統合型セキュリティ通信規格書

Integrated Secure Communication Layer Protocols

(最終 Draft)

V1.00

1998.8.15

MEDIS-DC

目次

図表目次

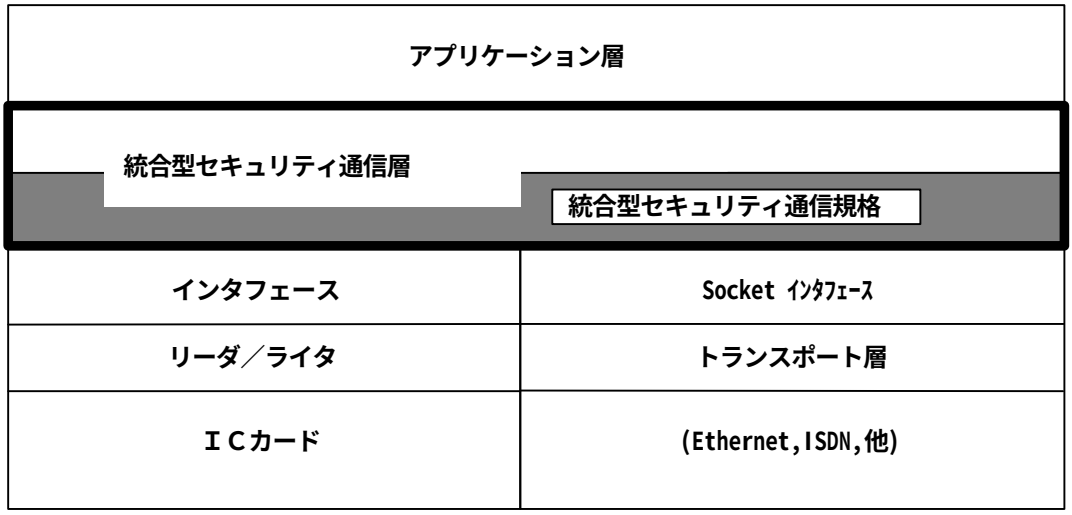
1.適用範囲

本規格は地域医療連携システムでの病院相互間の通信あるいは、院内のオンライン電子保存を行う際にトランスポート層の上位階層にセキュアな通信層を確保するために主に秘密鍵を使用する場合の通信規約を規定する。

具体的にはトランスポート層とセキュア層間の通信規約およびICカードをオペレーションカードとして用いる等のセキュアな鍵保管システムとの通信規約を規定する。

図 1にその位置づけを示す。

図 1 統合型セキュリティ通信規格の位置づけ



2.引用規格

ISO 7816−4: Indentification cards−Integrated circuit(s) with contacts Part 4:Interindustory commands for interchange

JIS X 6306: 外部端子付きICカード−共通コマンド

3.用語の定義

（1）統合型セキュリティ通信層（integrated secure communication layer） 統合型セキュリティ通信規格に基づきアプリケーション層からの指示により、セキュアな通信を行うことのできるプロトコル層。略称をISCLと言う。

（2）チャレンジコード(challenge code) ICカードにGET CHALLENGE コマンドを送った場合に得られるコード。通信相手を認証する場合の種として相手機器に送信する。相手機器はこれを内部認証キーにより暗号化して返送してくるので、正しく暗号化されていれば、正しい通信相手として認証する。

（3）レスポンスコード（response code） チャレンジコードを相手機器に送ったときに相手機器が返送してくる認証コード。

（4）相互認証（authentication） 相手が通信目的の正式な相手か相互確認すること。

(5) 通信ブロックサイズ (communication block size) ISCL 層と下位層との間で交換するデータ単位の最大長をいう。回線接続プロトコルで設定される。メッセージブロック・サイズはこのサイズ内で送られる。下位レベルへ設定するブロックサイズの最大値は通信ブロックサイズに32バイトを加えたものとする。

(6) メッセージ (message) 各セッションで送信すべきデータ全体をいう。

(7) 分割メッセージ (split message) メッセージを通信ブロックに分割して送る場合の単位をいう。

(8) メッセージブロック (message block) 通信プロトコルで交換されるデータ単位をいう。メッセージヘッダーとメッセージデータからなる。

(9) メッセージヘッダー (message header) メッセージブロックの前半部分を占め、メッセージブロックの目的を規定する。

(10) メッセージデータ (message data) メッセージブロックの後半部分を占め、メッセージブロックの目的に付帯する内容を送る。

4.基本機能

4.1.概要

通信におけるセキュリティに対する脅威には、「なりすまし」「改ざん」「盗聴」の3種類がある。「なりすまし」に対しては「相互認証機能」、「盗聴」に対しては「暗号化機能」、「改ざん」に対しては「メッセージ認証機能」により安全性を確保する。オンライン電子保存のように院内で盗聴のおそれが無い場合には暗号化機能を使用しないで通信を行うことも可能とする。

4.2. ICカードの認証機能

相互認証機能および暗号化機能としてISO 7816−4またはJIS X 6306で規定されている表1のICカードの認証用コマンドを使用する。ただし、これらをICカード以外で実現することも許容する。

表 1 ICカードの認証用コマンド

コマンド名	定義
GET CHALLENGE	チャレンジコードの出力を要求する。 EXTERNAL AUTHENTICATE の前に使用する。
INTERNAL AUTHENTICATE	接続装置から送られるチャレンジコード、及びカード内に格納されている内部認証キーを用いて、カードによるレスポンスコードの計算及び出力を要求する。 当該カードが正当であることを外部に認証させるために使用する。
EXTERNAL AUTHENTICATE	カードから出力されたチャレンジコード、及びカード内に格納されている外部認証キーを用いて、接続装置から送られるレスポンスコードの認証を行い、結果を出力することを要求する。 当該外部が正当であることをカードに認証させるために使用する。

4.3.相互認証機能

「なりすまし」対策として、相互認証を行なう。本規格では「チャレンジ‐レスポンス」方式で相互認証を行なう。手順を図2に示す。

⑴ 認証を必要とする相手に、“チャレンジコード”を与える。チャレンジコードはICカードに GET CHALLENGE コマンドを送ることにより得られる無作為な乱数である。

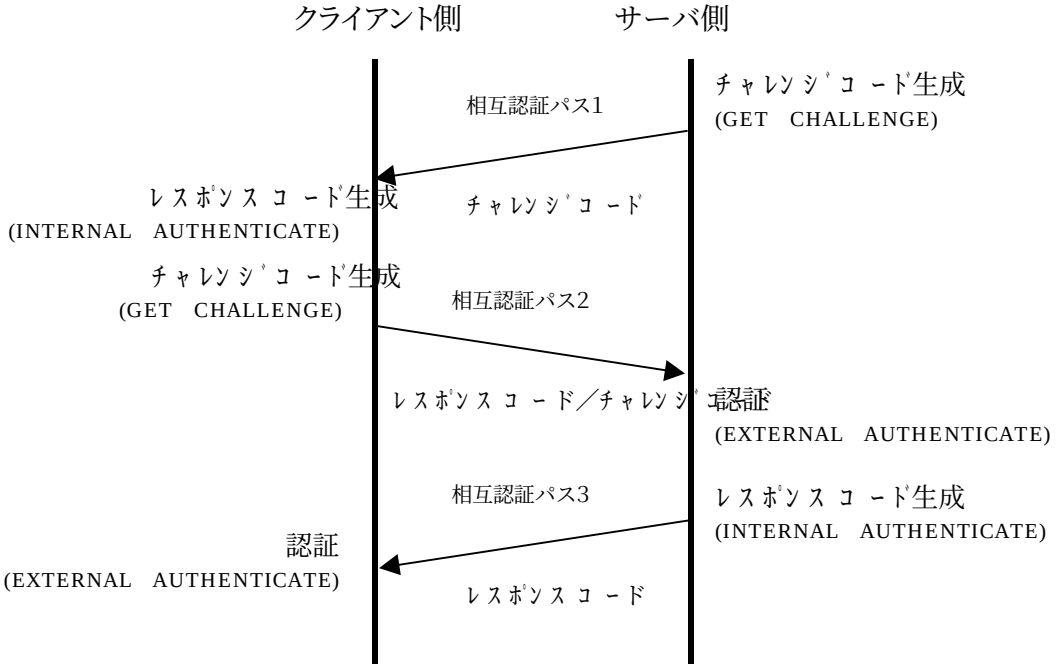
⑵ 相手はチャレンジコードに INTERNAL AUTHENTICATE コマンドにより所定の計算を施し、その結果の“ レスポンスコード”を返す。

⑶ 自分は同様に EXTERNAL AUTHENTICATE コマンドにより、チャレンジコードに所定の計算を施す。また相手から返されたレスポンスコードと、自分の計算結果を比較する。一致すれば（所定の計算方法を知っているという意味で）「正しい相手」と認証する。

⑷ ⑴ ～⑶ の手順を立場を変えて行なうことで、相互認証を行なう。

この相互認証方式を、「4ウェイ相互認証方式」という。本規格ではレスポンスコードと一緒に相手認証のためのチャレンジコードを送信する3パス4ウエイ相互認証方式を採用している。

