HMAC_SHA3_256		Concatenation based KDF with HMAC-SHA3-256
SP800_56B_5_5_1_KDFConcat_HMAC_SHA3_384		Concatenation based KDF with HMAC-SHA3-384
SP800_56B_5_5_1_KDFConcat_HMAC_SHA3_512		Concatenation based KDF with HMAC-SHA3-512
SP800_56B_5_5_1_KDFASN1_SHA1		ASN.1 based KDF with SHA-1
SP800_56B_5_5_1_KDFASN1_SHA224		ASN.1 based KDF with SHA-224
SP800_56B_5_5_1_KDFASN1_SHA256		ASN.1 based KDF with SHA-256
SP800_56B_5_5_1_KDFASN1_SHA384		ASN.1 based KDF with SHA-384
SP800_56B_5_5_1_KDFASN1_SHA512		ASN.1 based KDF with SHA-512
SP800_56B_5_5_1_KDFASN1_SHA512_224		ASN.1 based KDF with SHA-512/224
SP800_56B_5_5_1_KDFASN1_SHA512_256		ASN.1 based KDF with SHA-512/256
SP800_56B_5_5_1_KDFASN1_SHA3_256		ASN.1 based KDF with SHA3-256
SP800_56B_5_5_1_KDFASN1_SHA3_384		ASN.1 based KDF with SHA3-384
SP800_56B_5_5_1_KDFASN1_SHA3_512		ASN.1 based KDF with SHA3-512
SP800_56B_5_5_1_KDFASN1_HMAC_SHA1		ASN.1 based KDF with HMAC-SHA-1
SP800_56B_5_5_1_KDFASN1_HMAC_SHA224		ASN.1 based KDF with HMAC-SHA-224
SP800_56B_5_5_1_KDFASN1_HMAC_SHA256		ASN.1 based KDF with HMAC-SHA-256
SP800_56B_5_5_1_KDFASN1_HMAC_SHA384		ASN.1 based KDF with HMAC-SHA-384
SP800_56B_5_5_1_KDFASN1_HMAC_SHA512		ASN.1 based KDF with HMAC-SHA-512
SP800_56B_5_5_1_KDFASN1_HMAC_SHA512_224		ASN.1 based KDF with HMAC-SHA-512/224
SP800_56B_5_5_1_KDFASN1_HMAC_SHA512_256		ASN.1 based KDF with HMAC-SHA-512/256
SP800_56B_5_5_1_KDFASN1_HMAC_SHA3_256		ASN.1 based KDF with HMAC-SHA3-256
SP800_56B_5_5_1_KDFASN1_HMAC_SHA3_384		ASN.1 based KDF with HMAC-SHA3-384
SP800_56B_5_5_1_KDFASN1_HMAC_SHA3_512		ASN.1 based KDF with HMAC-SHA3-512
ANS_X942_7_7_2_KDFConcat_SHA1	ANS X9.42-2001	Concatenation based KDF with SHA-1
ANS_X942_7_7_2_KDFConcat_SHA224		Concatenation based KDF with SHA-224
ANS_X942_7_7_2_KDFConcat_SHA256		Concatenation based KDF with SHA-256
ANS_X942_7_7_2_KDFConcat_SHA384		Concatenation based KDF with SHA-384
ANS_X942_7_7_2_KDFConcat_SHA512		Concatenation based KDF with SHA-512
ANS_X942_7_7_2_KDFASN1_SHA1		ASN.1 based KDF with SHA-1
ANS_X942_7_7_2_KDFASN1_SHA224		ASN.1 based KDF with SHA-224
ANS_X942_7_7_2_KDFASN1_SHA256		ASN.1 based KDF with SHA-256
ANS_X942_7_7_2_KDFASN1_SHA384		ASN.1 based KDF with SHA-384
ANS_X942_7_7_2_KDFASN1_SHA512		ASN.1 based KDF with SHA-512

2.1.1　パラメータファイル (*.par)

表3　MAC アルゴリズム識別子

MAC アルゴリズム識別子	対応する MAC アルゴリズム
HMAC_SHA1	HMAC-SHA-1
HMAC_SHA224	HMAC-SHA-224
HMAC_SHA256	HMAC-SHA-256
HMAC_SHA384	HMAC-SHA-384
HMAC_SHA512	HMAC-SHA-512
HMAC_SHA512_224	HMAC-SHA-512/224
HMAC_SHA512_256	HMAC-SHA-512/256
HMAC_SHA3_256	HMAC-SHA3-256
HMAC_SHA3_384	HMAC-SHA3-384
HMAC_SHA3_512	HMAC-SHA3-512
CMAC_AES128	CMAC-AES-128
CMAC_AES192	CMAC-AES-192
CMAC_AES256	CMAC-AES-256

表4　NIST SP800-56B に記載された KAS1 パラメータファイル

機能	タグ		内容	表記
鍵共有	ヘッダ	AlgorithmName	KAS1_in_NIST_SP800_56B	文字列
		TargetFunction	KeyAgreement	文字列
		TargetRole	IUT が担う役割 (Party_U, Party_V)	文字列
		TypeOfPublicKey	Party V の公開鍵指数の種別 (65537:TYPE1, random:TYPE2)	文字列
		TypeOfPrivateKey	Party V のプライベート鍵の種別 (CRT なし:TYPE2)	文字列
		BitLengthOfModulusPartyV	Party V の公開鍵の法 n のビット長	10 進
		KDF	(鍵導出関数識別子)	文字列
		BitLengthOfSaltForHMACbasedKDF	HMAC ベースの KDF を使用する場合, salt のビット長	10 進
		BitLengthOfOI	OtherInfo のビット長	10 進
		BitLengthOfNonceV	Nonce_V のビット長	10 進
		ID_U	Party U の Identifier	16 進
		ID_V	Party V の Identifier	16 進
		KeyConfirmationSupported	サポートする Key confirmation のタイプ (Key Confirmation なし:NoKC, unilateral key confirmation from party V to party U:Unilateral_V_to_U)	文字列
		SelectedTestMethod	暗号アルゴリズム実装試験仕様書—鍵確立手法—の選択された試験項目の番号	10 進
		BitLengthOfDKM	DKM のビット長	10 進
		NumberOfDKM	DKM の個数	10 進
		RatioOfInvalidData	Unilateral key confirmation from party V to party U の選択時, 鍵確立に失敗するデータの割合. それ以外は省略.	浮動小数点
		MACforKeyConfirmation	Unilateral key confirmation from party V to party U の選択時, Key confirmation に使う MAC アルゴリズム. それ以外は省略.	文字列
		BitLengthOfMacKey	Unilateral key confirmation from party V to party U の選択時, Key confirmation に使う MacKey のビット長. それ以外は省略.	10 進
		BitLengthOfMacTag	Unilateral key confirmation from party V to party U の選択時, Key confirmation に使う MacTag のビット長. それ以外は省略.	10 進

2.1.2 リクエストファイル (*.req)

