

# Telnetサーバ リファレンスマニュアル

株式会社ルネサス北日本セミコンダクタ

## ご注意

1. 本製品(ソフトウェア製品及びその関連ソフトウェア製品を含む。以下、同じ。)の使用に際しては、「外国為替及び外国貿易法」等、技術輸出に関する日本及び関連諸国の関係法規の遵守が必要となります。
2. 弊社は、本製品の使用に際しては、弊社もしくは第三者の特許権、著作権、商標権、その他の知的所有権等の権利に関し、別途、個別の契約書等(マニュアルの記載を含む。以下、同じ。)にて弊社による明示的な許諾がある場合を除き、その保証または実施権の許諾を行うものではありません。また本製品を使用したことにより第三者の知的所有権等の権利に関わる問題が生じた場合、弊社はその責を負いませんので予めご了承ください。
3. 本製品およびその仕様、またはマニュアルに記載されている事柄については、将来、事前の予告なしに変更することがありますので、最終的な設計、ご購入、ご使用に際しましては、事前に最新の製品規格または仕様書(マニュアルを含む)をご確認ください。
4. 本製品の使用(マニュアル記載事項に基づくものも含む)により直接または間接に生ずるいかなる損害についても、弊社は一切の責任を負いません。また、本製品の配布に使用される搭載機器や媒体が原因の損害に対しましても、弊社は一切の責任を負いません。
5. 本製品を、宇宙、航空、原子力、燃焼制御、運輸、交通、各種安全装置、ライフサポート関連の医療機器等のように、特別な品質・信頼性が要求され、その故障や誤動作が直接人命を脅かしたり、人体に危害を及ぼす恐れのある用途向けには使用できません。お客様の用途がこれに該当するかどうか疑問のある場合には、事前に弊社営業担当迄ご相談をお願い致します。
6. 本製品を使用してお客様のシステム製品を設計される際には、通常予測される故障発生率、故障モードをご考慮の上、本製品の動作が原因での事故、その他の拡大損害を生じないようにフェールセーフ等の十分なシステム上の対策を講じて頂きますようお願い致します。
7. 本製品およびマニュアルの著作権は弊社が所有しております。お客様は、弊社から提供された本製品を、別途、個別の契約書等にて定める場合を除き、いかなる場合においても全体的または部分的に複写・解析・改変することはできないものとします。
8. お客様は、別途、個別の契約書等にて定める場合を除き、本製品のマニュアルの一部または全部を無断で使用、複製することはできません。
9. 弊社は、本製品を1台のコンピュータで使用する権利をお客様に対してのみ許諾します。よって、本製品を第三者へ譲渡、貸与、賃借することは許諾しないものとします。但し、別途、個別の契約書等にて定められる場合はその条件に従います。
10. 本製品をはじめ弊社製品およびその関連製品についてのお問い合わせ、ご相談は弊社営業担当迄お願い致します。

$\mu$  ITRON は、Micro Industrial TRON の略称です。TRON は、The Realtime Operating system Nucleus の略称です。

その他、本書で登場するシステム名、製品名は各社の登録商標または商標です。

---

## はじめに

---

このマニュアルは、TCP/IPマネージャ上で動作するTCP/IPネットワークアプリケーション「Telnetサーバ」について説明します。

Telnetサーバは、TCP/IPマネージャを経由し、ネットワーク上のリモート端末（Telnetクライアント）からの接続要求を受けて、遠隔操作を支援する機能を提供します。

このリファレンスマニュアルではTelnetサーバのサービスコールとその使い方および関連事項を説明します。TCP/IPマネージャについては関連マニュアルを参照してください。

### 【関連マニュアル】

- TCP/IPマネージャ リファレンスマニュアル
- 使用するOSのユーザーズマニュアル

---

# 目次

---

1.	概要.....	1
1.1	Telnetの機能概要.....	1
1.2	サービスタスクの実現.....	2
1.3	サービスコールのしくみ.....	3
1.4	サポートする機能.....	4
1.5	Telnetサーバ サービスコールの種類.....	6
2.	Telnetdサービスコールの使用法.....	7
2.1	Telnetサーバの基本フロー.....	7
2.2	Telnetdサービスポイントの状態遷移図.....	9
3.	Telnetdサービスコール.....	10
3.1	初期化サービスコール.....	11
3.1.1	<i>TLd_init</i> Telnetサーバを初期化する.....	11
3.2	基本サービスコール.....	12
3.2.1	<i>TLd_start</i> Telnetサーバのサービスを開始する.....	12
3.2.2	<i>TLd_stop</i> Telnetサーバのサービスを終了する.....	14
3.2.3	<i>TLd_acp_srv</i> Telnetサーバを接続待ち状態にする.....	15
3.2.4	<i>TLd_sht_srv</i> Telnetクライアントの接続を切断する.....	16
3.2.5	<i>TLd_put_dat</i> Telnetクライアントにデータを送信する.....	17
3.2.6	<i>TLd_get_dat</i> Telnetクライアントからデータを受信する.....	18
3.2.7	<i>TLd_set_mod</i> Telnetサーバの動作モードを設定する.....	19
3.2.8	<i>TLd_get_mod</i> Telnetサーバの動作モードを取得する.....	22
3.2.9	<i>TLd_isu_opt</i> Telnetオプションを送信する.....	24
(1)	TLOPT_BINARY バイナリ転送オプション.....	26
(2)	TLOPT_ECHO エコーバックオプション.....	27
(3)	TLOPT_SGA GO AHEAD送信抑制オプション.....	28
(4)	TLOPT_STATUS ステータス要求オプション.....	29
(5)	TLOPT_TM タイミングマークオプション.....	30
(6)	TLOPT_TTYPE ターミナルタイプオプション.....	31
(7)	TLOPT_NAWS ウィンドウサイズオプション.....	32
(8)	TLOPT_TSPEED ターミナル速度オプション.....	33
(9)	TLOPT_LFLOW リモートフロー制御オプション.....	34
(10)	TLOPT_LINEMODE ラインモードオプション.....	35
3.2.10	<i>TLd_can_svp</i> ペンディングしている処理をキャンセルする.....	39
4.	コールバック.....	40
4.1	コールバックルーチン.....	41
4.1.1	<i>callback</i> Telnetサーバイベントを通知する.....	41
(1)	TEV_TLD_CONN Telnetクライアントの接続イベント.....	42
(2)	TEV_TLD_DISCONN Telnetクライアントの接続切断イベント.....	43
(3)	TEV_TLD_RCVDAT データの受信イベント.....	44
(4)	TEV_TLD_SNDEMP 送信バッファの空きイベント.....	45
(5)	TEV_TLD_BINARY バイナリモード要求の受信イベント.....	46
(6)	TEV_TLD_ECHO エコーバック要求の受信イベント.....	47
(7)	TEV_TLD_SGA 送受信切り替え抑制の受信イベント.....	48
(8)	TEV_TLD_STATUS Telnetオプション状態の受信イベント.....	49
(9)	TEV_TLD_LOGOUT 強制ログアウトの受信イベント.....	51