図 2　3パス4ウエイ相互認証方式



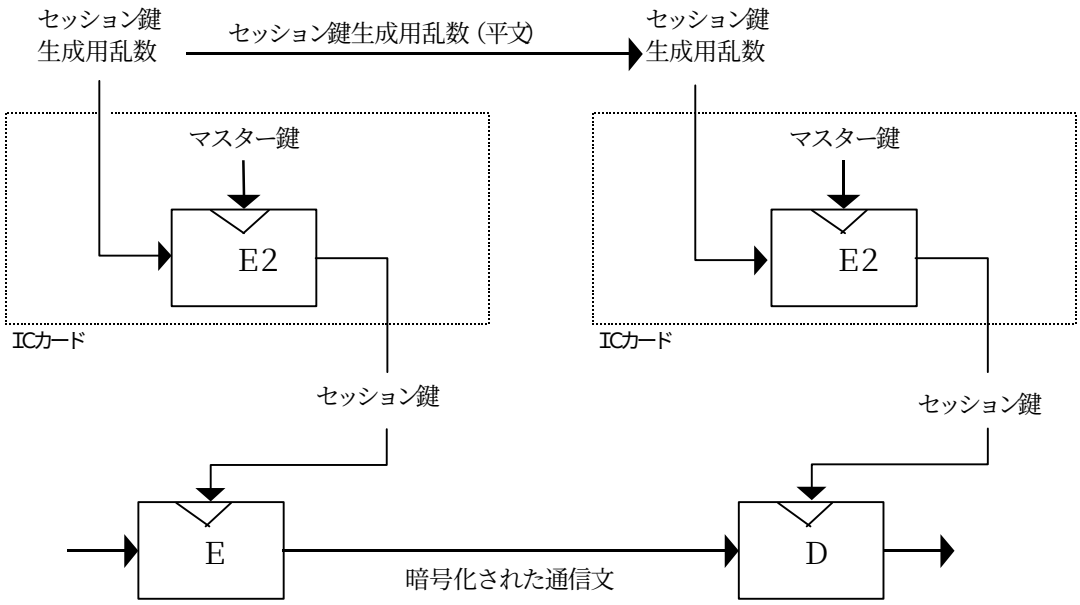
4.4.暗号化機能

「盗聴」対策（データ秘匿）として、暗号化を行なう。 暗号方式には秘密鍵方式、ブロック型暗号の DES（CBC 方式）を利用する。

常に同じ暗号鍵を使う場合は通信文を解読される危険がある。本規格では安全性を増すために、ある送受信単位（単位のとり方は任意）ごとに暗号鍵を変える（これをセッション鍵と呼ぶ）。

図4に示すように GET CHALLENGE コマンドにより生成したセッション鍵生成用乱数を送信し、各機器で INTERNAL AUTHENTICATE コマンドにより、得られたレスポンスコードをセッション鍵とする。これにより通信路上でセッション鍵を盗聴される危険性を低下させることができる。

図 3 セッション鍵の生成と暗号化

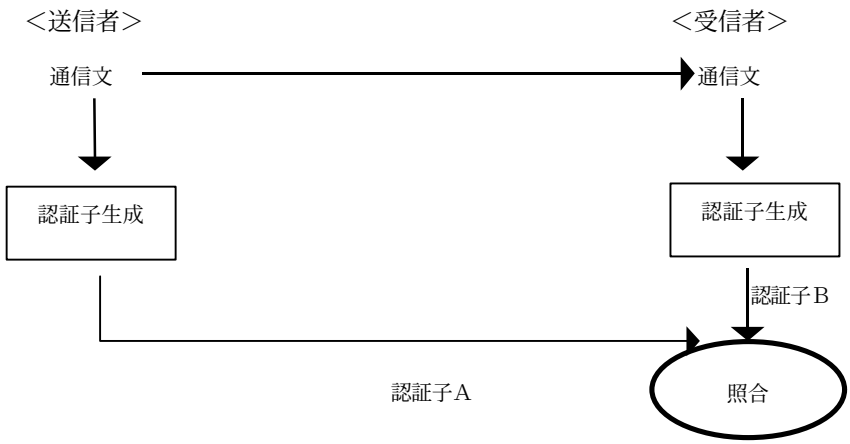


E：暗号器　　E2：暗号器２（ＩＣカードの INTERNAL　AUTHENTICATE コマンド）
D：復号器

4.5.メッセージ認証機能

「改ざん」を検知するためにメッセージ認証を行なう。本規格ではハッシュ関数を用いて通信文から生成した「メッセージ認証子」を照合することで、メッセージ認証を実現する。図 4に概要を示す。

図 4 メッセージ認証子によるメッセージ認証機能



5.メッセージブロック

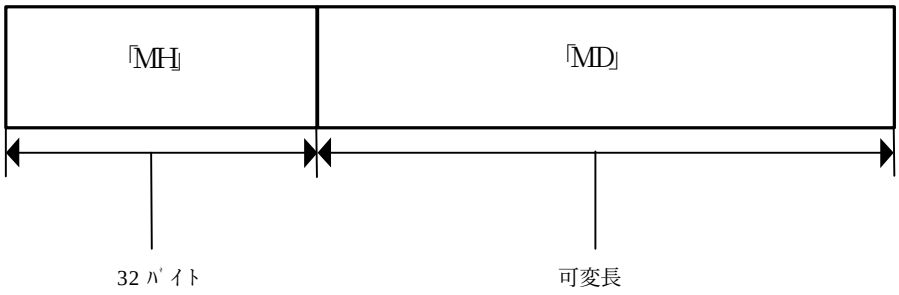
5.1.メッセージブロックの基本規約

5.1.1.メッセージブロックの構造

メッセージブロックは、メッセージブロックの目的を規定するメッセージヘッダと、その内容にあたるメッセージデータにより構成される。以降、メッセージヘッダを「MH」、メッセージデータを「MD」と記述する。

図 5にその構造を示す。

図 5 メッセージブロックの構造



MHは32バイト固定長であり、MDはMH内で長さを指示された可変長のデータである。MHの定義については、5. 2項に記述する。

5.1.2.メッセージブロック送受信の手順

本規格では、以下の手順に従ってメッセージブロックの送受信を行うものとする。

⑴ 送信側は送るべきMDに先立って必ずMHを送信する。
⑵ 受信側は一連の受信の初めに必ずMHが来るものと想定する。
⑶ MHのみでMDがない場合があるものとする。(MDのデータ長が 0)

5.2.MH

5.2.1.MHの構造

MHは本規格のプロトコルを規定する情報である。表 2にMHの構造を示す。

表 2 MHの構造

ﾊﾞｲﾄ位置	長さ	ﾌｨｰﾙﾄﾞ名	ﾌｨｰﾙﾄﾞ定義
1−4	4	indicator	指標（予約領域）
5−8	4	messageId	メッセージ識別子
9−12	4	dataLength	MD長
13−16	4	option	オプション
17−20	4	timeStamp	時刻（予約領域）
21−24	4	errno	結果応答 NAK の理由コード（予約領域）
25−32	8	stuff	MHを 32 バイトに揃えるための領域。（予約領域）

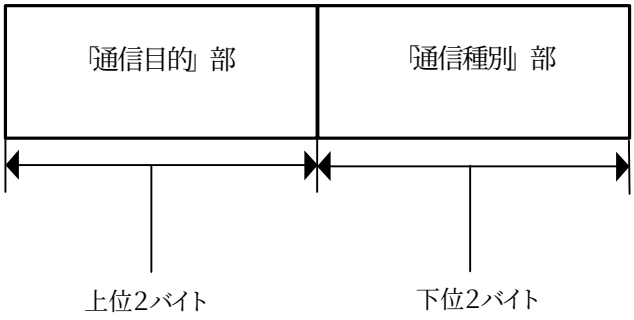
5.2.2.指標

MHの先頭に位置し、特定のビットパターンを持つことにより、MHの開始であることを示す。（V1. 00では使用しない。）

5.2.3.メッセージ識別子

メッセージ識別子はMDの種類を識別するものであり、「通信目的」部と「通信種別」部からなる。図 6に示すように、上位 2 バイトが「通信目的」部であり、下位 2 バイトが「通信種別」部である。

図 6 メッセージ識別子の構造



⑴ 「通信目的」部

表 3に「通信目的」部の一覧を示す。

表 3 「通信目的」部の一覧

通信目的	コード
回線接続確認	**００１１Ｈ**
相互認証	**００１２Ｈ**
相互認証パス１	**００１３Ｈ**
相互認証パス２	**００１４Ｈ**
相互認証パス３	**００１５Ｈ**
相互認証終了	**００１６Ｈ**
メッセージ送信	**００２０Ｈ**
セッション鍵生成用乱数送信	**００２１Ｈ**
メッセージ認証子送信	**００２３Ｈ**
スルーモード送信	**００２６Ｈ**
回線切断	**００ＦＦＨ**

⑵ 「通信種別」部

表 4に「通信種別」部の一覧を示す。

表 4 「通信種別」部の一覧

種別	コード
要求	**０００１Ｈ**
通知	**０００２Ｈ**
応答	**０００３Ｈ**

5.2.4.MD長

当該MHに後続するＭＤのデータ長を、4バイトの数値で示す。

5.2.5.オプション

当該メッセージブロックに付随するオプション情報を格納する。

⑴ 「要求」に対する結果応答

「通信種別」部が「要求」であるメッセージに対する「応答」結果を格納する。表 5に一覧を示す。

表 5 結果応答の一覧

種別	結果	コード	備考
結果応答	ACK	**００００００００H**	**肯定**
	NAK	**ＦＦＦＦＦＦＦＦH**	**否定**

⑵ 認証用コマンドの処理結果

認証用コマンドの処理結果を格納する。表 6に一覧を示す。

表 6 処理結果の一覧

種別	状態	コード
処理結果	**正常終了**	**００００００００H**
	異常終了	**正常終了コード以外の不定値**

5.2.6.時刻

送信した時刻を示す。(V1. 00では使用しない。)

5.2.7.理由コード

結果応答が NAK の場合の理由コードを示す。(V1．00では使用しない。)

5.2.8.スタッフ

MHを 32 バイトに揃えるために確保した領域を示す。(V1. 00では使用しない。)