表5: NIST SP800-56B に記載された KAS1 リクエストファイル

機能	分類	タグ	内容	暗号アルゴリズム実装試験仕様書 —鍵確立手法— 上の表記との対応	値の表記	例示
鍵共有	ヘッダ	AlgorithmName	KAS1_in_NIST_SP800_56B		文字列	[AlgorithmName = KAS1_in_NIST_SP800_56B]
		TargetFunction	KeyAgreement		文字列	[TargetFunction = KeyAgreement]
		TargetRole	IUT が担う役割 (Party_U, Party_V)		文字列	[TargetRole = Party_U]
		TypeOfPublicKey	Party V の公開鍵指数の種別 (65537:TYPE1, random:TYPE2)		文字列	[TypeOfPublicKey = TYPE1]
		TypeOfPrivateKey	Party V のプライベート鍵の種別 (CRT なし:TYPE2)		文字列	[TypeOfPrivateKey = TYPE2]
		BitLengthOfModulusPartyV	Party V の公開鍵の法 n のビット長		10 進表記	[BitLengthOfModulusPartyV = 3072]
		KDF	鍵導出関数識別子		文字列	[KDF = SP800_56B_5_5_1_KDFConcat_SHA1]
		BitLengthOfSaltForHMACbasedKDF	HMAC ベースの KDF を使用する場合, salt のビット長		10 進表記	[BitLengthOfSaltForHMACbasedKDF = 0]
		BitLengthOfOI	OtherInfo のビット長		10 進表記	[BitLengthOfOI = 384]
		BitLengthOfNonceV	Nonce_V のビット長のビット長		10 進表記	[BitLengthOfNonceV = 256]
		ID_U	Party U の Identifier		16 進表記	[ID_U = a1b2c3d4e5]
		ID_V	Party V の Identifier		16 進表記	[ID_V = 4a434154546964]
		KeyConfirmationSupported	サポートする Key Confirmation のタイプ (Key Confirmation なし:NoKC, unilateral key confirmation from party V to party U:Unilateral_V_to_U)		文字列	[KeyConfirmationSupported = NoKC]
		SelectedTestMethod	暗号アルゴリズム実装試験仕様書—鍵確立手法—の選択された試験項目の番号 (試験 1:1, 試験 2:2)		10 進表記	[SelectedTestMethod = 1]
		BitLengthOfDKM	DKM のビット長	$KBits$	10 進表記	[BitLengthOfDKM = 320]
		NumberOfDKM	DKM の個数		10 進表記	[NumberOfDKM = 2048]
		RatioOfInvalidData	Unilateral key confirmation from party V to party U の選択時, 鍵確立に失敗するデータの割合. それ以外は省略.		浮動小数点表記	[RatioOfInvalidData = 0.5]
		MACforKeyConfirmation	Unilateral key confirmation from party V to party U の選択時, Key confirmation に使う MAC アルゴリズム. それ以外は省略.		文字列	[MACforKeyConfirmation = HMAC_SHA512]
		BitLengthOfMacKey	Unilateral key confirmation from party V to party U の選択時, Key confirmation に使う MacKey のビット長. それ以外は省略.		10 進表記	[BitLengthOfMacKey = 128]
		BitLengthOfMacTag	Unilateral key confirmation from party V to party U の選択時, Key confirmation に使う MacTag のビット長. それ以外は省略.		10 進表記	[BitLengthOfMacTag = 512]
	Party U 試験 1 ヘッダ	nV	Party V の公開鍵の法 n	n	16 進表記	nV = b73c . . . 9215
		eV	Party V の公開鍵指数 e	e	16 進表記	eV = 010001
	Party U 試験 1 本体 *1	COUNT	0 以上 NumberOfDKM 未満の整数		10 進表記	COUNT = 0
		NonceV	Nonce V		16 進表記	NonceV = 6695 . . . fe09
		OI	OtherInfo	$OtherInfo$	16 進表記	OI = a1b2 . . . 0e21
		SaltForHMACbasedKDF	HMAC を用いる鍵導出関数の選択時, HMAC の鍵として用いられる salt. それ以外は省略.	$salt$	16 進表記	SaltForHMACbasedKDF = eb22 . . . aee8
		C	暗号文	C	16 進表記	C = ?
		DKM	Party U が計算した $DerivedKeyingMaterial$	$DerivedKeyingMaterial$ (DKM)	16 進表記	DKM = ?
	Party U 試験 2 ヘッダ	nV	Party V の公開鍵の法 n	n	16 進表記	nV = b73c . . . 9215
		eV	Party V の公開鍵指数 e	e	16 進表記	eV = 010001
	Party U 試験 2 本体 *2	COUNT	0 以上 NumberOfDKM 未満の整数		10 進表記	COUNT = 0
		NonceV	Nonce V		16 進表記	NonceV = 6695 . . . fe09
		OI	OtherInfo	$OtherInfo$	16 進表記	OI = a1b2 . . . 0e21
		SaltForHMACbasedKDF	HMAC を用いる鍵導出関数の選択時, HMAC の鍵として用いられる salt. それ以外は省略.	$salt$	16 進表記	SaltForHMACbasedKDF = eb22 . . . aee8
		C	暗号文	C	16 進表記	C = 1d10 . . . de15
		Z	secret value Z	Z	16 進表記	Z = 457f . . . e64b
		MacDataV	MacData	$MacData$	16 進表記	MacDataV = 4b43 . . . de15
		MacTagV	MacTag	$MacTag_V$	16 進表記	MacTagV = 9d6e . . . ae0d
		KeyData	Party U が計算した $KeyData$	$KeyData$	16 進表記	KeyData = ?
		Result	鍵確立の成功又は失敗		文字列	Result = ?
	Party V 試験 1 ヘッダ	nV	Party V の公開鍵の法 n	n	16 進表記	nV = abf7 . . . b8e3
		eV	Party V の公開鍵指数 e	e	16 進表記	eV = 010001
		dV	Party V のプライベート鍵指数 d	d	16 進表記	dV = 53a0 . . . 77c1
	Party V 試験 1 本体 *3	COUNT	0 以上 NumberOfDKM 未満の整数		10 進表記	COUNT = 0
		NonceV	Nonce V		16 進表記	NonceV = 6695 . . . fe09
		OI	OtherInfo	$OtherInfo$	16 進表記	OI = a1b2 . . . 0e21
		SaltForHMACbasedKDF	HMAC を用いる鍵導出関数の選択時, HMAC の鍵として用いられる salt. それ以外は省略.	$salt$	16 進表記	SaltForHMACbasedKDF = eb22 . . . aee8
		C	暗号文	C	16 進表記	C = abf7 . . . 52ba
		DKM	Party V が計算した $DerivedKeyingMaterial$	$DerivedKeyingMaterial$ (DKM)	16 進表記	DKM = ?
		Result	鍵確立の成功又は失敗		文字列	Result = ?
	Party V 試験 2 ヘッダ	nV	Party V の公開鍵の法 n	n	16 進表記	nV = abf7 . . . b8e3
		eV	Party V の公開鍵指数 e	e	16 進表記	eV = 010001
		dV	Party V のプライベート鍵指数 d	d	16 進表記	dV = 53a0 . . . 77c1
	Party V 試験 2 本体 *4	COUNT	0 以上 NumberOfDKM 未満の整数		10 進表記	COUNT = 0
		NonceV	Nonce V		16 進表記	NonceV = 6695 . . . fe09
		OI	OtherInfo	$OtherInfo$	16 進表記	OI = a1b2 . . . 0e21
		SaltForHMACbasedKDF	HMAC を用いる鍵導出関数の選択時, HMAC の鍵として用いられる salt. それ以外は省略.	$salt$	16 進表記	SaltForHMACbasedKDF = eb22 . . . aee8
		C	暗号文	C	16 進表記	C = abf7 . . . 52ba
		MacDataV	MacData	$MacData$	16 進表記	MacDataV = 4b43 . . . de15
		MacTagV	Party V が計算した MacTag	$MacTag_V$	16 進表記	MacTagV = ?
		KeyData	Party V が計算した $KeyData$	$KeyData$	16 進表記	KeyData = ?

*1 NumberOfDKM 個の各データの組を以下のように記述する.

```
COUNT = 0                          ♯ i = 0 のデータの組について記述する.
NonceV = 6695 ... fe09             ♯ i = 0 に対応する Nonce V を記述する.
OI = a1b2 ... 0e21                 ♯ i = 0 に対応する OtherInfo を記述する.
SaltForHMACbasedKDF = eb22 ... aee8  ♯ HMAC を用いる鍵導出関数の選択時, HMAC の鍵として用いられる salt を記述する. それ以外は省略.
C = ?                              ♯ i = 0 に対応する暗号文 C のプレースホルダ.
DKM = ?                            ♯ i = 0 に対応する DerivedKeyingMaterial (DKM) のプレースホルダ.

COUNT = 1                          ♯ i = 1 のデータの組について記述する.
NonceV = 6695 ... fe09             ♯ i = 1 に対応する Nonce V を記述する.
OI = a1b2 ... 0e21                 ♯ i = 1 に対応する OtherInfo を記述する.
SaltForHMACbasedKDF = eb22 ... aee8  ♯ HMAC を用いる鍵導出関数の選択時, HMAC の鍵として用いられる salt を記述する. それ以外は省略.
C = ?                              ♯ i = 1 に対応する暗号文 C のプレースホルダ.
DKM = ?                            ♯ i = 1 に対応する DerivedKeyingMaterial (DKM) のプレースホルダ.
⋮
COUNT = ⟨NumberOfDKM − 1⟩          ♯ i = ⟨NumberOfDKM − 1⟩ のデータの組について記述する.
NonceV = 6695 ... fe09             ♯ i = ⟨NumberOfDKM − 1⟩ に対応する Nonce V を記述する.
OI = a1b2 ... 0e21                 ♯ i = ⟨NumberOfDKM − 1⟩ に対応する OtherInfo を記述する.
SaltForHMACbasedKDF = eb22 ... aee8  ♯ HMAC を用いる鍵導出関数の選択時, HMAC の鍵として用いられる salt を記述する. それ以外は省略.
C = ?                              ♯ i = ⟨NumberOfDKM − 1⟩ に対応する暗号文 C のプレースホルダ.
DKM = ?                            ♯ i = ⟨NumberOfDKM − 1⟩ に対応する DerivedKeyingMaterial (DKM) のプレースホルダ.
```