(10)	TEV_TLD_TTYPE	ターミナルタイプ情報の受信イベント	52
(11)	TEV_TLD_NAWS	ウィンドウサイズの変更イベント	53
(12)	TEV_TLD_TSPEED	ターミナル速度情報の受信イベント	54
(13)	TEV_TLD_LFLOW	リモートフロー制御要求の受信イベント	55
(14)	TEV_TLD_LINEMODE	ラインモード変更要求の受信イベント	56
(15)	TEV_TLD_EDIT	編集モード変更要求の受信イベント	57
(16)	TEV_TLD_SIGNAL	割り込み/シグナル通知変更要求の受信イベント	58
(17)	TEV_TLD_SOFTTAB	ソフトタブ変更要求の受信イベント	59
(18)	TEV_TLD_LITECHO	リテラルエコー変更要求の受信イベント	60
(19)	TEV_TLD_SLC	特殊文字変更要求の受信イベント	61

<b>5.</b>	<b>サービスコール エラーコード一覧</b>	<b>62</b>
-----------	-------------------------	-----------

---

## 図表目次

---

図1.1	Telnet構成 .....	1
図1.2	サービスタスク .....	2
図1.3	各APIとサービスタスクとの通信 .....	3
図2.1	TelnetdサービスポイントID .....	8
図2.2	Telnetサーバの基本フロー .....	8
図2.3	Telnetdサービスポイント状態遷移図.....	9
表1.1	Telnetコマンド サポート機能.....	4
表1.2	Telnetオプション サポート機能.....	5
表1.3	Telnetdサービスコール一覧 .....	6
表5.1	Telnetdサービスコール エラーコード一覧.....	62

# 1. 概要

## 1.1 Telnet の機能概要

Telnet はネットワークを介してリモート端末にログインし、アクセスする為の protocols です。Telnet プロトコルは Telnet サーバと Telnet クライアント間の通信を規定したもので、Telnet クライアントから入力されたデータはネットワーク仮想端末 (NVT) のデータフォーマットに変換され、Telnet サーバに送られます。Telnet サーバでは、それをローカル端末のデータフォーマットに変換してアプリケーションに渡します。サーバからクライアントに送られる場合も同様です。

Telnet は TCP プロトコル上で動作し、ログインではユーザ名、パスワードによって認証取得する必要があります。また、サーバ/クライアント間のネゴシエーションにより、ローカルエコーの指定、転送モード、端末タイプの設定等、両者間の通信モードを変更することができます。