5.3.MD

5.3.1.MD一覧

表 7に本規格のプロトコルで使用されるMDの一覧を示す。

表 7 MDの一覧

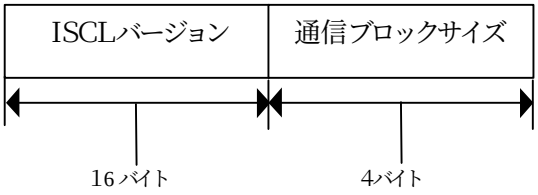
データ名	MD 長 (ﾊﾞｲﾄ)	備考
ISCLﾊﾞｰｼﾞｮﾝと通信ﾌﾟﾛｯｸｻｲｽﾞ	２０	回線接続で使用。(6.2 項参照)
相互認証方式ﾃﾞｰﾀ	不定	相互認証で使用。(6.3.2 項参照)
認証属性	不定	相互認証で使用。(6.3.2 項参照)
ﾁｬﾚﾝｼﾞｺｰﾄﾞ	８	GET CHALLENGE ｺﾏﾝﾄﾞにより生成。
ﾚｽﾎﾟﾝｽｺｰﾄﾞ	８	INTERNAL AUTHENTICATE ｺﾏﾝﾄﾞにより生成。
ﾚｽﾎﾟﾝｽｺｰﾄﾞとﾁｬﾚﾝｼﾞｺｰﾄﾞ	１６	ﾚｽﾎﾟﾝｽｺｰﾄﾞとﾁｬﾚﾝｼﾞｺｰﾄﾞをいっしょに送る場合。(6.3.4 項参照)
送信方式ﾃﾞｰﾀ	１６	上位ｱﾌﾟﾘから指定された送信方式により生成。
受信可能ﾒｯｾｰｼﾞ長	４	一度のﾒｯｾｰｼﾞ送受信ﾌﾟﾛﾄｺﾙで、送受信可能なﾒｯｾｰｼﾞ長。
ｾｯｼｮﾝ鍵生成用乱数	８	GET CHALLENGE ｺﾏﾝﾄﾞにより生成。
ﾒｯｾｰｼﾞ認証子	不定	ﾒｯｾｰｼﾞ認証子生成機能より生成。ｻｲｽﾞはｱﾙｺﾞﾘｽﾞﾑによる。
分割ﾒｯｾｰｼﾞ	不定	上位ｱﾌﾟﾘから引き渡されたﾒｯｾｰｼﾞを、回線接続ﾌﾟﾛﾄｺﾙで確定した通信ﾌﾞﾛｯｸｻｲｽﾞ以内に分割して生成。

5.3.2.ＭＤ詳細

⑴ ＩSCLバージョンと通信ブロックサイズ

ISCLバージョンと通信ブロックサイズを図 7のように構成する。本規格のＩSCLバージョンは“MEDIS-ISCL△V1.00” (ASCII コ ードﾞ、△はスペース) と記述することとする。プロトコルで一度に送信可能な通信ブロックサイズを指す。

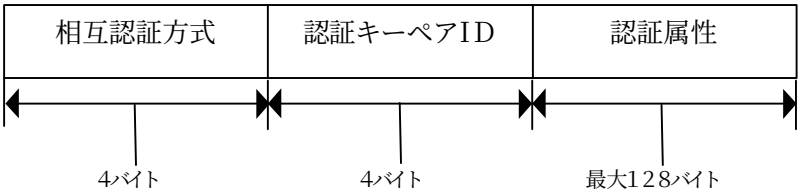
図 7 ＩSCLバージョンと通信ブロックサイズの構造



⑵ 相互認証方式データ

相互認証要求をする場合、受信側に相互認証方式データを送るものとする。図 8に相互認証方式データの構造を示す。

図 8 相互認証方式データの構造



各項目は以下の通りとする。

1) 相互認証方式

表 8に相互認証方式の一覧を示す。

表 8 相互認証方式の一覧

相互認証方式	コード	備考
４ウェイ相互認証方式	**００００００００Ｈ**	**デフォルト**

2) 認証キーペアＩＤ

表 9に相互認証に使用するキーペアＩＤ一覧を示す。

表 9 認証キーペアIDの一覧

認証キーペア名称	ID
認証キーペア１	**０００００００１Ｈ**
認証キーペア２	**０００００００２Ｈ**
認証キーペア３	**０００００００３Ｈ**
認証キーペア４	**０００００００４Ｈ**
認証キーペア５	**０００００００５Ｈ**
認証キーペア６	**０００００００６Ｈ**
認証キーペア７	**０００００００７Ｈ**
認証キーペア８	**０００００００８Ｈ**

3) 認証属性

認証のために使用する1～128バイトの属性データとする。その内容については、本規格では規定しない。

⑶ 認証属性

認証のために使用する1～128バイトの属性データとする。その内容については、本規格では規定しない。

⑷ チャレンジコード

GET CHALLENGE ｺﾏﾝﾄﾞにより生成される8バイトのコードを指す。

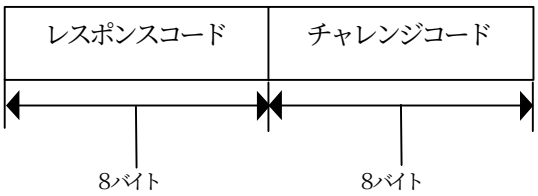
⑸ レスポンスコード

INTERNAL AUTHENTICATE ｺﾏﾝﾄﾞにより生成される8バイトのコードを指す。

⑹ レスポンスコードとチャレンジコード

GET CHALLENGE ｺﾏﾝﾄﾞにより生成れる8バイトのレスポンスコードとINTERNAL AUTHENTICATE ｺﾏﾝﾄﾞにより生成される8バイトのチャレンジコードを指す。図 9にその構造を示す。

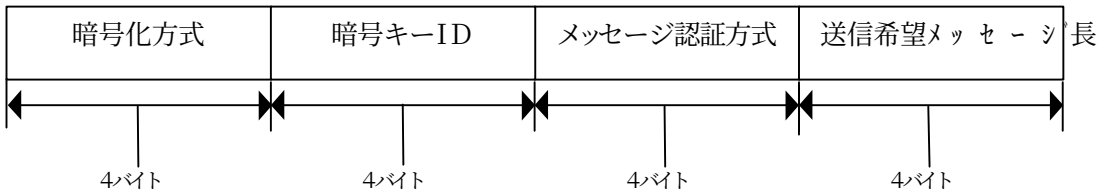
図 9 レスポンスコードとチャレンジコードの構造



⑺ 送信方式データ

送信要求をする場合、受信側に送信方式データを送るものとする。図 10に送信方式の構造を示す。

図 10 送信方式データの構造



各項目は以下の通りとする。

1) 暗号化方式

表 10に暗号方式の一覧を示す。

表 10 暗号化方式の一覧

暗号化方式	コード
暗号化なし	**００００００００H**
DES (CBC)	**００００１２１２H**

2) 暗号キーID

表 11に暗号化と復号化に使用するマスターキーIDに一覧を示す。

表 11 メッセージ認証方式の一覧

暗号キー名称	ID
暗号キー１	**００００００１H**
暗号キー２	**００００００２H**
暗号キー３	**００００００３H**
暗号キー４	**００００００４H**
暗号キー５	**００００００５H**
暗号キー６	**００００００６H**
暗号キー７	**００００００７H**
暗号キー８	**００００００８H**

3) メッセージ認証方式

表 12にメッセージ認証方式の一覧を示す。

表 12 メッセージ認証方式の一覧

メッセージ認証方式	コード
メッセージ認証なし	**００００００００H**
MD5	**００００１４４１H**
DESMAC	**００００４００１H**

4) 送信希望メッセージ長

送信側が一度に送信を希望するメッセージの長さとする。

⑧ 受信可能メッセージ長

一度のメッセージ送受信プロトコル (6.4 項参照) において、送信側が送信方式で要求した送信希望メッセージ長と、受信側が受信可能なメッセージ長のうち、小さい方を受信可能メッセージ長とする。4バイトで表わす。

⑨ セッション鍵生成用乱数

GET CHALLENGE ｺﾏﾝﾄﾞにより生成する8バイトのコードを指す。

⑩ メッセージ認証子

ﾒｯｾｰｼﾞ認証子生成機能により生成する。ｻｲｽﾞはﾒｯｾｰｼﾞ認証子生成ｱﾙｺﾞﾘｽﾞﾑによって変わる。

⑪ 分割メッセージ

上位アプリケーションから引き渡された送信メッセージを、回線接続プロトコルで確定した通信ブロックサイズ以内に分割して生成する。送信メッセージ長と通信ブロックサイズのうち、小さい方をMD長 (分割メッセージ長) とする。