*2 NumberOfDKM 個の各データの組を以下のように記述する.

```
COUNT = 0                          ♯ i = 0 のデータの組について記述する.
NonceV = 6695 ... fe09             ♯ i = 0 に対応する Nonce V を記述する.
OI = a1b2 ... 0e21                 ♯ i = 0 に対応する OtherInfo を記述する.
SaltForHMACbasedKDF = eb22 ... aee8  ♯ HMAC を用いる鍵導出関数の選択時, HMAC の鍵として用いられる salt を記述する. それ以外は省略.
C = 1d10 ... de15                  ♯ i = 0 に対応する暗号文 C を記述する.
Z = 457f ... e64b                  ♯ i = 0 に対応する, Party U が暗号文 C の生成に用いた secret value Z を記述する.
MacDataV = 457f ... e64b           ♯ i = 0 に対応する, MacTag_V の生成に用いた MacData を記述する.
MacTagV = 457f ... e64b            ♯ i = 0 に対応する, MacTag_V を記述する.
KeyData = ?                        ♯ i = 0 に対応する, KeyData のプレースホルダ.
Result = ?                         ♯ i = 0 に対応する, 鍵確立の成功又は失敗のプレースホルダ.

COUNT = 1                          ♯ i = 1 のデータの組について記述する.
NonceV = 6695 ... fe09             ♯ i = 1 に対応する Nonce V を記述する.
OI = a1b2 ... 0e21                 ♯ i = 1 に対応する OtherInfo を記述する.
SaltForHMACbasedKDF = eb22 ... aee8  ♯ HMAC を用いる鍵導出関数の選択時, HMAC の鍵として用いられる salt を記述する. それ以外は省略.
C = 1d10 ... de15                  ♯ i = 1 に対応する暗号文 C を記述する.
Z = 457f ... e64b                  ♯ i = 1 に対応する, Party U が暗号文 C の生成に用いた secret value Z を記述する.
MacDataV = 457f ... e64b           ♯ i = 1 に対応する, MacTag_V の生成に用いた MacData を記述する.
MacTagV = 457f ... e64b            ♯ i = 1 に対応する, MacTag_V を記述する.
KeyData = ?                        ♯ i = 1 に対応する, KeyData のプレースホルダ.
Result = ?                         ♯ i = 1 に対応する, 鍵確立の成功又は失敗のプレースホルダ.
⋮
COUNT = ⟨NumberOfDKM − 1⟩          ♯ i = ⟨NumberOfDKM − 1⟩ のデータの組について記述する.
NonceV = 6695 ... fe09             ♯ i = ⟨NumberOfDKM − 1⟩ に対応する Nonce V を記述する.
OI = a1b2 ... 0e21                 ♯ i = ⟨NumberOfDKM − 1⟩ に対応する OtherInfo を記述する.
SaltForHMACbasedKDF = eb22 ... aee8  ♯ HMAC を用いる鍵導出関数の選択時, HMAC の鍵として用いられる salt を記述する. それ以外は省略.
C = 1d10 ... de15                  ♯ i = ⟨NumberOfDKM − 1⟩ に対応する暗号文 C を記述する.
Z = 457f ... e64b                  ♯ i = ⟨NumberOfDKM − 1⟩ に対応する, Party U が暗号文 C の生成に用いた secret value Z を記述する.
MacDataV = 457f ... e64b           ♯ i = ⟨NumberOfDKM − 1⟩ に対応する, MacTag_V の生成に用いた MacData を記述する.
MacTagV = 457f ... e64b            ♯ i = ⟨NumberOfDKM − 1⟩ に対応する, MacTag_V を記述する.
KeyData = ?                        ♯ i = ⟨NumberOfDKM − 1⟩ に対応する, KeyData のプレースホルダ.
Result = ?                         ♯ i = ⟨NumberOfDKM − 1⟩ に対応する, 鍵確立の成功又は失敗のプレースホルダ.
```

∗3 NumberOfDKM 個の各データの組を以下のように記述する.

```
COUNT = 0                          ♯ i = 0 のデータの組について記述する.
NonceV = 6695 ... fe09             ♯ i = 0 に対応する Nonce V を記述する.
OI = a1b2 ... 0e21                 ♯ i = 0 に対応する OtherInfo を記述する.
SaltForHMACbasedKDF = eb22 ... aee8  ♯ HMAC を用いる鍵導出関数の選択時, HMAC の鍵として用いられる salt を記述する. それ以外は省略.
C = abf7 ... 52ba                  ♯ i = 0 に対応する暗号文 C を記述する.
DKM = ?                            ♯ i = 0 に対応する DerivedKeyingMaterial (DKM) のプレースホルダ.
Result = ?                         ♯ i = 0 に対応する, 鍵確立の成功又は失敗のプレースホルダ.

COUNT = 1                          ♯ i = 1 のデータの組について記述する.
NonceV = 6695 ... fe09             ♯ i = 1 に対応する Nonce V を記述する.
OI = a1b2 ... 0e21                 ♯ i = 1 に対応する OtherInfo を記述する.
SaltForHMACbasedKDF = eb22 ... aee8  ♯ HMAC を用いる鍵導出関数の選択時, HMAC の鍵として用いられる salt を記述する. それ以外は省略.
C = abf7 ... 52ba                  ♯ i = 1 に対応する暗号文 C を記述する.
DKM = ?                            ♯ i = 1 に対応する DerivedKeyingMaterial (DKM) のプレースホルダ.
Result = ?                         ♯ i = 1 に対応する, 鍵確立の成功又は失敗のプレースホルダ.
⋮
COUNT = ⟨NumberOfDKM − 1⟩          ♯ i = ⟨NumberOfDKM − 1⟩ のデータの組について記述する.
NonceV = 6695 ... fe09             ♯ i = ⟨NumberOfDKM − 1⟩ に対応する Nonce V を記述する.
OI = a1b2 ... 0e21                 ♯ i = ⟨NumberOfDKM − 1⟩ に対応する OtherInfo を記述する.
SaltForHMACbasedKDF = eb22 ... aee8  ♯ HMAC を用いる鍵導出関数の選択時, HMAC の鍵として用いられる salt を記述する. それ以外は省略.
C = abf7 ... 52ba                  ♯ i = ⟨NumberOfDKM − 1⟩ に対応する暗号文 C を記述する.
DKM = ?                            ♯ i = ⟨NumberOfDKM − 1⟩ に対応する DerivedKeyingMaterial (DKM) のプレースホルダ.
Result = ?                         ♯ i = ⟨NumberOfDKM − 1⟩ に対応する, 鍵確立の成功又は失敗のプレースホルダ.
```

∗4 NumberOfDKM 個の各データの組を以下のように記述する.

```
COUNT = 0                          ♯ i = 0 のデータの組について記述する.
NonceV = 6695 ... fe09             ♯ i = 0 に対応する Nonce V を記述する.
OI = a1b2 ... 0e21                 ♯ i = 0 に対応する OtherInfo を記述する.
SaltForHMACbasedKDF = eb22 ... aee8  ♯ HMAC を用いる鍵導出関数の選択時, HMAC の鍵として用いられる salt を記述する. それ以外は省略.
C = abf7 ... 52ba                  ♯ i = 0 に対応する暗号文 C を記述する.
MacDataV = 457f ... e64b           ♯ i = 0 に対応する, MacTag_V の生成に用いる MacData を記述する.
MacTagV = ?                        ♯ i = 0 に対応する, MacTag_V のプレースホルダ.
KeyData = ?                        ♯ i = 0 に対応する, KeyData のプレースホルダ.

COUNT = 1                          ♯ i = 1 のデータの組について記述する.
NonceV = 6695 ... fe09             ♯ i = 1 に対応する Nonce V を記述する.
OI = a1b2 ... 0e21                 ♯ i = 1 に対応する OtherInfo を記述する.
SaltForHMACbasedKDF = eb22 ... aee8  ♯ HMAC を用いる鍵導出関数の選択時, HMAC の鍵として用いられる salt を記述する. それ以外は省略.
C = abf7 ... 52ba                  ♯ i = 1 に対応する暗号文 C を記述する.
MacDataV = 457f ... e64b           ♯ i = 1 に対応する, MacTag_V の生成に用いる MacData を記述する.
MacTagV = ?                        ♯ i = 1 に対応する, MacTag_V のプレースホルダ.
KeyData = ?                        ♯ i = 1 に対応する, KeyData のプレースホルダ.
⋮
COUNT = ⟨NumberOfDKM − 1⟩          ♯ i = ⟨NumberOfDKM − 1⟩ のデータの組について記述する.
NonceV = 6695 ... fe09             ♯ i = ⟨NumberOfDKM − 1⟩ に対応する Nonce V を記述する.
OI = a1b2 ... 0e21                 ♯ i = ⟨NumberOfDKM − 1⟩ に対応する OtherInfo を記述する.
SaltForHMACbasedKDF = eb22 ... aee8  ♯ HMAC を用いる鍵導出関数の選択時, HMAC の鍵として用いられる salt を記述する. それ以外は省略.
C = abf7 ... 52ba                  ♯ i = ⟨NumberOfDKM − 1⟩ に対応する暗号文 C を記述する.
MacDataV = 457f ... e64b           ♯ i = ⟨NumberOfDKM − 1⟩ に対応する, MacTag_V の生成に用いる MacData を記述する.
MacTagV = ?                        ♯ i = ⟨NumberOfDKM − 1⟩ に対応する, MacTag_V のプレースホルダ.
KeyData = ?                        ♯ i = ⟨NumberOfDKM − 1⟩ に対応する, KeyData のプレースホルダ.
```