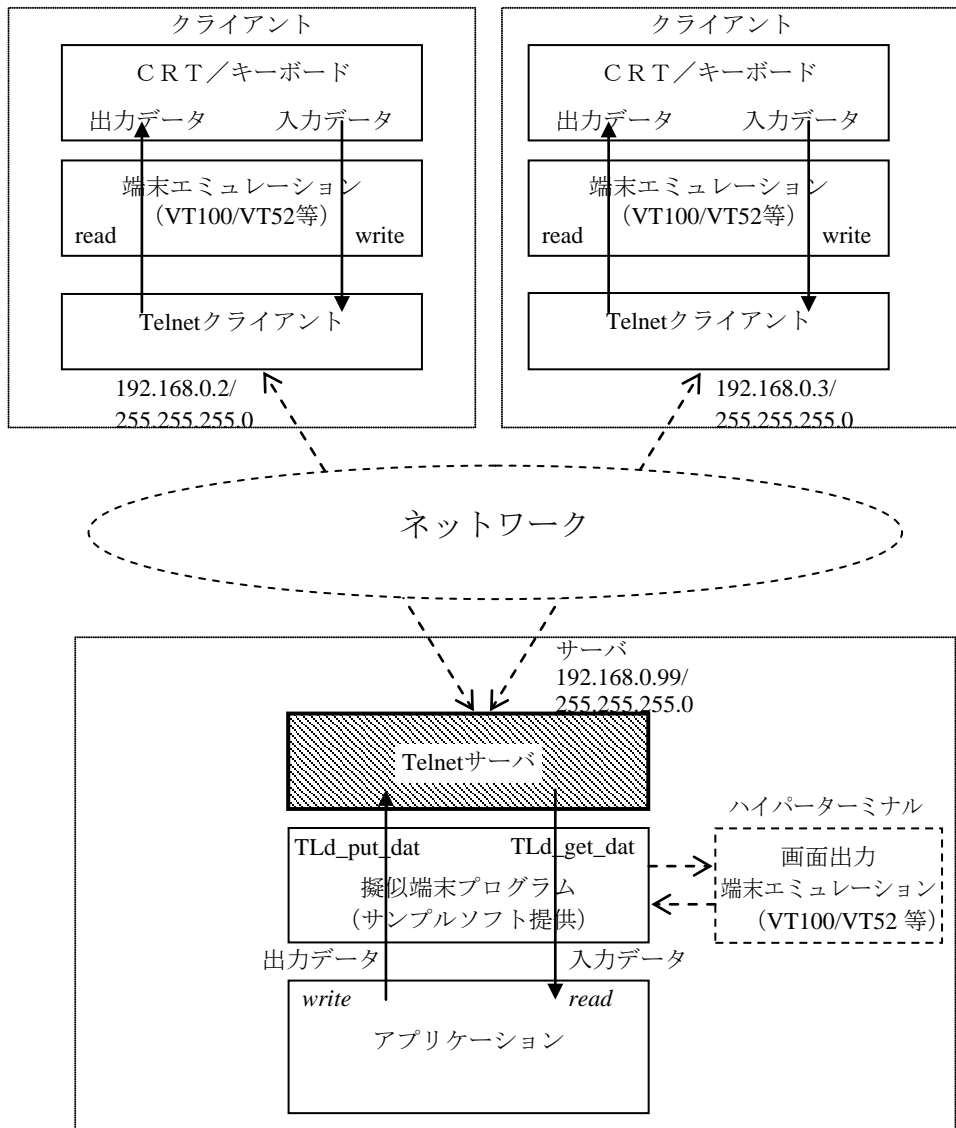


図1.1 Telnet構成

## 1.2 サービスタスクの実現

Telnet サーバ (以降、Telnetd と呼ぶ) のサービスは「タスク」として実現します。図 1.2 にサービスタスクの概念を示します。

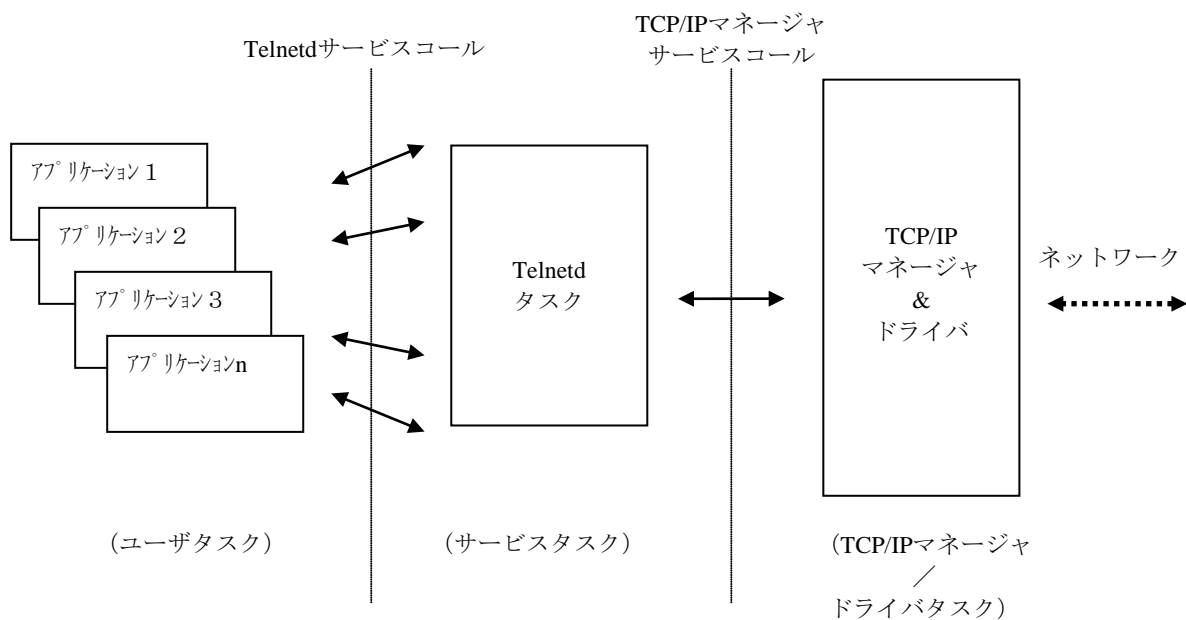


図1.2 サービスタスク



### 1.3 サービスコールのしくみ

各サービスコールは、拡張サービスコール形式で呼び出されます（TLd\_init のみサブルーチン形式）。各サービスコールとサービスタスクとは OS のメールボックス、およびイベントフラグ機能を利用してデータのやり取りを行います。

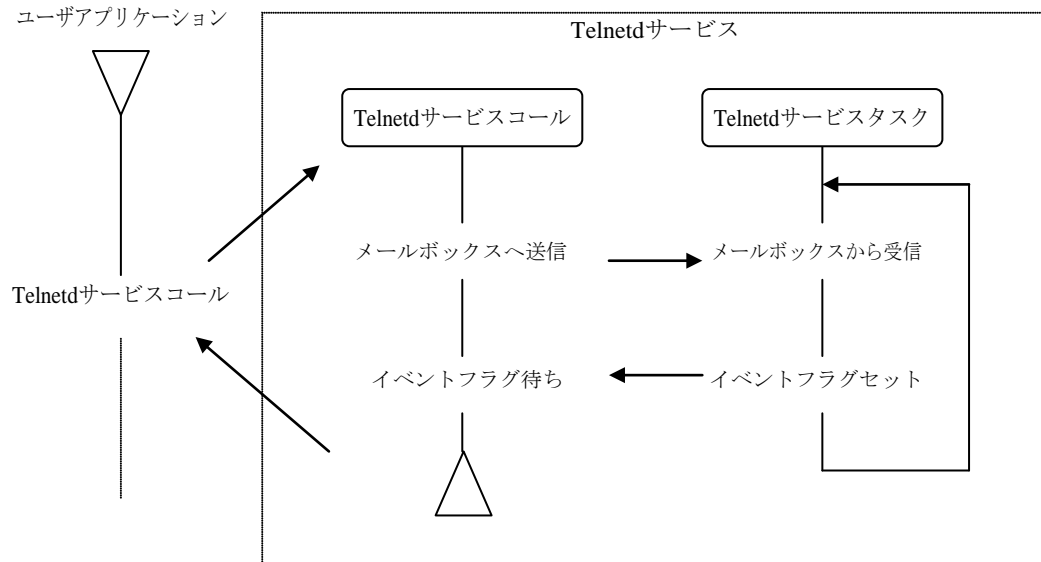


図1.3 各APIとサービスタスクとの通信

本 Telnet サーバでは OS のリソースのうち、1 タスク、1 メールボックス、1 イベントフラグ、および 10 個の拡張サービスコールを使用します。

## 1.4 サポートする機能

本 Telnet サーバがサポートする Telnet コマンドを表 1.1 に、Telnet オプション機能を表 1.2 に示します。サポート有無の欄に○印があるものがサポートする機能で、×印は未サポートの機能です。