5.4. メッセージブロック一覧

表 13にメッセージブロック一覧を示す。

表 13 メッセージブロック一覧

ﾒｯｾｰｼﾞ識別子		MH				MD	備考	
通信目的	種別	ﾒｯｾｰｼﾞ識別子(ｺｰﾄﾞ)	MD長(ﾊﾞｲﾄ)	ｵﾌﾟｼｮﾝ 種	ｵﾌﾟｼｮﾝ 内容		送信方向	その他
回線接続確認	要求	00110001H	20	―	―	ISCLﾊﾞｰｼﾞｮﾝ、通信ﾌﾞﾛｯｸｻｲｽﾞ	C←S	
回線接続確認	応答	00110003H	20/16	R	ACK/NAK	ISCLﾊﾞｰｼﾞｮﾝ、通信ﾌﾞﾛｯｸｻｲｽﾞ	C→S	NAKの時、MD長16とし、ISCLﾊﾞｰｼﾞｮﾝのみ
相互認証	要求	00120001H	不定	―	―	相互認証方式ﾃﾞｰﾀ	C←S	MD長は9～136バイト
相互認証	応答	00120003H	不定	R	ACK/NAK	認証属性	C→S	MD長は1～128バイト
相互認証ﾊﾟｽ1	通知	00130002H	8/0	S	GET CHALLENGE ｺﾏﾝﾄﾞ結果ｺｰﾄﾞ	ﾁｬﾚﾝｼﾞｺｰﾄﾞ	C←S	GET CHALLENGE ｺﾏﾝﾄﾞがｴﾗｰ時、MD長0
相互認証ﾊﾟｽ2	通知	00140002H	16/0	S	INTERNAL AUTHENTICATE/GET CHALLENGE ｺﾏﾝﾄﾞの結果ｺｰﾄﾞ	ﾚｽﾎﾟﾝｽｺｰﾄﾞ、ﾁｬﾚﾝｼﾞｺｰﾄﾞ	C→S	INTERNAL AUTHENTICATE又はGET CHALLENGE ｺﾏﾝﾄﾞがｴﾗｰの時、MD長0
相互認証ﾊﾟｽ3	通知	00150002H	8/0	S	EXTERNAL AUTHENTICATE/ INTERNAL AUTHENTICATE ｺﾏﾝﾄﾞ結果ｺｰﾄﾞ	ﾚｽﾎﾟﾝｽｺｰﾄﾞ	C←S	EXTERNAL AUTHENTICATE又はINTERNAL AUTHENTICATE ｺﾏﾝﾄﾞがｴﾗｰの時、MD長0
相互認証終了	通知	00160002H	0	S	EXTERNAL AUTHENTICATE ｺﾏﾝﾄﾞ結果ｺｰﾄﾞ	―	C→S	
ﾒｯｾｰｼﾞ送信	要求	00200001H	16	―	―	送信方式ﾃﾞｰﾀ	送→受	
ﾒｯｾｰｼﾞ送信	応答	00200003H	4/0	R	ACK/NAK	受信可能ﾃﾞｰﾀ長	送←受	NAKの時、MD長0
ｾｯｼｮﾝ鍵生成用乱数送信	要求	00210001H	8/0	S	GET CHALLENGE/INTERNAL AUTHENTICATE ｺﾏﾝﾄﾞの結果ｺｰﾄﾞ	ｾｯｼｮﾝ鍵生成用乱数	送→受	GET CHALLENGE又はINTERNAL AUTHENTICATE ｺﾏﾝﾄﾞがｴﾗｰ時、MD長0
ｾｯｼｮﾝ鍵生成用乱数送信	応答	00210003H	0	S	INTERNAL AUTHENTICATE ｺﾏﾝﾄﾞの結果ｺｰﾄﾞ	―	送←受	
ﾒｯｾｰｼﾞ送信	通知	00200002H	不定	―	―	分割ﾒｯｾｰｼﾞ	送→受	通信ﾌﾞﾛｯｸｻｲｽﾞ以内に分割
ﾒｯｾｰｼﾞ認証子送信	通知	00230002H	不定	―	―	ﾒｯｾｰｼﾞ認証子	送→受	ﾒｯｾｰｼﾞ認証子の長さは方式によって異なる
ｽﾙｰﾓｰﾄﾞ送信	通知	00260002H	不定	―	―	分割ﾒｯｾｰｼﾞ	送→受	通信ﾌﾞﾛｯｸｻｲｽﾞ以内に分割
回線切断	要求	00FF0001H	0	―	―	―	要→応	
回線切断	応答	00FF0003H	0	R	ACK/NAK	―	要←応	

注1) ｵﾌﾟｼｮﾝ R種結果応答、S処理結果　注2) 送信方向　C:クライアント側、Sサーバ側、送送信側、受受信側、要要求側、応応答側

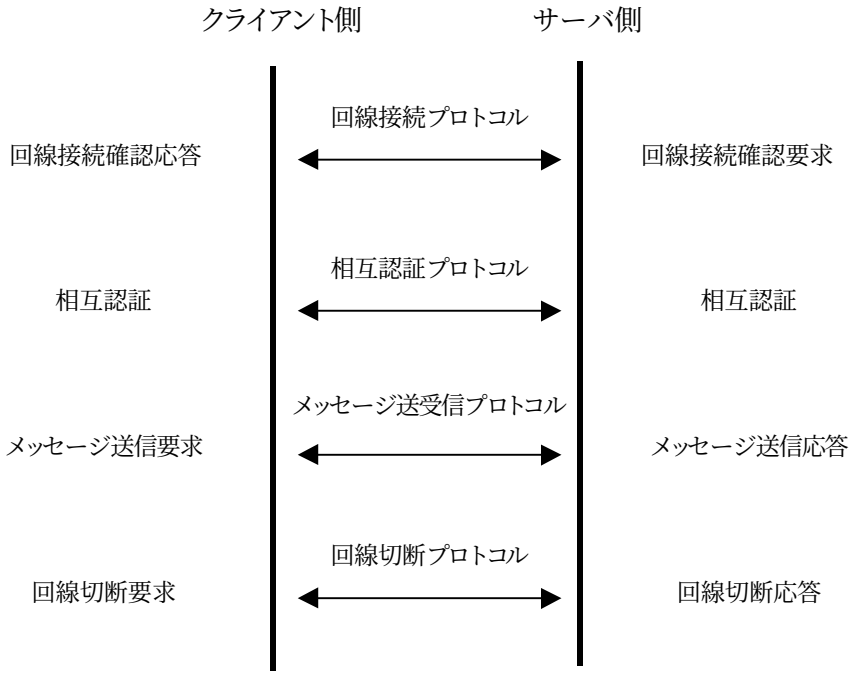
6.プロトコル

本規格におけるプロトコルは大別すると以下の4つになる。

⑴ 回線接続プロトコル
⑵ 相互認証プロトコル
⑶ メッセージ送受信プロトコル
⑷ 回線切断プロトコル

図 11にプロトコルの概要を示す。

図 11 プロトコルの概要



6.1.プロトコル記述方式

本記述で用いる用語を以下に定義する。

“MH”	MH全体を指す。
“MH.msgId”	MHの messageId フィールドを指す。
“MH.length”	MHの dataLength フィールドを指す。
“MH.opt”	MHの option フィールドを指す。

6.2.回線接続プロトコル

6.2.1.機能

回線接続プロトコルは、ＩSCLにおける最初のプロトコルである。このプロトコルでは、ＩSCLバージョン及び、通信ブロックサイズの確認を行う。

6.2.2.シーケンス

（1）クライアント側が接続を要求する。

（2）サーバ側が接続要求を受理または拒絶する。

（3）サーバ側が回線接続確認要求とＩSCLバージョン＋通信ブロックサイズを送信する。

「回線接続確認要求」MH

- MH.msgId=回線接続確認要求
- MH.length=20
- MH.opt=0

＊ＩSCLバージョンと通信ブロックサイズは以下の（16＋4バイト）のデータとする。

- ＩSCLバージョン（"MEDIS-ISCL△V1.00"）
- 通信ブロックサイズ（サーバ側が可能な）

（4）クライアント側が回線接続確認要求とＩSCLバージョン＋通信ブロックサイズを受信する。この時、以下の場合はエラーとする。

- MH.msgId が回線接続確認要求でない。（この場合は、回線接続確認応答を返さず、終了する。）
- ＩSCLバージョンが一致しない。

（5）サーバ通信ブロックサイズとクライアント通信ブロックサイズの小さい方を通信ブロックサイズとする。

（6）クライアント側が回線接続確認応答とＩSCLバージョン＋通信ブロックサイズを送信する。

「回線接続確認応答」MH

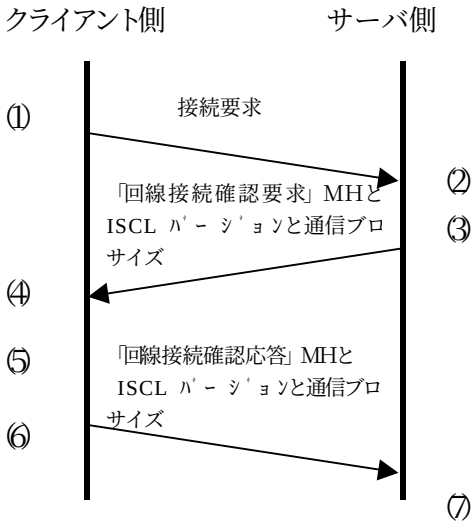
- MH.msgId=回線接続確認応答
- MH.length=20（NAK の場合 16）
- MH.opt=ACK。エラーの場合は NAK 。

＊ＩSCLバージョンと通信ブロックサイズは（3）の形式と同様とする。

（7）サーバ側が回線接続確認応答とＩSCLバージョン＋通信ブロックサイズを受信する。 以下の場合はエラー終了する。

- MH.msgId が回線接続確認応答でない。
- MH.opt が ACK でない。
- 通信ブロックサイズが通信ブロックサイズ（サーバ側が可能な）より大きい。