2.1.3 Facts ファイル (*.fax)

表6: NIST SP800-56B に記載された KAS1 Facts ファイル

機能	分類	タグ	内容	暗号アルゴリズム実装試験仕様書 —鍵確立手法— 上の表記との対応	値の表記	例示
鍵共有	ヘッダ	AlgorithmName	KAS1_in_NIST_SP800_56B		文字列	[AlgorithmName = KAS1_in_NIST_SP800_56B]
		TargetFunction	KeyAgreement		文字列	[TargetFunction = KeyAgreement]
		TargetRole	IUT が担う役割 (Party_U, Party_V)		文字列	[TargetRole = Party_U]
		TypeOfPublicKey	Party V の公開鍵指数の種別 (65537:TYPE1, random:TYPE2)		文字列	[TypeOfPublicKey = TYPE1]
		TypeOfPrivateKey	Party V のプライベート鍵の種別 (CRT なし:TYPE2)		文字列	[TypeOfPrivateKey = TYPE2]
		BitLengthOfModulusPartyV	Party V の公開鍵の法 n のビット長		10 進表記	[BitLengthOfModulusPartyV = 3072]
		KDF	鍵導出関数識別子		文字列	[KDF = SP800_56B_5_5_1_KDFConcat_SHA1]
		BitLengthOfSaltForHMACbasedKDF	HMAC ベースの KDF を使用する場合, salt のビット長		10 進表記	[BitLengthOfSaltForHMACbasedKDF = 0]
		BitLengthOfOI	OtherInfo のビット長		10 進表記	[BitLengthOfOI = 384]
		BitLengthOfNonceV	Nonce_V のビット長のビット長		10 進表記	[BitLengthOfNonceV = 256]
		ID_U	Party U の Identifier		16 進表記	[ID_U = a1b2c3d4e5]
		ID_V	Party V の Identifier		16 進表記	[ID_V = 4a434154546964]
		KeyConfirmationSupported	サポートする Key Confirmation のタイプ (Key Confirmation なし:NoKC, unilateral key confirmation from party V to party U:Unilateral_V_to_U)		文字列	[KeyConfirmationSupported = NoKC]
		SelectedTestMethod	暗号アルゴリズム実装試験仕様書—鍵確立手法—の選択された試験項目の番号 (試験 1:1, 試験 2:2)		10 進表記	[SelectedTestMethod = 1]
		BitLengthOfDKM	DKM のビット長	$KBits$	10 進表記	[BitLengthOfDKM = 320]
		NumberOfDKM	DKM の個数		10 進表記	[NumberOfDKM = 2048]
		RatioOfInvalidData	Unilateral key confirmation from party V to party U の選択時, 鍵確立に失敗するデータの割合. それ以外は省略.		浮動小数点表記	[RatioOfInvalidData = 0.5]
		MACforKeyConfirmation	Unilateral key confirmation from party V to party U の選択時, Key confirmation に使う MAC アルゴリズム. それ以外は省略.		文字列	[MACforKeyConfirmation = HMAC_SHA512]
		BitLengthOfMacKey	Unilateral key confirmation from party V to party U の選択時, Key confirmation に使う MacKey のビット長. それ以外は省略.		10 進表記	[BitLengthOfMacKey = 128]
		BitLengthOfMacTag	Unilateral key confirmation from party V to party U の選択時, Key confirmation に使う MacTag のビット長. それ以外は省略.		10 進表記	[BitLengthOfMacTag = 512]
	Party U 試験 1 ヘッダ	nV	Party V の公開鍵の法 n	n	16 進表記	nV = cc57 . . . 406d
		eV	Party V の公開鍵指数 e	e	16 進表記	eV = 010001
		dV	Party V のプライベート鍵指数 d	d	16 進表記	dV = 0d0c . . . 7041
		pV	Party V の素数 p		16 進表記	pV = d2cd . . . 97bd
		qV	Party V の素数 q		16 進表記	qV = f827 . . . be71
	Party U 試験 1 本体 *1	COUNT	0 以上 NumberOfDKM 未満の整数		10 進表記	COUNT = 0
		NonceV	Nonce V		16 進表記	NonceV = 6695 . . . fe09
		OI	OtherInfo	$OtherInfo$	16 進表記	OI = a1b2 . . . 0e21
		SaltForHMACbasedKDF	HMAC を用いる鍵導出関数の選択時, HMAC の鍵として用いられる salt. それ以外は省略.	$salt$	16 進表記	SaltForHMACbasedKDF = eb22 . . . aee8
	Party U 試験 2 ヘッダ	nV	Party V の公開鍵の法 n	n	16 進表記	nV = cc57 . . . 406d
		eV	Party V の公開鍵指数 e	e	16 進表記	eV = 010001
		dV	Party V のプライベート鍵指数 d	d	16 進表記	dV = 0d0c . . . 7041
		pV	Party V の素数 p		16 進表記	pV = d2cd . . . 97bd
		qV	Party V の素数 q		16 進表記	qV = f827 . . . be71
	Party U 試験 2 本体 *2	COUNT	0 以上 NumberOfDKM 未満の整数		10 進表記	COUNT = 0
		NonceV	Nonce V		16 進表記	NonceV = 6695 . . . fe09
		OI	OtherInfo	$OtherInfo$	16 進表記	OI = a1b2 . . . 0e21
		SaltForHMACbasedKDF	HMAC を用いる鍵導出関数の選択時, HMAC の鍵として用いられる salt. それ以外は省略.	$salt$	16 進表記	SaltForHMACbasedKDF = eb22 . . . aee8
		C	暗号文	C	16 進表記	C = 1d10 . . . de15
		Z	secret value Z	Z	16 進表記	Z = 457f . . . e64b
		MacDataV	MacData	$MacData$	16 進表記	MacDataV = 4b43 . . . de15
		MacTagV	MacTag	$MacTag_V$	16 進表記	MacTagV = 9d6e . . . ae0d
		KeyData	$KeyData$ の期待値	$KeyData$	16 進表記	KeyData = 159c . . . a983
		Result	鍵確立の成功又は失敗 (成功:P, 失敗:F)		文字列	Result = P
	Party V 試験 1 ヘッダ	nV	Party V の公開鍵の法 n	n	16 進表記	nV = cc57 . . . 406d
		eV	Party V の公開鍵指数 e	e	16 進表記	eV = 010001
		dV	Party V のプライベート鍵指数 d	d	16 進表記	dV = 0d0c . . . 7041
		pV	Party V の素数 p		16 進表記	pV = d2cd . . . 97bd
		qV	Party V の素数 q		16 進表記	qV = f827 . . . be71
	Party V 試験 1 本体 *3	COUNT	0 以上 NumberOfDKM 未満の整数		10 進表記	COUNT = 0
		NonceV	Nonce V		16 進表記	NonceV = 6695 . . . fe09
		OI	OtherInfo	$OtherInfo$	16 進表記	OI = a1b2 . . . 0e21
		SaltForHMACbasedKDF	HMAC を用いる鍵導出関数の選択時, HMAC の鍵として用いられる salt. それ以外は省略.	$salt$	16 進表記	SaltForHMACbasedKDF = eb22 . . . aee8
		C	暗号文	C	16 進表記	C = abf7 . . . 52ba
		Z	Party U が暗号文 C の計算に用いた secret value Z. 暗号文 C が定義域外の場合にはダミーデータを記述する.	Z	16 進表記	Z = 8b33 . . . dceb
		DKM	$DerivedKeyingMaterial$ の期待値. 暗号文 C が定義域外の場合にはダミーデータを記述する.	$DerivedKeyingMaterial$ (DKM)	16 進表記	DKM = 9011 . . . 71f8
		Result	鍵確立の成功又は失敗 (成功:P, 失敗:F)		文字列	Result = P
	Party V 試験 2 ヘッダ	nV	Party V の公開鍵の法 n	n	16 進表記	nV = cc57 . . . 406d
		eV	Party V の公開鍵指数 e	e	16 進表記	eV = 010001
		dV	Party V のプライベート鍵指数 d	d	16 進表記	dV = 0d0c . . . 7041
		pV	Party V の素数 p		16 進表記	pV = d2cd . . . 97bd
		qV	Party V の素数 q		16 進表記	qV = f827 . . . be71
	Party V 試験 2 本体 *4	COUNT	0 以上 NumberOfDKM 未満の整数		10 進表記	COUNT = 0
		NonceV	Nonce V		16 進表記	NonceV = 6695 . . . fe09
		OI	OtherInfo	$OtherInfo$	16 進表記	OI = a1b2 . . . 0e21
		SaltForHMACbasedKDF	HMAC を用いる鍵導出関数の選択時, HMAC の鍵として用いられる salt. それ以外は省略.	$salt$	16 進表記	SaltForHMACbasedKDF = eb22 . . . aee8
		C	暗号文	C	16 進表記	C = abf7 . . . 52ba
		Z	Party U が暗号文 C の計算に用いた secret value Z.	Z	16 進表記	Z = 8b33 . . . dceb
		MacDataV	MacData	$MacData$	16 進表記	MacDataV = 4b43 . . . de15
		MacTagV	MacTag	$MacTag_V$	16 進表記	MacTagV = e98a . . . d160
		KeyData	$KeyData$	$KeyData$	16 進表記	KeyData = 985a . . . 6ea2