表1.1 Telnetコマンド サポート機能

項目	説明	サポート有無	
Telnet コマンド	EOF	ファイル終端	○
	SUSP	プロセス一時停止	○
	ABORT	プロセス中断	○
	EOR	レコード終端	○
	SE	サブオプション終了	○
	NOP	何もしない	○
	DM	データマーク	○
	BRK	ブレーク	○
	IP	プロセス割り込み	○
	AO	出力停止	○
	ATY	相手存在確認	○
	EC	文字削除	○
	EL	行削除	○
	GA	送受信切り替え	○
	SB	サブオプション開始	○
	DO	オプション交渉	○
	DONT	オプション交渉	○
	WILL	オプション交渉	○
WONT	オプション交渉	○	

表1.2 Telnetオプション サポート機能

項目	説明	サポート有無	
Telnet オプション	TRANSMIT-BINARY	バイナリ転送	○
	ECHO	エコーバック	○
	RCP	再接続オプション	×
	SUPPRESS-GA	送受信切替の抑制	○
	NAMS	メッセージ長の調停	×
	STATUS	Telnet オプション状態通知	○
	TIMING-MARK	タイミングマーク	○
	RCTE	送信&エコーのリモート制御	×
	NAOL	行幅出力の調停	×
	NAOP	ページサイズ出力の調停	×
	NAOCRD	キャリッジリターン出力の調停	×
	NAOHTS	水平タブストップ出力の調停	×
	NAOHTD	水平タブキャラクタ処理の調停	×
	NAOFFD	フォームフィード出力処理の調停	×
	NAOVTS	垂直タブストップ出力の調停	×
	NAOVTD	垂直タブ出力処理の調停	×
	NAOLFD	ラインフィード出力の調停	×
	EXTEND-ASCII	拡張 ASCII	×
	LOGOUT	強制ログアウト	○
	BM	バイトマクロ	×
	DET	データエントリ端末	×
	SUPDUP	SUPDUP ディスプレイ制御	×
	SUPDUP-OUTPUT	SUPDUP 出力	×
	SEND-LOCATION	ロケーション ID 送信	×
	TERMINAL-TYPE	ターミナルタイプ	○
	END-OF-RECORD	レコード終端を示す文字の選択	×
	TUID	TACACS ユーザ識別	×
	OUTMRK	マーク出力	×
	TTYLOC	端末ロケーション番号	×
	3270-REGIME	TELNET3270 様式	×
	X3-PAD	X3PAD	×
	NAWS	ウィンドウサイズの調停	○
	TERMINAL-SPEED	端末スピード	○
	TOGGLE-FLOW-CONTROL	リモートフロー制御	○
	LINEMODE	ラインモード	○
	XDISPLOC	X ディスプレイロケーション	×
	ENVIRON	環境変数	×
	AUTHENTICATION	認証	×
	ENCRYPTION	暗号	×
	NEW-ENVIRON	新環境変数	×
TN3270E	TELNET3270 拡張	×	
EXOPL	拡張オプション	×	

## 1.5 Telnet サーバ サービスコールの種類

以下に本 Telnetd のサービスコールの一覧を示します。

表1.3 Telnetdサービスコール一覧

区分	サービスコール名称	サービスコールの機能
初期化サービスコール	TLd_init	Telnetサーバのサービスを初期化する
基本サービスコール	TLd_start	Telnetサーバのサービスを開始する
	TLd_stop	Telnetサーバのサービスを終了する
	TLd_acp_srv	Telnetサーバを接続待ち状態にする
	TLd_sht_srv	Telnetクライアントの接続を切断する
	TLd_put_dat	Telnetクライアントにデータを送信する
	TLd_get_dat	Telnetクライアントからデータを受信する
	TLd_set_mod	Telnetサーバの動作モードを設定する
	TLd_get_mod	Telnetサーバの動作モードを取得する
	TLd_isu_opt	Telnetオプションを送信する
	TLd_can_svp	Telnetサーバのペンディング処理のキャンセル
コールバックルーチン	TLd_callback (仮称)	Telnetサーバイベントの通知 (ユーザ実装ルーチン)

---

## 2. Telnetd サービスコールの使用方法

---

Telnetd サービスコールを使用するには TCP/IP マネージャおよびドライバプログラムを動作可能な状態にする必要があります。TCP/IP マネージャおよびドライバプログラムの使用方法に関しては、TCP/IP マネージャ リファレンスマニュアル、および各ドライバプログラムのマニュアルを参照してください。

Telnetd を使うプログラムでは次のヘッダファイルをインクルードしてください。

```
#include "tcpip03.h"          /* TCP/IP マネージャ定義ファイル */
#include "tld02.h"           /* Telnetd 定義ファイル */
```

一般的な Telnet サーバの使用方法を以下に説明します。

### 2.1 Telnet サーバの基本フロー

- (1) 最初に `TLd_init` をコールする必要があります。 `TLd_init` はシステムリセットの際などに一度のみコールします。 `TLd_init` にて各 Telnetd サービスコールが定義されます。(図 2.2 ①)
- (2) `TLd_start` をコールすると、Telnetd が使用するタスク、メモリ等のリソースが確保され、Telnet サーバの各サービスコールが使用可能となります。(図 2.2 ②)
- (3) `TLd_acp_srv` をコールすることでクライアントからの接続要求待ちとなります。(図 2.2 ③)
- (4) クライアントから接続される度に Telnetd サービスポイントが生成されます。Telnet サーバでは接続されたクライアントを Telnetd サービスポイント ID で区別します(図 2.1参照)。Telnetd サービスポイント ID は、クライアント接続イベントの通知 (`TLd_callback` (仮称)) のパラメータにより通知されます。これによって、同時に複数のクライアントからの接続を受け、制御することも可能です。(図 2.2 ④)
- (5) Telnet クライアントからデータを受信するには `TLd_get_dat` を使用します。Telnet クライアントにデータを送信するには `TLd_put_dat` を使用します。(図 2.2 ⑤)
- (6) クライアントを切断、および Telnetd サービスポイントを削除するには `TLd_sht_svp` をコールします。切断された Telnetd サービスポイント ID はフリーとなり、以降のクライアントの接続にて再び使用されます。また、クライアント側から切断された場合も同様です。(図 2.2 ⑥)
- (7) Telnet サーバを停止する場合は、`TLd_stop` をコールします。Telnetd が使用するタスク、メモリ等のリソースが解放されます。Telnet サーバを停止した後であれば、再度 `TLd_start` をコールすることができます。(図 2.2 ⑦)