図 12 回線接続プロトコル

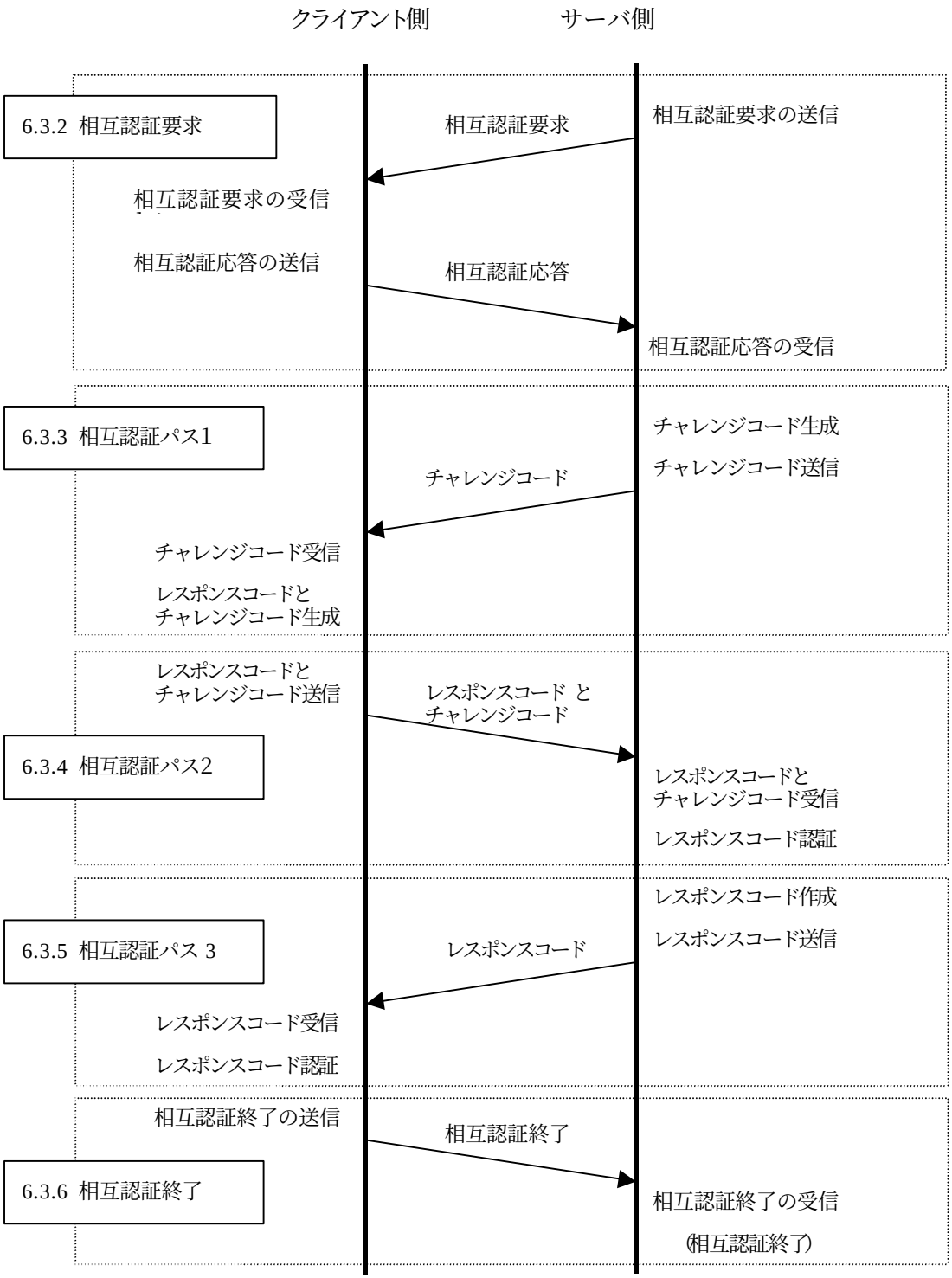


6.3.相互認証プロトコル

6.3.1.相互認証プロトコル概要

図 13に相互認証プロトコルの概要を示す。

図 13 相互認証プロトコル概要



6.3.2.相互認証要求プロトコル

6.3.2.1.機能

相互認証要求プロトコルは、相互認証プロトコルの最初のプロトコルである。このプロトコルでは、相互認証方式の確認を行う。

6.3.2.2.シーケンス

⑴ サーバ側が相互認証要求と相互認証方式データを送信する。

「相互認証要求」MH

- MH.msgId=相互認証要求
- MH.length=9～136
- MH.opt=0

「相互認証方式データ」は(4×2＋1～128)バイト

- 相互認証方式
- 認証キーペアＩＤ
- 認証属性

⑵ クライアント側が相互認証要求と相互認証方式データを受信する。 以下の場合はエラーとする。

- MH.msgId が相互認証要求でない。(この場合は、相互認証応答を返さず、終了する。)
- MH.opt の相互認証方式が一致しない。

⑶ クライアント側が相互認証応答と認証属性を送信する。

「相互認証応答」MH

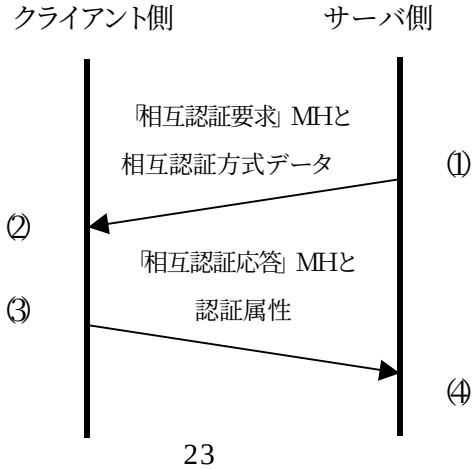
- MH.msgId=相互認証応答
- MH.length=1～128
- MH.opt=相互認証方式一致なら ACK。不一致なら NAK。

認証属性は1～128バイトのデータとする。

⑷ サーバ側が相互認証応答を受信する。 以下の場合はエラー終了する。

- MH.msgId が相互認証応答でない。
- MH.opt が ACK でない。

図 14 相互認証要求プロトコル



6.3.2.3.相互認証パス１プロトコル

6.3.2.4.機能

相互認証パス１プロトコルは、サーバ側が生成したチャレンジコードをクライアント側に送信し、クライアント側がレスポンスコードとチャレンジコードを生成するプロトコルである。

6.3.2.5.シーケンス

（1）サーバ側が GET CHALLENGEコマンドによりチャレンジコードを生成する。

② サーバ側が相互認証パス１通知とチャレンジコードを送信する。① がエラーの場合、送信後、終了する。

「相互認証パス１通知」MH
- MH.msgId=相互認証パス１通知
- MH.length= 8（GET CHALLENGE ｺﾏﾝﾄﾞがｴﾗｰの場合0）
- MH.opt= GET CHALLENGE ｺﾏﾝﾄﾞ結果ｺｰﾄﾞ

＊チャレンジコードは8バイトデータとする。

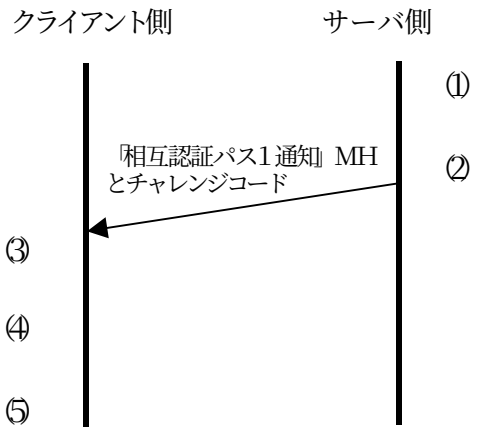
③ クライアント側が相互認証パス１通知とチャレンジコードを受信する。以下の場合はエラー終了する。

- MH.msgId が相互認証パス１通知でない。
- MH.opt がエラー(≠0)

（4）クライアント側が受信したチャレンジコードを使用し、INTERNAL AUTHENTICATE コマンドによりレスポンスコードを生成する。

⑤ クライアント側が GET CHALLENGE コマンドによりチャレンジコードを生成する。

図 15 相互認証パス1プロトコル



6.3.3.相互認証パス２プロトコル

6.3.3.1.機能

相互認証パス２プロトコルは、クライアント側が受信したチャレンジコードにより生成したレスポンスコードと生成したチャレンジコードをサーバ側に送信し、サーバ側が認証するプロトコルである。例外プロトコルは6．7．１項及び6．7．2項を参照。

6.3.3.2.シーケンス

⑴ クライアント側が相互認証パス２通知と生成したレスポンスコード＋チャレンジコードを送信する。

「相互認証パス２通知』MH

- MH.msgId=相互認証パス２通知
- MH.length= 16
- MH.opt= 0

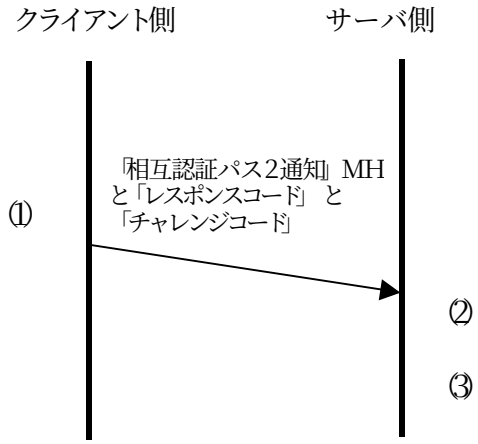
＊レスポンスコードとチャレンジコードは以下の（8バイト×2）のデータを指す。

- レスポンスコード
- チャレンジコード

⑵ サーバ側が相互認証パス２通知とレスポンスコード＋チャレンジコードを受信する。

⑶ サーバ側は EXTERNAL AUTHENTICATE コマンドにより、レスポンスコードの認証を行う。

図 16 相互認証パス2プロトコル



6.3.4.相互認証パス３プロトコル

6.3.4.1.機能

相互認証パス3プロトコルは、サーバ側が受信したチャレンジコードにより生成したレスポンスコードをクライアント側に送信し、クライアント側が認証するプロトコルである。例外プロトコルは6．7．3項及び6．7．4項を参照。