*1 NumberOfDKM 個の各データの組を以下のように記述する.

```
COUNT = 0                          ♯ i = 0 のデータの組について記述する.
NonceV = 6695 ... fe09             ♯ i = 0 に対応する Nonce V を記述する.
OI = a1b2 ... 0e21                 ♯ i = 0 に対応する OtherInfo を記述する.
SaltForHMACbasedKDF = eb22 ... aee8  ♯ HMAC を用いる鍵導出関数の選択時, HMAC の鍵として用いられる salt を記述する. それ以外は省略.

COUNT = 1                          ♯ i = 1 のデータの組について記述する.
NonceV = 6695 ... fe09             ♯ i = 1 に対応する Nonce V を記述する.
OI = a1b2 ... 0e21                 ♯ i = 1 に対応する OtherInfo を記述する.
SaltForHMACbasedKDF = eb22 ... aee8  ♯ HMAC を用いる鍵導出関数の選択時, HMAC の鍵として用いられる salt を記述する. それ以外は省略.
⋮
COUNT = ⟨NumberOfDKM − 1⟩          ♯ i = ⟨NumberOfDKM − 1⟩ のデータの組について記述する.
NonceV = 6695 ... fe09             ♯ i = ⟨NumberOfDKM − 1⟩ に対応する Nonce V を記述する.
OI = a1b2 ... 0e21                 ♯ i = ⟨NumberOfDKM − 1⟩ に対応する OtherInfo を記述する.
SaltForHMACbasedKDF = eb22 ... aee8  ♯ HMAC を用いる鍵導出関数の選択時, HMAC の鍵として用いられる salt を記述する. それ以外は省略.
```

*2 NumberOfDKM 個の各データの組を以下のように記述する.

```
COUNT = 0                          ♯ i = 0 のデータの組について記述する.
NonceV = 6695 ... fe09             ♯ i = 0 に対応する Nonce V を記述する.
OI = a1b2 ... 0e21                 ♯ i = 0 に対応する OtherInfo を記述する.
SaltForHMACbasedKDF = eb22 ... aee8  ♯ HMAC を用いる鍵導出関数の選択時, HMAC の鍵として用いられる salt を記述する. それ以外は省略.
C = 1d10 ... de15                  ♯ i = 0 に対応する暗号文 C を記述する.
Z = 457f ... e64b                  ♯ i = 0 に対応する, Party U が暗号文 C の生成に用いた secret value Z を記述する.
MacDataV = 457f ... e64b           ♯ i = 0 に対応する, MacTag_V の生成に用いた MacData を記述する.
MacTagV = 457f ... e64b            ♯ i = 0 に対応する, MacTag_V を記述する.
KeyData = 159c ... a983            ♯ i = 0 に対応する, KeyData の期待値を記述する.
Result = P                         ♯ i = 0 に対応する, 鍵確立の成功又は失敗の期待値を記述する.

COUNT = 1                          ♯ i = 1 のデータの組について記述する.
NonceV = 6695 ... fe09             ♯ i = 1 に対応する Nonce V を記述する.
OI = a1b2 ... 0e21                 ♯ i = 1 に対応する OtherInfo を記述する.
SaltForHMACbasedKDF = eb22 ... aee8  ♯ HMAC を用いる鍵導出関数の選択時, HMAC の鍵として用いられる salt を記述する. それ以外は省略.
C = 1d10 ... de15                  ♯ i = 1 に対応する暗号文 C を記述する.
Z = 457f ... e64b                  ♯ i = 1 に対応する, Party U が暗号文 C の生成に用いた secret value Z を記述する.
MacDataV = 457f ... e64b           ♯ i = 1 に対応する, MacTag_V の生成に用いた MacData を記述する.
MacTagV = 457f ... e64b            ♯ i = 1 に対応する, MacTag_V を記述する.
KeyData = 159c ... a983            ♯ i = 1 に対応する, KeyData の期待値を記述する.
Result = P                         ♯ i = 1 に対応する, 鍵確立の成功又は失敗の期待値を記述する.
⋮
COUNT = ⟨NumberOfDKM − 1⟩          ♯ i = ⟨NumberOfDKM − 1⟩ のデータの組について記述する.
NonceV = 6695 ... fe09             ♯ i = ⟨NumberOfDKM − 1⟩ に対応する Nonce V を記述する.
OI = a1b2 ... 0e21                 ♯ i = ⟨NumberOfDKM − 1⟩ に対応する OtherInfo を記述する.
SaltForHMACbasedKDF = eb22 ... aee8  ♯ HMAC を用いる鍵導出関数の選択時, HMAC の鍵として用いられる salt を記述する. それ以外は省略.
C = 1d10 ... de15                  ♯ i = ⟨NumberOfDKM − 1⟩ に対応する暗号文 C を記述する.
Z = 457f ... e64b                  ♯ i = ⟨NumberOfDKM − 1⟩ に対応する, Party U が暗号文 C の生成に用いた secret value Z を記述する.
MacDataV = 457f ... e64b           ♯ i = ⟨NumberOfDKM − 1⟩ に対応する, MacTag_V の生成に用いた MacData を記述する.
MacTagV = 457f ... e64b            ♯ i = ⟨NumberOfDKM − 1⟩ に対応する, MacTag_V を記述する.
KeyData = 159c ... a983            ♯ i = ⟨NumberOfDKM − 1⟩ に対応する, KeyData の期待値を記述する.
Result = P                         ♯ i = ⟨NumberOfDKM − 1⟩ に対応する, 鍵確立の成功又は失敗の期待値を記述する.
```

∗3 NumberOfDKM 個の各データの組を以下のように記述する.

```
COUNT = 0                          ♯ i = 0 のデータの組について記述する.
NonceV = 6695 ... fe09             ♯ i = 0 に対応する Nonce V を記述する.
OI = a1b2 ... 0e21                 ♯ i = 0 に対応する OtherInfo を記述する.
SaltForHMACbasedKDF = eb22 ... aee8  ♯ HMAC を用いる鍵導出関数の選択時, HMAC の鍵として用いられる salt を記述する. それ以外は省略.
C = abf7 ... 52ba                  ♯ i = 0 に対応する暗号文 C を記述する.
Z = 8b33 ... dceb                  ♯ i = 0 に対応する, Party U が暗号文 C の生成に用いた secret value Z の期待値を記述する. 暗号文 C が定義域外の場合にはダミーデータを記述する.
DKM = 9011 ... 71f8                ♯ i = 0 に対応する DerivedKeyingMaterial (DKM) の期待値を記述する.
Result = P                         ♯ i = 0 に対応する, 鍵確立の成功又は失敗の期待値を記述する.

COUNT = 1                          ♯ i = 1 のデータの組について記述する.
NonceV = 6695 ... fe09             ♯ i = 1 に対応する Nonce V を記述する.
OI = a1b2 ... 0e21                 ♯ i = 1 に対応する OtherInfo を記述する.
SaltForHMACbasedKDF = eb22 ... aee8  ♯ HMAC を用いる鍵導出関数の選択時, HMAC の鍵として用いられる salt を記述する. それ以外は省略.
C = abf7 ... 52ba                  ♯ i = 1 に対応する暗号文 C を記述する.
Z = 8b33 ... dceb                  ♯ i = 1 に対応する, Party U が暗号文 C の生成に用いた secret value Z の期待値を記述する. 暗号文 C が定義域外の場合にはダミーデータを記述する.
DKM = 9011 ... 71f8                ♯ i = 1 に対応する DerivedKeyingMaterial (DKM) の期待値を記述する.
Result = P                         ♯ i = 1 に対応する, 鍵確立の成功又は失敗の期待値を記述する.
⋮
COUNT = ⟨NumberOfDKM − 1⟩          ♯ i = ⟨NumberOfDKM − 1⟩ のデータの組について記述する.
NonceV = 6695 ... fe09             ♯ i = ⟨NumberOfDKM − 1⟩ に対応する Nonce V を記述する.
OI = a1b2 ... 0e21                 ♯ i = ⟨NumberOfDKM − 1⟩ に対応する OtherInfo を記述する.
SaltForHMACbasedKDF = eb22 ... aee8  ♯ HMAC を用いる鍵導出関数の選択時, HMAC の鍵として用いられる salt を記述する. それ以外は省略.
C = abf7 ... 52ba                  ♯ i = ⟨NumberOfDKM − 1⟩ に対応する暗号文 C を記述する.
Z = 8b33 ... dceb                  ♯ i = ⟨NumberOfDKM − 1⟩ に対応する, Party U が暗号文 C の生成に用いた secret value Z の期待値を記述する. 暗号文 C が定義域外の場合にはダミーデータを
DKM = 9011 ... 71f8                ♯ i = ⟨NumberOfDKM − 1⟩ に対応する DerivedKeyingMaterial (DKM) の期待値を記述する.
Result = P                         ♯ i = ⟨NumberOfDKM − 1⟩ に対応する, 鍵確立の成功又は失敗の期待値を記述する.
```