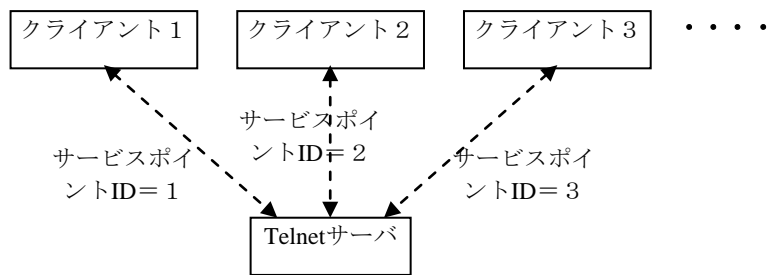


図2.1 TelnetdサービスポイントID

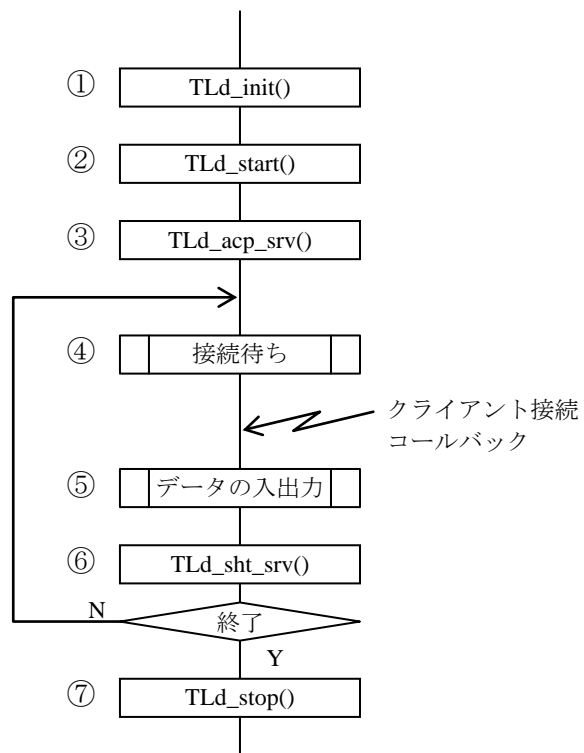


図2.2 Telnetサーバの基本フロー

## 2.2 Telnetd サービスポイントの状態遷移図

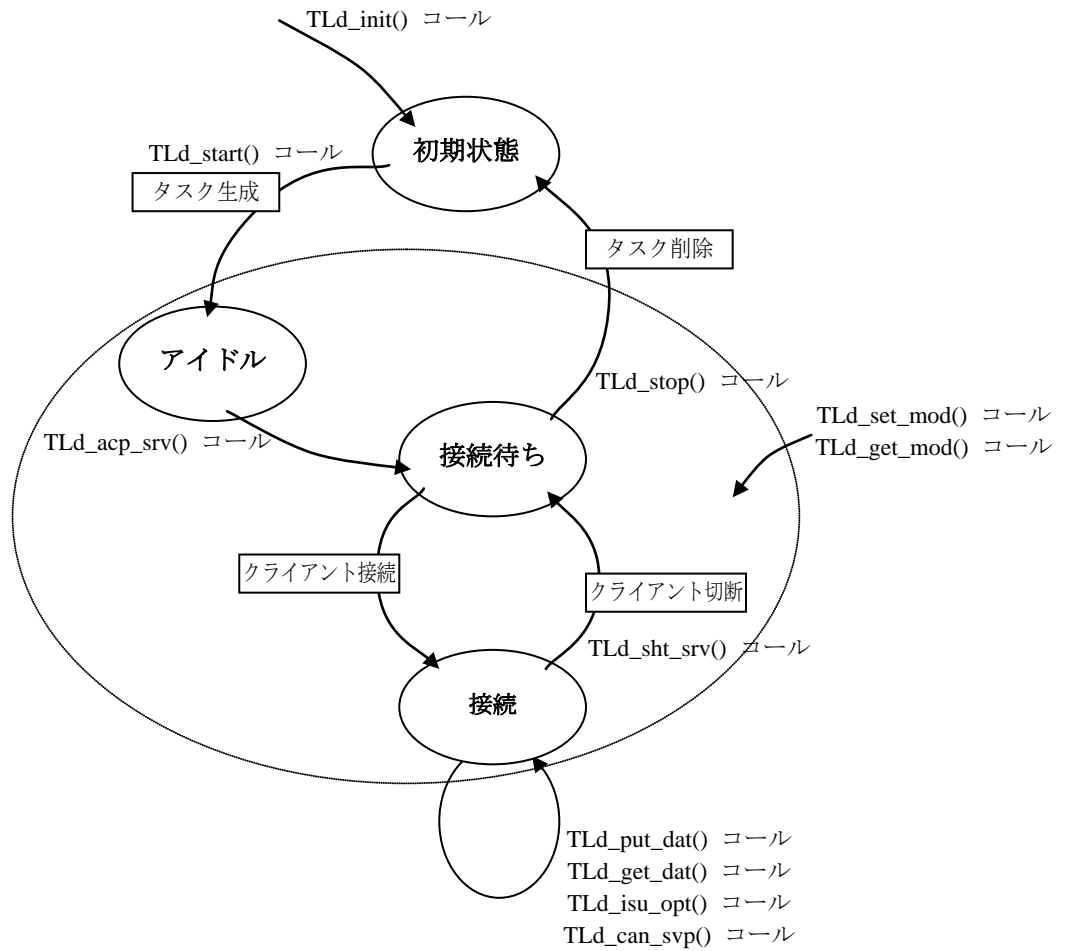


図2.3 Telnetdサービスポイント状態遷移図

---

### 3. Telnetd サービスコール

---

各サービスコールについての説明は次の形式で行います。

No.	サービスコール名	機能	【発行可能なシステム状態*1】
C言語インタフェース			
サービスコール呼出し形式			
パラメータ			
型	パラメータ	パラメータの意味	
・	・	・	
・	・	・	
・	・	・	
リターンパラメータ			
型	パラメータ	パラメータの意味	
・	・	・	
・	・	・	
パケットの構造			
リターン値/エラーコード			
リターン値またはニモニック	リターン値またはエラーコードの意味		
・	・		
・	・		
・	・		
解 説			
.....			