6.3.4.2.シーケンス

（1サーバ側が INTERNAL AUTHENTICATE コマンドにより、レスポンスコードを生成する。

⑵ サーバ側が相互認証パス 3 通知とレスポンスコードを送信する。

「相互認証パス3通知」MH

- MH.msgId=相互認証パス 3 通知
- MH.length= 8
- MH.opt= 0

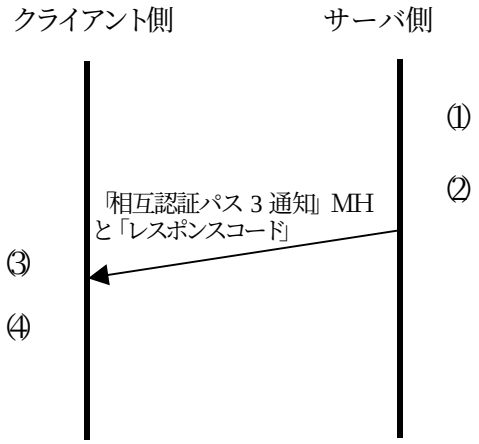
＊レスポンスコードは8バイトデータとする。

⑶ クライアント側が相互認証パス 3 通知とレスポンスコードを受信する。 以下の場合はエラー終了する。

- MH.msgId が相互認証パス 3 通知でない。
- MH.opt が 0 でない。

⑷ クライアント側が EXTERNAL AUTHENTICATE コマンドにより、レスポンスコードの認証を行う。

図 17 相互認証パス3プロトコル



6.3.5.相互認証終了プロトコル

6.3.5.1.機能

相互認証終了プロトコルは、クライアント側が相互認証パス3プロトコルの認証結果をサーバ側に送信し、相互認証を終了するプロトコルである。

6.3.5.2.シーケンス

⑴ クライアント側が相互認証終了通知を送信する。

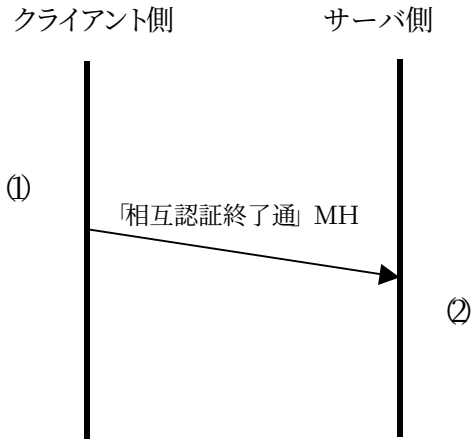
「相互認証終了通知」MH

- MH.msgId=相互認証終了通知
- MH.length=0
- MH.opt=認証結果（相互認証ﾊﾟｽ 3ﾌﾟﾛﾄｺﾙにおける EXTERNAL AUTHENTICATE ｺﾏﾝﾄﾞ結果ｺｰﾄﾞ）

⑵ サーバ側が相互認証終了通知を受信する。　以下の場合はエラー終了する。

- MH.msgId が相互認証終了通知でない。
- MH.opt が 0 でない。

図 18 相互認証終了プロトコル

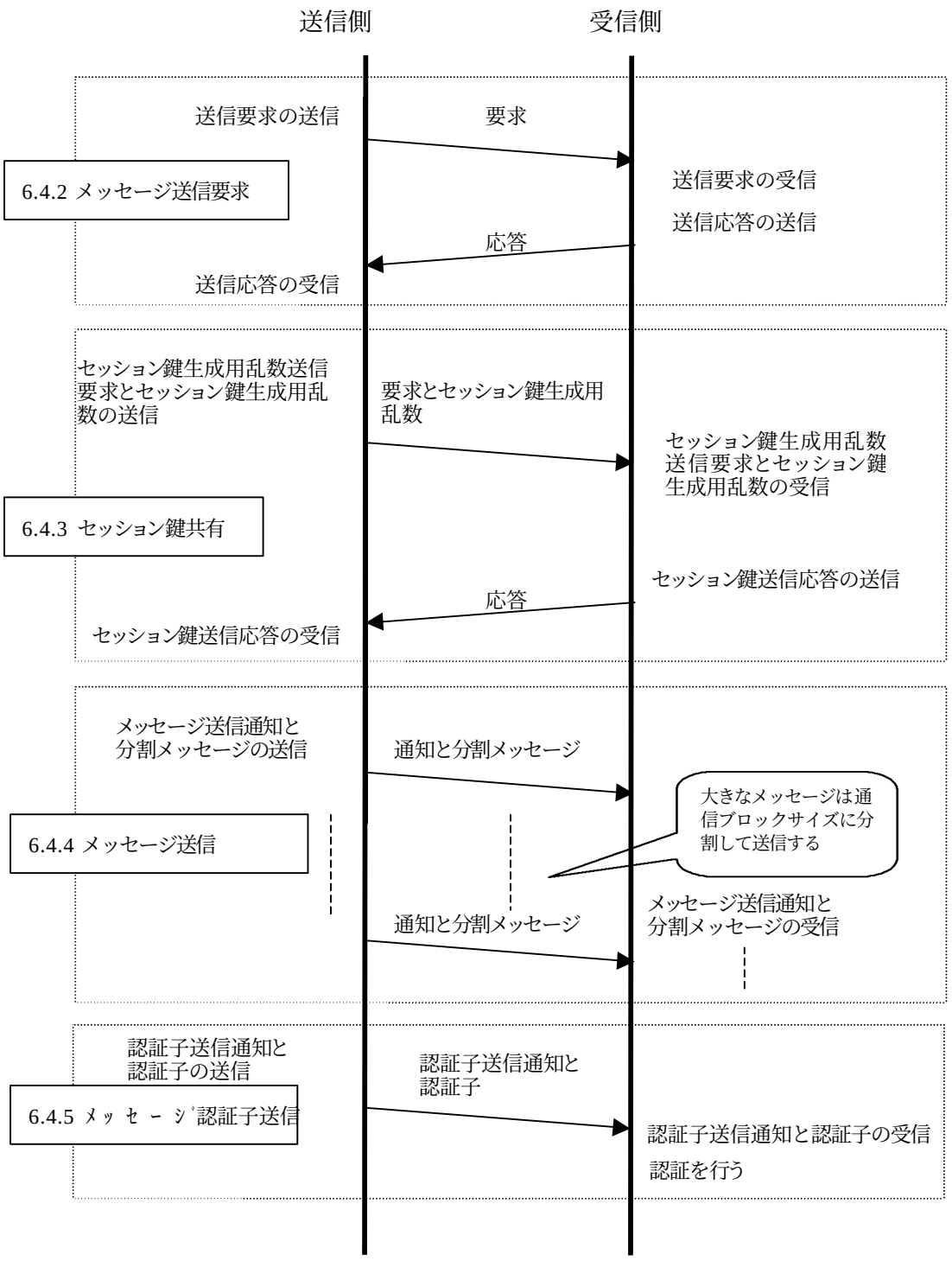


6.4.メッセージ送受信プロトコル

6.4.1.メッセージ送受信プロトコル概要

図 19にメッセージ送受信プロトコルの概要を示す。

図 19 メッセージ送受信プロトコル概要



6.4.2.メッセージ送信要求プロトコル

6.4.2.1.機能

メッセージ送信要求プロトコルは、送信側が送信方式とともに送信開始を要求し、受信側が受信可能長を応答するプロトコルである。例外プロトコルは6．7．5項を参照。

6.4.2.2.シーケンス

⑴ 送信側がメッセージ送信要求と送信方式データを作成する。

「メッセージ送信要求」MH

- MH.msgId=メッセージ送信要求
- MH.length=16
- MH.opt=0

「送信方式データ」は(4バイト×4)

- 暗号化方式
- 暗号キーＩＤ
- メッセージ認証方式
- 送信希望メッセージ長

⑵ 送信側がメッセージ送信要求と送信方式データを送信する。

⑶ 受信側がメッセージ送信要求と送信方式データを受信する。以下の場合はエラー終了する。

- MH.msgId がメッセージ送信要求でない。
- MH.length が 16 でない。

⑷ 受信側がメッセージ送信応答 MH と受信可能メッセージ長(4 バイト領域)を作成する。

「メッセージ送信応答」MH

- MH.msgId=メッセージ送信応答
- MH.length=4（NAK の場合 0）
- MH.opt=暗号方式及び MAC 方式が一致の場合 ACK
  不一致の場合 NAK

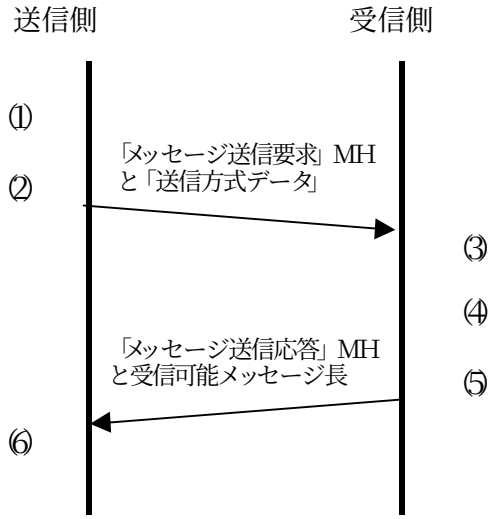
送信側のメッセージ長と受信側の受信可能メッセージ長のうち、小さい方を受信可能メッセージ長とする。

⑸ 受信側がメッセージ送信応答と受信可能メッセージ長を送信する。

⑹ 送信側がメッセージ送信応答と受信可能メッセージ長を受信する。以下の場合はエラー終了する。

- MH.msgId がメッセージ送信応答でない。
- MH.length が 4 でない。
- MH.opt が ACK でない。

図 20 メッセージ送信要求プロトコル



6.4.3.セッション鍵共有プロトコル

6.4.3.1.機能

セッション鍵共有プロトコルは、セッション鍵の共有が必要なときのみ発生するプロトコルである。例外プロトコルは6．7．6項を参照。

6.4.3.2.シーケンス

⑴ 送信側が GET CHALLENGE コマンドにより、セッション鍵生成用乱数を作成し、INTERNAL AUTHENTICATE コマンドにより、セッション鍵生成用乱数からセッション鍵を生成する。

⑵ 送信側がセッション鍵生成用乱数送信要求とセッション鍵生成用乱数を送信する。

「セッション鍵生成用乱数送信要求」MH
- MH.msgId=セッション鍵生成用乱数送信要求
- MH.length=8
- MH.opt= GET CHALLENG コマンド結果コードまたは INTERNAL AUTHENTICATE ｺﾏﾝﾄﾞ結果コード（正常時 0，　エラー時≠0)