∗4 NumberOfDKM 個の各データの組を以下のように記述する.

```
COUNT = 0                          ♯ i = 0 のデータの組について記述する.
NonceV = 6695 ... fe09             ♯ i = 0 に対応する Nonce V を記述する.
OI = a1b2 ... 0e21                 ♯ i = 0 に対応する OtherInfo を記述する.
SaltForHMACbasedKDF = eb22 ... aee8  ♯ HMAC を用いる鍵導出関数の選択時, HMAC の鍵として用いられる salt を記述する. それ以外は省略.
C = abf7 ... 52ba                  ♯ i = 0 に対応する暗号文 C を記述する.
MacDataV = 457f ... e64b           ♯ i = 0 に対応する, MacTag_V の生成に用いる MacData を記述する.
MacTagV = e98a ... d160            ♯ i = 0 に対応する, MacTag_V の期待値を記述する.
KeyData = 985a ... 6ea2            ♯ i = 0 に対応する, KeyData の期待値を記述する.

COUNT = 1                          ♯ i = 1 のデータの組について記述する.
NonceV = 6695 ... fe09             ♯ i = 1 に対応する Nonce V を記述する.
OI = a1b2 ... 0e21                 ♯ i = 1 に対応する OtherInfo を記述する.
SaltForHMACbasedKDF = eb22 ... aee8  ♯ HMAC を用いる鍵導出関数の選択時, HMAC の鍵として用いられる salt を記述する. それ以外は省略.
C = abf7 ... 52ba                  ♯ i = 1 に対応する暗号文 C を記述する.
MacDataV = 457f ... e64b           ♯ i = 1 に対応する, MacTag_V の生成に用いる MacData を記述する.
MacTagV = e98a ... d160            ♯ i = 1 に対応する, MacTag_V の期待値を記述する.
KeyData = 985a ... 6ea2            ♯ i = 1 に対応する, KeyData の期待値を記述する.
⋮
COUNT = ⟨NumberOfDKM − 1⟩          ♯ i = ⟨NumberOfDKM − 1⟩ のデータの組について記述する.
NonceV = 6695 ... fe09             ♯ i = ⟨NumberOfDKM − 1⟩ に対応する Nonce V を記述する.
OI = a1b2 ... 0e21                 ♯ i = ⟨NumberOfDKM − 1⟩ に対応する OtherInfo を記述する.
SaltForHMACbasedKDF = eb22 ... aee8  ♯ HMAC を用いる鍵導出関数の選択時, HMAC の鍵として用いられる salt を記述する. それ以外は省略.
C = abf7 ... 52ba                  ♯ i = ⟨NumberOfDKM − 1⟩ に対応する暗号文 C を記述する.
MacDataV = 457f ... e64b           ♯ i = ⟨NumberOfDKM − 1⟩ に対応する, MacTag_V の生成に用いる MacData を記述する.
MacTagV = e98a ... d160            ♯ i = ⟨NumberOfDKM − 1⟩ に対応する, MacTag_V の期待値を記述する.
KeyData = 985a ... 6ea2            ♯ i = ⟨NumberOfDKM − 1⟩ に対応する, KeyData の期待値を記述する.
```

2.1.4 レスポンスファイル (*.rsp)

表7: NIST SP800-56B に記載された KAS1 レスポンスファイル

機能	分類	タグ	内容	暗号アルゴリズム実装試験仕様書 —鍵確立手法— 上の表記との対応	値の表記	例示
鍵共有	ヘッダ	AlgorithmName	KAS1_in_NIST_SP800_56B		文字列	[AlgorithmName = KAS1_in_NIST_SP800_56B]
		TargetFunction	KeyAgreement		文字列	[TargetFunction = KeyAgreement]
		TargetRole	IUT が担う役割 (Party_U, Party_V)		文字列	[TargetRole = Party_U]
		TypeOfPublicKey	Party V の公開鍵指数の種別 (65537:TYPE1, random:TYPE2)		文字列	[TypeOfPublicKey = TYPE1]
		TypeOfPrivateKey	Party V のプライベート鍵の種別 (CRT なし:TYPE2)		文字列	[TypeOfPrivateKey = TYPE2]
		BitLengthOfModulusPartyV	Party V の公開鍵の法 n のビット長		10 進表記	[BitLengthOfModulusPartyV = 3072]
		KDF	鍵導出関数識別子		文字列	[KDF = SP800_56B_5_5_1_KDFConcat_SHA1]
		BitLengthOfSaltForHMACbasedKDF	HMAC ベースの KDF を使用する場合, salt のビット長		10 進表記	[BitLengthOfSaltForHMACbasedKDF = 0]
		BitLengthOfOI	OtherInfo のビット長		10 進表記	[BitLengthOfOI = 384]
		BitLengthOfNonceV	Nonce_V のビット長のビット長		10 進表記	[BitLengthOfNonceV = 256]
		ID_U	Party U の Identifier		16 進表記	[ID_U = a1b2c3d4e5]
		ID_V	Party V の Identifier		16 進表記	[ID_V = 4a434154546964]
		KeyConfirmationSupported	サポートする Key Confirmation のタイプ (Key Confirmation なし:NoKC, unilateral key confirmation from party V to party U:Unilateral_V_to_U)		文字列	[KeyConfirmationSupported = NoKC]
		SelectedTestMethod	暗号アルゴリズム実装試験仕様書—鍵確立手法—の選択された試験項目の番号 (試験 1:1, 試験 2:2)		10 進表記	[SelectedTestMethod = 1]
		BitLengthOfDKM	DKM のビット長	$KBits$	10 進表記	[BitLengthOfDKM = 320]
		NumberOfDKM	DKM の個数		10 進表記	[NumberOfDKM = 2048]
		RatioOfInvalidData	Unilateral key confirmation from party V to party U の選択時, 鍵確立に失敗するデータの割合. それ以外は省略.		浮動小数点表記	[RatioOfInvalidData = 0.5]
		MACforKeyConfirmation	Unilateral key confirmation from party V to party U の選択時, Key confirmation に使う MAC アルゴリズム. それ以外は省略.		文字列	[MACforKeyConfirmation = HMAC_SHA512]
		BitLengthOfMacKey	Unilateral key confirmation from party V to party U の選択時, Key confirmation に使う MacKey のビット長. それ以外は省略.		10 進表記	[BitLengthOfMacKey = 128]
		BitLengthOfMacTag	Unilateral key confirmation from party V to party U の選択時, Key confirmation に使う MacTag のビット長. それ以外は省略.		10 進表記	[BitLengthOfMacTag = 512]
	Party U 試験 1 ヘッダ	nV	Party V の公開鍵の法 n	n	16 進表記	nV = b73c . . . 9215
		eV	Party V の公開鍵指数 e	e	16 進表記	eV = 010001
	Party U 試験 1 本体 *1	COUNT	0 以上 NumberOfDKM 未満の整数		10 進表記	COUNT = 0
		NonceV	Nonce V		16 進表記	NonceV = 6695 . . . fe09
		OI	OtherInfo	$OtherInfo$	16 進表記	OI = a1b2 . . . 0e21
		SaltForHMACbasedKDF	HMAC を用いる鍵導出関数の選択時, HMAC の鍵として用いられる salt. それ以外は省略.	$salt$	16 進表記	SaltForHMACbasedKDF = eb22 . . . aee8
		C	【出力】Party U が生成した暗号文	C	16 進表記	C = 12a1 . . . 2900
		DKM	【出力】Party U が計算した $DerivedKeyingMaterial$	$DerivedKeyingMaterial$ (DKM)	16 進表記	DKM = 4c61 . . . af35
	Party U 試験 2 ヘッダ	nV	Party V の公開鍵の法 n	n	16 進表記	nV = b73c . . . 9215
		eV	Party V の公開鍵指数 e	e	16 進表記	eV = 010001
	Party U 試験 2 本体 *2	COUNT	0 以上 NumberOfDKM 未満の整数		10 進表記	COUNT = 0
		NonceV	Nonce V		16 進表記	NonceV = 6695 . . . fe09
		OI	OtherInfo	$OtherInfo$	16 進表記	OI = a1b2 . . . 0e21
		SaltForHMACbasedKDF	HMAC を用いる鍵導出関数の選択時, HMAC の鍵として用いられる salt. それ以外は省略.	$salt$	16 進表記	SaltForHMACbasedKDF = eb22 . . . aee8
		C	暗号文	C	16 進表記	C = 1d10 . . . de15
		MacDataV	MacData	$MacData$	16 進表記	MacDataV = 4b43 . . . de15
		MacTagV	MacTag	$MacTag_V$	16 進表記	MacTagV = 9d6e . . . ae0d
		KeyData	【出力】Party U が計算した $KeyData$	$KeyData$	16 進表記	KeyData = b0a7 . . . c0a8
		Result	【出力】鍵確立の成功又は失敗 (成功:P, 失敗:F)		文字列	Result = P
	Party V 試験 1 ヘッダ	nV	Party V の公開鍵の法 n	n	16 進表記	nV = abf7 . . . b8e3
		eV	Party V の公開鍵指数 e	e	16 進表記	eV = 010001
		dV	Party V のプライベート鍵指数 d	d	16 進表記	dV = 53a0 . . . 77c1
	Party V 試験 1 本体 *3	COUNT	0 以上 NumberOfDKM 未満の整数		10 進表記	COUNT = 0
		NonceV	Nonce V		16 進表記	NonceV = 6695 . . . fe09
		OI	OtherInfo	$OtherInfo$	16 進表記	OI = a1b2 . . . 0e21
		SaltForHMACbasedKDF	HMAC を用いる鍵導出関数の選択時, HMAC の鍵として用いられる salt. それ以外は省略.	$salt$	16 進表記	SaltForHMACbasedKDF = eb22 . . . aee8
		C	暗号文	C	16 進表記	C = abf7 . . . 52ba
		DKM	【出力】Party V が計算した $DerivedKeyingMaterial$	$DerivedKeyingMaterial$ (DKM)	16 進表記	DKM = 7bad . . . 4a45
		Result	【出力】鍵確立の成功又は失敗 (成功:P, 失敗:F)		文字列	Result = P
	Party V 試験 2 ヘッダ	nV	Party V の公開鍵の法 n	n	16 進表記	nV = abf7 . . . b8e3
		eV	Party V の公開鍵指数 e	e	16 進表記	eV = 010001
		dV	Party V のプライベート鍵指数 d	d	16 進表記	dV = 53a0 . . . 77c1
	Party V 試験 2 本体 *4	COUNT	0 以上 NumberOfDKM 未満の整数		10 進表記	COUNT = 0
		NonceV	Nonce V		16 進表記	NonceV = 6695 . . . fe09
		OI	OtherInfo	$OtherInfo$	16 進表記	OI = a1b2 . . . 0e21
		SaltForHMACbasedKDF	HMAC を用いる鍵導出関数の選択時, HMAC の鍵として用いられる salt. それ以外は省略.	$salt$	16 進表記	SaltForHMACbasedKDF = eb22 . . . aee8
		C	暗号文	C	16 進表記	C = abf7 . . . 52ba
		MacDataV	MacData	$MacData$	16 進表記	MacDataV = 4b43 . . . de15
		MacTagV	【出力】Party V が計算した MacTag	$MacTag_V$	16 進表記	MacTagV = c447 . . . fcab
		KeyData	【出力】Party V が計算した $KeyData$	$KeyData$	16 進表記	KeyData = c447 . . . fcab