\*1発行可能なシステム状態を以下のアルファベットで示します

T：タスク実行状態

D：ディスパッチ禁止状態

L：CPUロック状態

I：非タスク部実行状態

なお、各状態の詳細は各OSのユーザーズマニュアルを参照してください。

発行可能なシステム状態以外の状態でサービスコールを発行した場合、システムの正常な動作は保証されません。



## 3.1 初期化サービスコール

### 3.1.1 TLd\_init Telnet サーバを初期化する

【T/D/I】

C言語インタフェース

void TLd\_init ( void );

パラメータ

なし

リターンパラメータ

なし

リターン値/エラーコード

なし

解 説

Telnetサーバで使用する内部変数を初期化し、各Telnetdサービスコールを登録します。  
Telnetサーバの各サービスコールを使用する前に、必ず本サービスコールを呼び出す必要があります。

拡張サービスコールの機能コード不足等で拡張サービスコールが登録できなかった場合でもエラーコードは返りません。この場合、TLd\_startをコールすると拡張サービスコールが未定義であるエラーコード(E\_RSFN)、またはシステムエラー(E\_SYS)が返されます。

## 3.2 基本サービスコール

### 3.2.1 TLd\_start Telnet サーバのサービスを開始する

【T】

C 言語インタフェース

```
ER ercd = TLd_start ( T_TLD_STA *pk_sta );
```

パラメータ

T_TLD_STA	*pk_sta;	Telnetサーバ情報を格納した領域の先頭アドレス
-----------	----------	---------------------------

リターンパラメータ

ER	ercd	リターン値またはエラーコード
----	------	----------------

パケットの構造

```
typedef struct{
    T_TLD_RSC    resource;    リソース情報
    T_TLD_OSC    osconf;     OSコンフィグレーション情報
    H            Perrno;     発生したエラーの詳細を示す番号
} T_TLD_STA;
```

```
typedef struct{
    ID    maxsvpid;    最大TelnetdサービスポイントID
    UW    *memadr;    使用するメモリの先頭アドレス
    W     memlen;     使用可能なメモリの長さ
} T_TLD_RSC;
```

```
typedef struct{
    ATR    tskatr;    Telnetサーバタスク属性
    ID     tskid;    TelnetサーバタスクID番号
    PRI    tskpri;    Telnetサーバタスク優先度
    INT    stksz;    Telnetサーバタスクスタックサイズ
    ID     mbxid;    TelnetサーバメールボックスID番号
    ID     evfid;    TelnetサーバイベントフラグID番号
} T_TLD_OSC;
```

リターン値/エラーコード

E_OK	正常終了
E_PAR	パラメータエラー (pk_sta、およびmemadrが4の倍数以外、maxsvpid ≤ 0、maxsvpid > 10)
E_NOMEM	メモリ不足 (必要なメモリが確保できない)
E_OBJ	オブジェクト状態エラー (すでにサービスが開始している)
E_SYS	システムエラー (OSのリソース確保に失敗)

解 説

Telnetサーバのサービスを開始します。

maxsvpidには使用するTelnetdサービスポイントIDの最大値を指定します。TelnetdサービスポイントIDは同時に接続を許可するクライアントの数分必要となります。(注1)

Telnetdサービスコールでは、必要となるワークメモリをmemadrで指定されたメモリアドレスから切り出して使用します。切り出したメモリの長さをmemlenに返します。最大TelnetdサービスポイントIDに比例して使用するワークメモリが多くなります。

ワークメモリはTLd\_stopをコールした後、解放されます。解放前にワークメモリの内容を書き換えた場合の動作は保証しません。

指定したmemlenが必要なメモリサイズみ満たない場合は、エラーコードとしてE\_NOMEMを返します。

Telnetサーバが必要とするワークメモリのサイズは以下の計算式にて算出できます。

$$\begin{aligned} \text{ワークメモリサイズ} = & 664 + (\text{maxsvpid} \times 3284) \\ & + (\text{maxsvpid} \times \text{送信ウィンドウバッファサイズ}) \\ & + (\text{maxsvpid} \times \text{受信ウィンドウバッファサイズ}) \quad (\text{バイト}) \end{aligned}$$

tasktr、tskid、tskpri、stkszには、Telnetサーバタスクのタスク属性、タスクID番号、タスク優先度、タスクスタックサイズを指定します。tskidに0を指定した場合、OSのID番号自動割付け機能によりID番号を割り付け、割り付けられたID番号をtskidに返します。

mbxid、evfidには、Telnetサーバが生成するメールボックスのID番号、およびイベントフラグのID番号を指定します。mbxid、evfidに0を指定した場合は、OSのID番号自動割付け機能によりID番号を割り付け、割り付けられたID番号をmbxid、evfidに返します。

指定したパラメータが不正の場合は、エラーコードとしてE\_PARを返し、perrnoに詳細エラーコードを返します。

OSのリソース確保に失敗した場合は、エラーコードとしてE\_SYSを返し、perrnoに詳細エラーコードを返します。

Telnetサーバ定義ファイル本体で設定した送信/受信用のTCPウィンドウバッファサイズが2より小さい、または奇数であった場合には、エラーコードとしてE\_SYSを返し、perrnoに-5を返します。

TLd\_initコール時に拡張サービスコールの登録に失敗している場合には、エラーコードとしてE\_SYSを返し、perrnoに登録に失敗した拡張サービスコールの機能コードを返します。

すでにTelnetサーバが開始している状態で本サービスコールをコールした場合、エラーコードとしてE\_OBJを返します。

リターン値がE\_PAR、またはE\_SYSの場合、発生したエラーの詳細を示す番号をperrnoに格納します。perrnoに格納される番号の種類を以下に示します。ただし、pk\_staが不正なアドレスの場合、perrnoには格納しません。