⑶ 受信側がセッション鍵生成用乱数送信要求とセッション鍵生成用乱数を受信する。以下の場合はエラー終了する。

- MH.msgId がセッション鍵生成用乱数送信要求でない。
- MH.opt が 0 でない。

⑷ 受信側が INTERNAL AUTHENTICATE コマンドにより、受信したセッション鍵生成用乱数からセッション鍵を生成する。

⑸ 受信側がセッション鍵生成用乱数送信応答を送信する。

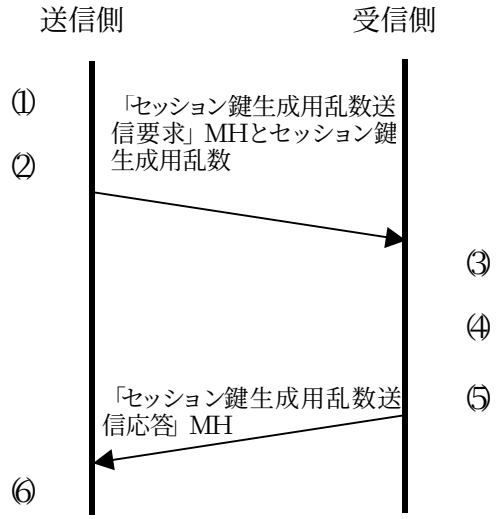
「セッション鍵生成用乱数送信応答」MH
- MH.msgId=セッション鍵生成用乱数送信応答
- MH.length=0
- MH.opt= INTERNAL AUTHENTICATE ｺﾏﾝﾄﾞ結果ｺｰﾄﾞ（正常時 0，　エラー時≠0)

＊セッション鍵生成時、エラーだった場合はここで終了する。

⑹ 送信側がセッション鍵生成用乱数送信応答を受信する。以下の場合はエラー終了する。

- MH.msgId がセッション鍵生成用乱数送信応答でない。
- MH.opt が 0 でない。

図 21 セッション鍵共有プロトコル



6.4.4.メッセージ送信プロトコル

6.4.4.1.機能

メッセージ送信プロトコルは、メッセージを6．２項の回線接続プロトコルでネゴシエーションした通信ブロックサイズに区切って、送信するプロトコルである。また、６．４．２項のメッセージ送信要求プロトコルにおいて、暗号化方式を指定してある場合は、メッセージ全体に暗号化を行う。

6.4.4.2.シーケンス

⑴ 送信側がメッセージ送信要求で受信側から与えられた受信可能メッセージ長（6.4.2 項参照）をメッセージ長とする。

⑵ 送信側が残メッセージ長と通信ブロックサイズの小さい方を分割メッセージ長とする。

⑶ 送信側が分割メッセージ長分のメッセージを切り出す。暗号化方式を指定してある場合は、当該分割メッセージに対して暗号化を行う。暗号化はメッセージ全体に渡るようにする。

⑷ 送信側がメッセージ送信通知と分割メッセージを送信する。

「メッセージ送信通知」MH

- MH.msgId=メッセージ送信通知
- MH.length=分割メッセージ長
- MH.opt=0

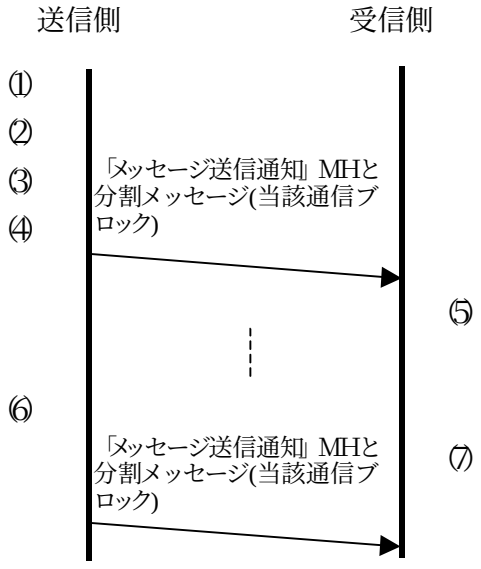
（５受信側がメッセージ送信通知と分割メッセージ（当該ブロック)を受信する。暗号方式を指定してある場合は、復号化を行う。以下の場合はエラーとし、終了する。

- MH.msgId がメッセージ送信通知でない。

⑹ 送信側が⑵～⑷を、全メッセージ分処理するまで繰り返す。

⑺ 受信側が⑸を、全受信可能メッセージ(6.4.2 項参照)分処理するまで繰り返す。

図 22 メッセージ送信プロトコル



6.4.5.メッセージ認証子送信プロトコル

6.4.5.1.機能

メッセージ認証子送信プロトコルは、改ざん検知指定でのメッセージ送信の際に、送信側からメッセージ認証子を送信するプロトコルである。

6.4.5.2.シーケンス

⑴ 送信側がメッセージ認証子を作成する。6. 4. 2項のメッセージ送信要求プロトコルにおいて、暗号化方式を指定してある場合は、作成したメッセージ認証子を暗号化する。

⑵ 送信側がメッセージ認証子送信通知とメッセージ認証子を送信する。

「メッセージ認証子送信通知」MH

- MH.msgId=メッセージ認証子送信通知
- MH.length=メッセージ認証子長
- MH.opt=0

⑶ 受信側がメッセージ認証子送信通知とメッセージ認証子を受信する。暗号化指定時は受信したメッセージ認証子を復号する。以下の場合はエラーとし終了する。

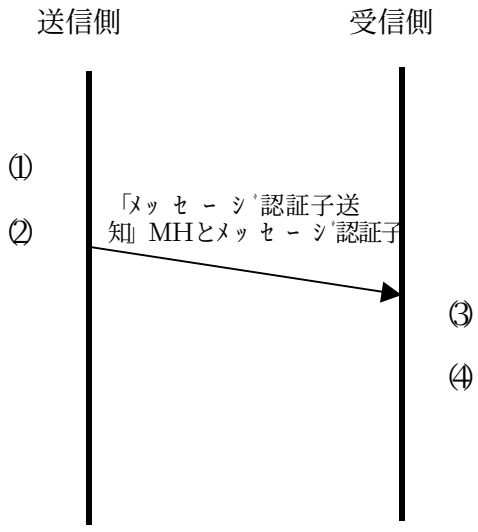
・MH.msgId がメッセージ認証子送信通知でない。

⑷ 受信側がメッセージ認証を行う。メッセージ認証子はメッセージ全体に対して作成される。また、⑴と同様に暗号化方式を指定してある場合は、以下の通りとする。

・メッセージ認証子は暗号化前のメッセージに対して作成される。

・作成したメッセージ認証子に対して暗号化が行われる。

図 23 メッセージ認証子送信プロトコル



6.5.スルーモード送信プロトコル

6.5.1.機能

スルーモード送信プロトコルは、高速通信用に手順を省略したメッセージ送信のプロトコルである。本プロトコルは6．4項のメッセージ送受信プロトコルとは独立したものであり、暗号化及び改ざん検知の指定は行えない。また、通信ブロックサイズに制限されず、ソケット等の下位プロトコルのバッファサイズの制限を受ける。

6.5.2.シーケンス

⑴ 送信側がスルーモード送信通知とメッセージを送信する。

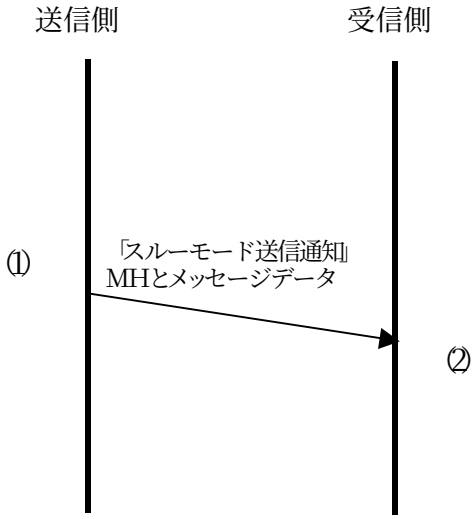
「スルーモード送信通知」MH

- MH.msgId=スルーモード送信通知
- MH.length=メッセージ長
- MH.opt=0

⑵ 受信側がスルーモード送信通知とメッセージを受信する。

MH.msgId がスルーモード送信通知の時、MH.lengt h分のメッセージを受信する。

図 24 スルーモード送信プロトコル



6.6.回線切断プロトコル

6.6.1.機能

回線切断プロトコルは、切断要求側からの要求に対し、応答するプロトコルである。

6.6.2.シーケンス

⑴ 要求側が切断要求を送信する。

「切断要求」MH

- MH.msgId=切断要求
- MH.length=0
- MH.opt=0

⑵ 応答側が切断要求を受信する。

⑶ 応答側が切断応答を送信する。

「切断応答」MH

- MH.msgId=切断応答
- MH.length=0
- MH.opt=切断可能なら ACK、不可能なら NAK

・切断可能の場合、接続を終了する。

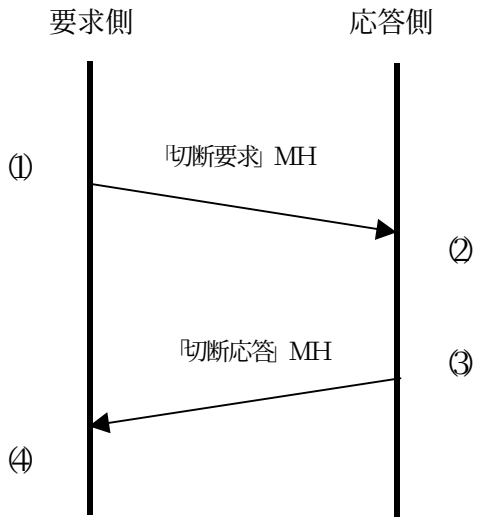
・切断不可能の場合、本プロトコルを終了して、次の処理へ移行する。

⑷ 要求側が切断応答を受信する。

・MH.opt が ACK の場合、接続を終了する。

・MH.opt が NAK の場合、本プロトコルを終了して、次の処理へ移行する。

図 25 回線切断プロトコル



6.7.例外プロトコル

6.7.1.相互認証パス２プロトコルの例外プロトコル1

6.7.1.1.機能

本プロトコルは、相互認証パス１プロトコルにおいて INTERNAL AUTHENTICATE コマンドが失敗した場合の相互認証パス２プロトコルである。

6.7.1.2.シーケンス

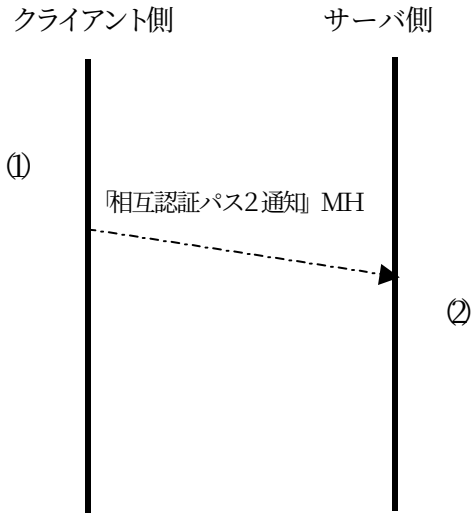
⑴ クライアント側が以下のMHを送信し、終了する。

「相互認証パス２通知」MH

- MH.msgId=相互認証パス 2 通知
- MH.length=0
- MH.opt= INTERNAL AUTHENTICATE ｺﾏﾝﾄﾞ結果ｺｰﾄﾞ(≠0)