*1 NumberOfDKM 個の各データの組を以下のように記述する.

```
COUNT = 0                          ♯ i = 0 のデータの組について記述する.
NonceV = 6695 ... fe09             ♯ i = 0 に対応する Nonce V を記述する.
OI = a1b2 ... 0e21                 ♯ i = 0 に対応する OtherInfo を記述する.
SaltForHMACbasedKDF = eb22 ... aee8  ♯ HMAC を用いる鍵導出関数の選択時, HMAC の鍵として用いられる salt を記述する. それ以外は省略.
C = abf7 ... 52ba                  ♯ i = 0 に対応して, IUT が生成した C.
DKM = 4c61 ... af35                ♯ i = 0 に対応して, IUT が導出した DerivedKeyingMaterial (DKM).

COUNT = 1                          ♯ i = 1 のデータの組について記述する.
NonceV = 6695 ... fe09             ♯ i = 1 に対応する Nonce V を記述する.
OI = a1b2 ... 0e21                 ♯ i = 1 に対応する OtherInfo を記述する.
SaltForHMACbasedKDF = eb22 ... aee8  ♯ HMAC を用いる鍵導出関数の選択時, HMAC の鍵として用いられる salt を記述する. それ以外は省略.
C = abf7 ... 52ba                  ♯ i = 1 に対応して, IUT が生成した C.
DKM = 4c61 ... af35                ♯ i = 1 に対応して, IUT が導出した DerivedKeyingMaterial (DKM).
⋮
COUNT = ⟨NumberOfDKM − 1⟩          ♯ i = ⟨NumberOfDKM − 1⟩ のデータの組について記述する.
NonceV = 6695 ... fe09             ♯ i = ⟨NumberOfDKM − 1⟩ に対応する Nonce V を記述する.
OI = a1b2 ... 0e21                 ♯ i = ⟨NumberOfDKM − 1⟩ に対応する OtherInfo を記述する.
SaltForHMACbasedKDF = eb22 ... aee8  ♯ HMAC を用いる鍵導出関数の選択時, HMAC の鍵として用いられる salt を記述する. それ以外は省略.
C = abf7 ... 52ba                  ♯ i = ⟨NumberOfDKM − 1⟩ に対応して, IUT が生成した C.
DKM = 4c61 ... af35                ♯ i = ⟨NumberOfDKM − 1⟩ に対応して, IUT が導出した DerivedKeyingMaterial (DKM).
```

*2 NumberOfDKM 個の各データの組を以下のように記述する.

```
COUNT = 0                              ♯ i = 0 のデータの組について記述する.
NonceV = 6695 ... fe09                 ♯ i = 0 に対応する Nonce V を記述する.
OI = a1b2 ... 0e21                     ♯ i = 0 に対応する OtherInfo を記述する.
SaltForHMACbasedKDF = eb22 ... aee8    ♯ HMAC を用いる鍵導出関数の選択時, HMAC の鍵として用いられる salt を記述する. それ以外は省略.
C = 1d10 ... de15                      ♯ i = 0 に対応する暗号文 C を記述する.
MacDataV = 457f ... e64b               ♯ i = 0 に対応する, MacTag_V の生成に用いた MacData を記述する.
MacTagV = 9d6e ... ae0d                ♯ i = 0 に対応する, MacTag_V を記述する.
KeyData = b0a7 ... c0a8                ♯ i = 0 に対応して, IUT が生成した KeyData.
Result = P                             ♯ i = 0 に対応して, IUT が計算した鍵確立の成功又は失敗.

COUNT = 1                              ♯ i = 1 のデータの組について記述する.
NonceV = 6695 ... fe09                 ♯ i = 1 に対応する Nonce V を記述する.
OI = a1b2 ... 0e21                     ♯ i = 1 に対応する OtherInfo を記述する.
SaltForHMACbasedKDF = eb22 ... aee8    ♯ HMAC を用いる鍵導出関数の選択時, HMAC の鍵として用いられる salt を記述する. それ以外は省略.
C = 1d10 ... de15                      ♯ i = 1 に対応する暗号文 C を記述する.
MacDataV = 457f ... e64b               ♯ i = 1 に対応する, MacTag_V の生成に用いた MacData を記述する.
MacTagV = 9d6e ... ae0d                ♯ i = 1 に対応する, MacTag_V を記述する.
KeyData = b0a7 ... c0a8                ♯ i = 1 に対応して, IUT が生成した KeyData.
Result = P                             ♯ i = 1 に対応して, IUT が計算した鍵確立の成功又は失敗.
⋮
COUNT = ⟨NumberOfDKM − 1⟩              ♯ i = ⟨NumberOfDKM − 1⟩ のデータの組について記述する.
NonceV = 6695 ... fe09                 ♯ i = ⟨NumberOfDKM − 1⟩ に対応する Nonce V を記述する.
OI = a1b2 ... 0e21                     ♯ i = ⟨NumberOfDKM − 1⟩ に対応する OtherInfo を記述する.
SaltForHMACbasedKDF = eb22 ... aee8    ♯ HMAC を用いる鍵導出関数の選択時, HMAC の鍵として用いられる salt を記述する. それ以外は省略.
C = 1d10 ... de15                      ♯ i = ⟨NumberOfDKM − 1⟩ に対応する暗号文 C を記述する.
MacDataV = 457f ... e64b               ♯ i = ⟨NumberOfDKM − 1⟩ に対応する, MacTag_V の生成に用いた MacData を記述する.
MacTagV = 9d6e ... ae0d                ♯ i = ⟨NumberOfDKM − 1⟩ に対応する, MacTag_V を記述する.
KeyData = b0a7 ... c0a8                ♯ i = ⟨NumberOfDKM − 1⟩ に対応して, IUT が生成した KeyData.
Result = P                             ♯ i = ⟨NumberOfDKM − 1⟩ に対応して, IUT が計算した鍵確立の成功又は失敗.
```