リターン値がE\_PARの場合：

perrno	エラー内容
-1	maxsvpidが11以上、または0以下
-2	memadrが4の倍数以外、または0
-255	その他のパラメータエラー

リターン値がE\_SYSの場合：

perrno	エラー内容
-1	メールボックスの生成に失敗
-2	イベントフラグの生成に失敗
-3	タスクの生成に失敗
-4	タスクの起動に失敗
-5	送信/受信用TCPウィンドウバッファサイズに2より小さい値または奇数を設定
-255	その他のシステムエラー
1以上	TLd_initにて登録に失敗した拡張サービスコールの機能コード

(注1) TCP/IPマネージャの最大通信端点数、最大受付口数、およびTCP/IPマネージャを使用している他プログラムの通信端点、受付口の状況によって同時に接続できるクライアントの数が最大TelnetdサービスポイントIDより少なくなる場合があります。

### 3.2.2 TLd\_stop

### Telnet サーバのサービスを終了する

【T】

C 言語インタフェース

```
ER ercd = TLd_stop ( void );
```

パラメータ

なし

リターンパラメータ

ER                      ercd                      リターン値またはエラーコード

リターン値/エラーコード

E\_OK                      正常終了

解 説

Telnetサーバのサービスを終了し、TLd\_startで切り出したワークメモリを解放します。  
クライアントがまだ接続されている場合は全て強制切断されます。

全てのTelnetdサービスポイントに対し、ペンディング中の処理がある場合は強制終了します。強制終了されたサービスコールは、エラーコードとしてE\_RLWAIを返します。

### 3.2.3 TLd\_acp\_srv Telnet サーバを接続待ち状態にする

【T】

C 言語インタフェース

```
ER_UINT ercd = TLd_acp_srv ( T_TLD_PAR *par );
```

パラメータ

T_TLD_PAR	*par	Telnetサーバ接続情報の先頭アドレス
-----------	------	----------------------

リターンパラメータ

ER	ercd	リターン値またはエラーコード
----	------	----------------

パケットの構造

```
typedef struct{
    UW    myaddr;    Telnetサーバが使用するIPアドレス
    UH    myport;    Telnetサーバが使用するポート番号 (0の場合、
                    デフォルトポート番号23を使用)
    FP    callback;  イベント通知コールバックルーチンのアドレス
} T_TLD_PAR;
```

リターン値/エラーコード

E_OK	正常終了
E_PAR	パラメータエラー (parが4の倍数以外、IPアドレスが不正、ポート番号がすでに使用されている、callbackが0または奇数)
E_NOID	ID番号不足 (TCP通信端点ID、またはTCP受付口IDが不足)
E_OBJ	オブジェクト状態エラー (サービスが開始していない、またはすでに接続待ちになっている)

解 説

Telnetサーバのサービスを接続待ち状態にします。

myaddrには、Telnetサーバが使用するIPアドレスを32ビット (4バイト) で指定します。

myportには、Telnetサーバが使用するポート番号を指定します。myportが0の場合は、デフォルトポート番号23を使用します。

callbackには、Telnetdイベントを通知するためのコールバックルーチンのアドレスを指定します。

コールバックルーチンの詳細に関しては「4 コールバック」の項を参照してください。

Telnetクライアントから接続されると、Telnetオプションのネゴシエーションが行われた後、クライアントが接続したことを通知するコールバックルーチンがコールされ、その際にTelnetdサービスポイントIDがパラメータとして渡されます。TelnetdサービスポイントIDは、以降のTelnetサーバのサービスを使用する場合に必要となります。

一度TLd\_acp\_srvをコールするとTLd\_stopがコールされるか、または同時接続されたクライアントの数が最大TelnetdサービスポイントIDに達するまで、クライアントの接続が許可されます。(注1)

指定したパラメータが不正の場合、または指定したポート番号がすでに使用されている場合は、エラーコードとしてE\_PARを返します。

TCP/IPマネージャの通信端点、または受付口が確保できなかった場合は、エラーコードとしてE\_NOIDを返します。

すでに接続待ちとなっている状態で本サービスコールをコールした場合、エラーコードとしてE\_OBJを返します。

(注1) TCP/IPマネージャの最大通信端点数、最大受付口数、およびTCP/IPマネージャを使用している他プログラムの通信端点、受付口の状況によって同時に接続できるクライアントの数が最大TelnetdサービスポイントIDより少なくなる場合があります。

### 3.2.4 TLd\_sht\_srv Telnet クライアントの接続を切断する

【T】

C言語インタフェース

```
ER ercd = TLd_sht_srv ( ID svpid , TMO tmout );
```

パラメータ

ID	svpid	TelnetdサービスポイントID
TMO	tmout	タイムアウト値

リターンパラメータ

ER	ercd	リターン値またはエラーコード
----	------	----------------

リターン値/エラーコード

E_OK	正常終了
E_PAR	パラメータエラー (tmout < -1)
E_ID	不正ID番号 (TelnetdサービスポイントIDは無効)
E_OBJ	オブジェクト状態エラー (サービスが開始していない、または接続状態でない)
E_TMOUT	ポーリング失敗またはタイムアウト
E_RLWAI	処理のキャンセル、待ち状態の強制解除

解 説

接続されているクライアントを切断します。

svpidには切断するTelnetdサービスポイントIDを指定します。

tmoutにはクライアントからの応答待ち時間を指定します。tmoutに正の値を指定した場合は待ち時間(単位はms)、TMO\_POL(0)を指定した場合はポーリング、TMO\_FEVR(-1)を指定した場合は永久待ちとなります。

ポーリング指定、および待ち時間指定によりタイムアウトが発生した場合は、RST送信による強制切断を行い、リターン値としてE\_TMOUTを返します。ただし、ペンディング中の処理がある場合はペンディング中の処理が完了後、切断処理を開始します。切断処理を開始する前にタイムアウトが発生した場合は切断処理を行わず、リターン値としてE\_TMOUTを返します。

切断されたTelnetdサービスポイントIDは使用不可となり、以降のクライアントの接続にて再び使用されません。

サーバ側から切断した場合、TCP通信端点がTIMEWAIT状態となり、2×MSL時間の間、制御が戻りません。MSLについてはTCP/IPマネージャの関連するマニュアルを参照してください。