⑵ サーバ側が相互認証パス２通知を受信し、MH.opt が 0 でないので、エラー終了する。

図 26 相互認証パス2プロトコルの例外プロトコル1



6.7.2.相互認証パス２プロトコルの例外プロトコル2

6.7.2.1.機能

本プロトコルは、相互認証パス１プロトコルにおいて GET CHALLENGE コマンドが失敗した場合の相互認証パス２プロトコルである。

6.7.2.2.シーケンス

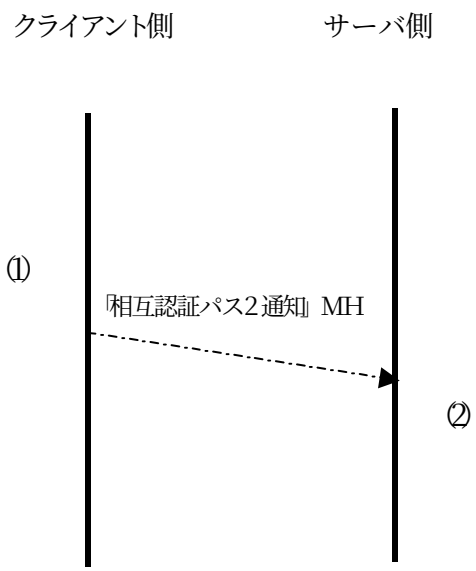
⑴ クライアント側が以下のMHを送信し、終了する。

「相互認証パス２通知」MH

- MH.msgId=相互認証パス 2 通知
- MH.length=0
- MH.opt= GET CHALLENGE ｺﾏﾝﾄﾞ結果ｺｰﾄﾞ(≠H0)

⑵ サーバ側が相互認証パス２通知を受信し、MH.opt が 0 でないので、エラー終了する。

図 27 相互認証パス2プロトコルの例外プロトコル2



6.7.3.相互認証パス３プロトコルの例外プロトコル1

6.7.3.1.機能

本プロトコルは、相互認証パス２プロトコルにおいて EXTERNAL AUTHENTICATE コマンドに失敗した場合の相互認証パス３プロトコルである。

6.7.3.2.シーケンス

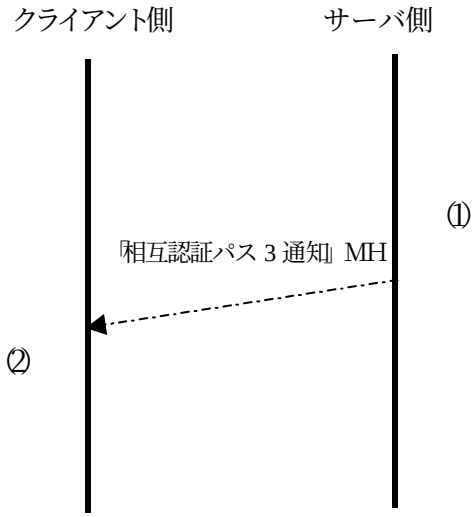
⑴ サーバ側が以下のMHを送信し、終了する。

「相互認証パス 3 通知」 MH

- MH.msgId=相互認証パス 3 通知
- MH.length=0
- MH.opt= EXTERNAL AUTHENTICATE ｺﾏﾝﾄﾞ結果ｺｰﾄﾞ(≠0)

⑵ クライアント側が相互認証パス 3 通知を受信し、MH.opt が 0 でないのでエラー終了する。

図 28 相互認証パス3プロトコルの例外プロトコル1



6.7.4.相互認証パス３プロトコルの例外プロトコル2

6.7.4.1.機能

本プロトコルは、相互認証パス３プロトコルにおいて INTERNAL AUTHENTICATE コマンドに失敗した場合のプロトコルである。

6.7.4.2.シーケンス

⑴ サーバ側が INTERNAL AUTHENTICATE コマンドによるレスポンスコードの生成に失敗する。

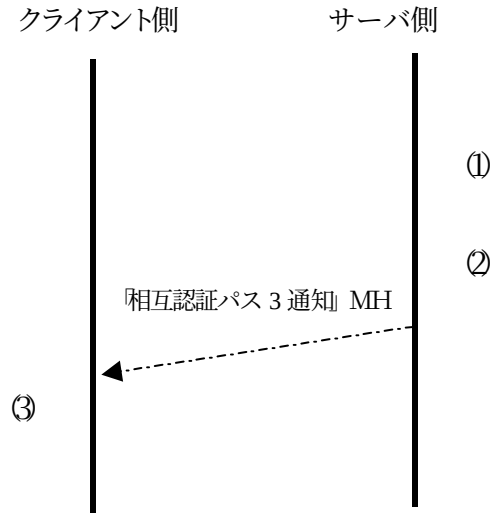
⑵ サーバ側が以下のMHを送信し、終了する。

「相互認証パス 3 通知」MH

- MH.msgId=相互認証パス 3 通知
- MH.length=0
- MH.opt= INTERNAL AUTHENTICATE ｺﾏﾝﾄﾞ結果ｺｰﾄﾞ(≠0)

⑶ クライアント側が相互認証パス 3 通知を受信し、MH.opt が 0 でないのでエラー終了する。

図 29 相互認証パス3プロトコルの例外プロトコル2



6.7.5.メッセージ送信要求プロトコルの例外プロトコル

6.7.5.1.機能

本プロトコルは、メッセージ送信要求プロトコルの送信方式が不一致の場合のプロトコルである。

6.7.5.2.シーケンス

⑴ 送信側がメッセージ送信要求を作成する。

「メッセージ送信要求」MH

- MH.msgId=メッセージ送信要求
- MH.length=16
- MH.opt=0

⑵ 送信側が送信方式データを作成する。「送信方式データ」とは以下の(4バイト×4の)データを指す。

- 暗号化方式
- 暗号キーＩＤ
- メッセージ認証方式
- 送信希望メッセージ長

⑶ 送信側がメッセージ送信要求と送信方式データを送信する。

⑷ 受信側がメッセージ送信要求と送信方式データを受信する。

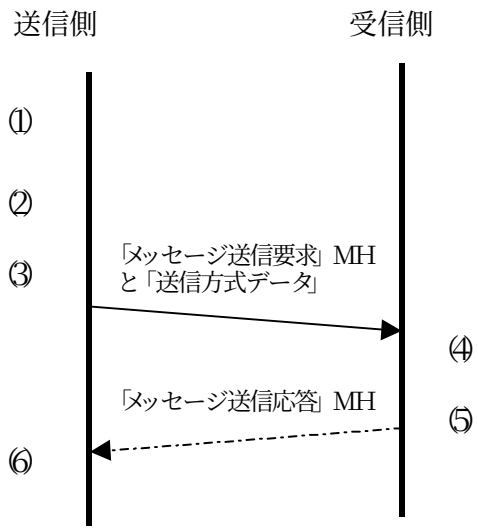
⑸ 受信側が暗号方式あるいはメッセージ認証方式が一致しない場合、以下のMHを送信し、終了する。

「メッセージ送信応答」MH

- MH.msgId=メッセージ送信応答
- MH.length=0
- MH.opt=NAK

⑹ 送信側がメッセージ送信応答を受信し、MH.opt が ACK でないのでエラー終了する。

図 30 メッセージ送信要求プロトコルの例外プロトコル



6.7.6.セッション鍵共有プロトコルの例外プロトコル

6.7.6.1.機能

本プロトコルは、セッション鍵共有プロトコルにおいて送信側がセッション鍵の生成に失敗した場合、受信側にエラーを通知するプロトコルである。また、本プロトコルはセッション鍵の共有が必要なときのみ発生するプロトコルである。

6.7.6.2.シーケンス

（1）送信側が GET CHALLENGE コマンドにより、セッション鍵生成用乱数を作成し、INTERNAL AUTHENTICATE コマンドにより、セッション鍵生成用乱数からセッション鍵を生成する。

（2）送信側が以下のMHを送信し、終了する。

「セッション鍵生成用乱数送信要求」MH

- MH.msgId=セッション鍵生成用乱数送信要求
- MH.length=0
- MH.opt= GET CHALLENGE ｺﾏﾝﾄﾞ結果ｺｰﾄﾞ
  または INTERNAL AUTHENTICATE
  ｺﾏﾝﾄﾞ結果ｺ (≠0)

（3）受信側がセッション鍵生成用乱数送信要求を受信し、MH.opt が 0 でないので、エラー終了する。

図 31 セッション鍵共有プロトコルの例外プロトコル