∗3 NumberOfDKM 個の各データの組を以下のように記述する.

```
COUNT = 0                              ♯ i = 0 のデータの組について記述する.
NonceV = 6695 ... fe09                 ♯ i = 0 に対応する Nonce V を記述する.
OI = a1b2 ... 0e21                     ♯ i = 0 に対応する OtherInfo を記述する.
SaltForHMACbasedKDF = eb22 ... aee8    ♯ HMAC を用いる鍵導出関数の選択時, HMAC の鍵として用いられる salt を記述する. それ以外は省略.
C = abf7 ... 52ba                      ♯ i = 0 に対応する暗号文 C を記述する.
DKM = 9011 ... 71f8                    ♯ i = 0 に対応して, IUT が導出した DerivedKeyingMaterial (DKM).
Result = P                             ♯ i = 0 に対応して, IUT が計算した鍵確立の成功又は失敗.

COUNT = 1                              ♯ i = 1 のデータの組について記述する.
NonceV = 6695 ... fe09                 ♯ i = 1 に対応する Nonce V を記述する.
OI = a1b2 ... 0e21                     ♯ i = 1 に対応する OtherInfo を記述する.
SaltForHMACbasedKDF = eb22 ... aee8    ♯ HMAC を用いる鍵導出関数の選択時, HMAC の鍵として用いられる salt を記述する. それ以外は省略.
C = abf7 ... 52ba                      ♯ i = 1 に対応する暗号文 C を記述する.
DKM = 9011 ... 71f8                    ♯ i = 1 に対応して, IUT が導出した DerivedKeyingMaterial (DKM).
Result = P                             ♯ i = 1 に対応して, IUT が計算した鍵確立の成功又は失敗.
⋮
COUNT = ⟨NumberOfDKM − 1⟩              ♯ i = ⟨NumberOfDKM − 1⟩ のデータの組について記述する.
NonceV = 6695 ... fe09                 ♯ i = ⟨NumberOfDKM − 1⟩ に対応する Nonce V を記述する.
OI = a1b2 ... 0e21                     ♯ i = ⟨NumberOfDKM − 1⟩ に対応する OtherInfo を記述する.
SaltForHMACbasedKDF = eb22 ... aee8    ♯ HMAC を用いる鍵導出関数の選択時, HMAC の鍵として用いられる salt を記述する. それ以外は省略.
C = abf7 ... 52ba                      ♯ i = ⟨NumberOfDKM − 1⟩ に対応する暗号文 C を記述する.
DKM = 9011 ... 71f8                    ♯ i = ⟨NumberOfDKM − 1⟩ に対応して, IUT が導出した DerivedKeyingMaterial (DKM).
Result = P                             ♯ i = ⟨NumberOfDKM − 1⟩ に対応して, IUT が計算した鍵確立の成功又は失敗.
```

∗4 NumberOfDKM 個の各データの組を以下のように記述する.

```
COUNT = 0                              ♯ i = 0 のデータの組について記述する.
NonceV = 6695 ... fe09                 ♯ i = 0 に対応する Nonce V を記述する.
OI = a1b2 ... 0e21                     ♯ i = 0 に対応する OtherInfo を記述する.
SaltForHMACbasedKDF = eb22 ... aee8    ♯ HMAC を用いる鍵導出関数の選択時, HMAC の鍵として用いられる salt を記述する. それ以外は省略.
C = abf7 ... 52ba                      ♯ i = 0 に対応する暗号文 C を記述する.
MacDataV = 457f ... e64b               ♯ i = 0 に対応する, MacTag_V の生成に用いる MacData を記述する.
MacTagV = e98a ... d160                ♯ i = 0 に対応して, IUT が計算した MacTag_V.
KeyData = 985a ... 6ea2                ♯ i = 0 に対応して, IUT が導出した KeyData.

COUNT = 1                              ♯ i = 1 のデータの組について記述する.
NonceV = 6695 ... fe09                 ♯ i = 1 に対応する Nonce V を記述する.
OI = a1b2 ... 0e21                     ♯ i = 1 に対応する OtherInfo を記述する.
SaltForHMACbasedKDF = eb22 ... aee8    ♯ HMAC を用いる鍵導出関数の選択時, HMAC の鍵として用いられる salt を記述する. それ以外は省略.
C = abf7 ... 52ba                      ♯ i = 1 に対応する暗号文 C を記述する.
MacDataV = 457f ... e64b               ♯ i = 1 に対応する, MacTag_V の生成に用いる MacData を記述する.
MacTagV = e98a ... d160                ♯ i = 1 に対応して, IUT が計算した MacTag_V.
KeyData = 985a ... 6ea2                ♯ i = 1 に対応して, IUT が導出した KeyData.
⋮
COUNT = ⟨NumberOfDKM − 1⟩              ♯ i = ⟨NumberOfDKM − 1⟩ のデータの組について記述する.
NonceV = 6695 ... fe09                 ♯ i = ⟨NumberOfDKM − 1⟩ に対応する Nonce V を記述する.
OI = a1b2 ... 0e21                     ♯ i = ⟨NumberOfDKM − 1⟩ に対応する OtherInfo を記述する.
SaltForHMACbasedKDF = eb22 ... aee8    ♯ HMAC を用いる鍵導出関数の選択時, HMAC の鍵として用いられる salt を記述する. それ以外は省略.
C = abf7 ... 52ba                      ♯ i = ⟨NumberOfDKM − 1⟩ に対応する暗号文 C を記述する.
MacDataV = 457f ... e64b               ♯ i = ⟨NumberOfDKM − 1⟩ に対応する, MacTag_V の生成に用いる MacData を記述する.
MacTagV = e98a ... d160                ♯ i = ⟨NumberOfDKM − 1⟩ に対応して, IUT が計算した MacTag_V.
KeyData = 985a ... 6ea2                ♯ i = ⟨NumberOfDKM − 1⟩ に対応して, IUT が導出した KeyData.
```

2.1.5　結果ファイル (*.out)

表8: NIST SP800-56B に記載された KAS1 結果ファイル

機能	分類	タグ	内容	値の表記	例示
鍵共有	ヘッダ	AlgorithmName	KAS1_in_NIST_SP800_56B	文字列	[AlgorithmName = KAS1_in_NIST_SP800_56B]
		TargetFunction	KeyAgreement	文字列	[TargetFunction = KeyAgreement]
		TargetRole	IUT が担う役割 (Party_U, Party_V)	文字列	[TargetRole = Party_U]
		TypeOfPublicKey	Party V の公開鍵指数の種別 (65537:TYPE1, random:TYPE2)	文字列	[TypeOfPublicKey = TYPE1]
		TypeOfPrivateKey	Party V のプライベート鍵の種別 (CRT なし:TYPE2)	文字列	[TypeOfPrivateKey = TYPE2]
		BitLengthOfModulusPartyV	Party V の公開鍵の法 n のビット長	10 進表記	[BitLengthOfModulusPartyV = 3072]
		KDF	鍵導出関数識別子	文字列	[KDF = SP800_56B_5_5_1_KDFConcat_SHA1]
		BitLengthOfSaltForHMACbasedKDF	HMAC ベースの KDF を使用する場合, salt のビット長	10 進表記	[BitLengthOfSaltForHMACbasedKDF = 0]
		BitLengthOfOI	OtherInfo のビット長	10 進表記	[BitLengthOfOI = 384]
		BitLengthOfNonceV	Nonce_V のビット長のビット長	10 進表記	[BitLengthOfNonceV = 256]
		ID_U	Party U の Identifier	16 進表記	[ID_U = a1b2c3d4e5]
		ID_V	Party V の Identifier	16 進表記	[ID_V = 4a434154546964]
		KeyConfirmationSupported	サポートする Key Confirmation のタイプ (Key Confirmation なし:NoKC, unilateral key confirmation from party V to party U:Unilateral_V_to_U)	文字列	[KeyConfirmationSupported = NoKC]
		SelectedTestMethod	暗号アルゴリズム実装試験仕様書—鍵確立手法—の選択された試験項目の番号 (試験 1:1, 試験 2:2)	10 進表記	[SelectedTestMethod = 1]
		BitLengthOfDKM	DKM のビット長	10 進表記	[BitLengthOfDKM = 320]
		NumberOfDKM	DKM の個数	10 進表記	[NumberOfDKM = 2048]
		RatioOfInvalidData	Unilateral key confirmation from party V to party U の選択時, 鍵確立に失敗するデータの割合. それ以外は省略.	浮動小数点表記	[RatioOfInvalidData = 0.5]
		MACforKeyConfirmation	Unilateral key confirmation from party V to party U の選択時, Key confirmation に使う MAC アルゴリズム. それ以外は省略.	文字列	[MACforKeyConfirmation = HMAC_SHA512]
		BitLengthOfMacKey	Unilateral key confirmation from party V to party U の選択時, Key confirmation に使う MacKey のビット長. それ以外は省略.	10 進表記	[BitLengthOfMacKey = 128]
		BitLengthOfMacTag	Unilateral key confirmation from party V to party U の選択時, Key confirmation に使う MacTag のビット長. それ以外は省略.	10 進表記	[BitLengthOfMacTag = 512]
		〈 Results 〉	OK 又は NG	文字列	OK

注

- 試験合格の場合，〈 Results 〉に OK と表示される．
- 試験不合格の場合，〈 Results 〉に何らかの形式で NG と表示される．また，〈 Results 〉には，レスポンスファイル内の不合格となったデータが記述されている何番目 (COUNT，# 等の記号で番号を表す) のデータが不合格となったかが表示される．不合格となったデータが記述されているタグ名は，前記のレスポンスファイル仕様に【出力】と記述したタグである．ただし，【出力】と記述したタグが 1 つしかない場合，タグ名は省略することがある．