指定したパラメータが不正の場合は、エラーコードとしてE\_PARを返します。

指定したTelnetdサービスポイントIDが未生成の場合は、エラーコードとしてE\_IDを返します。

接続状態以外のサービスポイントに対して本サービスコールを発行した場合、エラーコードとしてE\_OBJを返します。

### 3.2.5 TLd\_put\_dat Telnet クライアントにデータを送信する

【T】

C 言語インタフェース

```
ER_UINT sdatlen = TLd_put_dat ( ID svpid , VP data , INT len , TMO tmout );
```

パラメータ

ID	svpid	TelnetdサービスポイントID
VP	data	送信するデータの先頭アドレス
INT	len	送信したいデータの長さ
TMO	tmout	タイムアウト値

リターンパラメータ

ER_UINT	sdatlen	リターン値またはエラーコード
---------	---------	----------------

リターン値/エラーコード

正の値	正常終了 (送信バッファに入れたデータの長さ)
E_PAR	パラメータエラー (dataが0、len≤0、tmout < -1)
E_ID	不正ID番号 (TelnetdサービスポイントIDは無効)
E_OBJ	オブジェクト状態エラー (サービスが開始していない、または接続状態でない)
E_TMOUT	ポーリング失敗またはタイムアウト
E_RLWAI	処理のキャンセル、待ち状態の強制解除
E_CLS	接続切断

解 説

dataで指定されたアドレスから長さlenのデータをTelnetクライアントに送信します。送信バッファに空きがある場合、送信バッファに入れたデータの長さを返します。送信バッファに空きがない場合は待ち状態となります。

svpidには送信に使用するTelnetdサービスポイントIDを指定します。

tmoutには送信バッファが空くまでの待ち時間を指定します。tmoutに正の値を指定した場合は待ち時間(単位はms)、TMO\_POL(0)を指定した場合はポーリング、TMO\_FEVR(-1)を指定した場合は永久待ちとなります。

送信バッファに空きが無い状態にて、ポーリング指定、および待ち時間指定によりタイムアウトが発生した場合、データは送信されず、リターン値としてE\_TMOUTを返します。

指定したパラメータが不正の場合は、エラーコードとしてE\_PARを返します。

無効なTelnetdサービスポイントIDを指定した場合は、エラーコードとしてE\_IDを返します。

接続状態以外のサービスポイントに対して本サービスコールを発行した場合は、エラーコードとしてE\_OBJを返します。

TCPコネクションが切断された場合は、エラーコードとしてE\_CLSを返します。この場合、受信データがあればTLd\_get\_datにてデータを取得することができますが、TLd\_put\_datにてデータを送信することはできません。

本サービスコールがペンディング状態でない場合に、コールバックにより送信バッファの空きイベントを通知することができます。コールバックルーチンの詳細に関しては「4 コールバック」の項を参照してください。

### 3.2.6 TLd\_get\_dat Telnet クライアントからデータを受信する

【T】

C 言語インタフェース

```
ER_UINT rdatlen = TLd_get_dat ( ID svpid , VP data , INT len , TMO tmout );
```

パラメータ

ID	svpid	TelnetdサービスポイントID
VP	data	受信データを格納する領域の先頭アドレス
INT	len	受信データを格納する領域の長さ
TMO	tmout	タイムアウト値

リターンパラメータ

ER_UINT	rdatlen	リターン値またはエラーコード
---------	---------	----------------

リターン値/エラーコード

正の値	正常終了 (受信したデータのサイズ)
E_PAR	パラメータエラー (dataが0、len≤0、tmout < -1)
E_ID	不正ID番号 (TelnetdサービスポイントIDは無効)
E_OBJ	オブジェクト状態エラー (サービスが開始していない、または接続状態でない)
E_TMOUT	ポーリング失敗またはタイムアウト
E_RLWAI	処理のキャンセル、待ち状態の強制解除
E_CLS	接続切断

解 説

Telnetクライアントからデータを受信し、dataで指定されたアドレスに格納します。受信データが無くなるか受信したデータの長さがlenに達するまでデータを取得し、取得したデータの長さを返します。受信データが無い場合は待ち状態となります。

svpidには受信に使用するTelnetdサービスポイントIDを指定します。

tmoutにはTelnetクライアントからの受信待ち時間を指定します。tmoutに正の値を指定した場合は待ち時間 (単位はms)、TMO\_POL(0)を指定した場合はポーリング、TMO\_FEVR(-1)を指定した場合は永久待ちとなります。

受信データが無い状態にて、ポーリング指定、および待ち時間指定によりタイムアウトが発生した場合、リターン値としてE\_TMOUTを返します。

指定したパラメータが不正の場合は、エラーコードとしてE\_PARを返します。

無効なTelnetdサービスポイントIDを指定した場合は、エラーコードとしてE\_IDを返します。

接続状態以外のサービスポイントに対して本サービスコールを発行した場合、エラーコードとしてE\_OBJを返します。

TCPコネクションが切断された場合は、エラーコードとしてE\_CLSを返します。

本サービスコールがペンディング状態でない場合、コールバックによりデータの受信イベントを通知することができます。コールバックルーチンの詳細に関しては「4 コールバック」の項を参照してください。



### 3.2.7 TLd\_set\_mod Telnet サーバの動作モードを設定する

【T】

C 言語インタフェース

```
ER ercd = TLd_set_mod ( ID svpid , INT modname , VP modval );
```

パラメータ

ID	svpid	TelnetdサービスポイントID
INT	modname	設定する動作モードの種類
VP	modval	設定内容を格納した領域の先頭アドレス

リターンパラメータ

ER	ercd	リターン値またはエラーコード
----	------	----------------

リターン値/エラーコード

E_OK	正常終了
E_PAR	パラメータエラー (modnameの値が不正、modvalが0、または4の倍数以外、modvalの内容が不正)
E_ID	不正ID番号 (TelnetdサービスポイントIDは無効)
E_OBJ	オブジェクト状態エラー (サービスが開始していない)

解 説

Telnetサーバのサービス、または各Telnetdサービスポイントの動作モードを変更します。

svpidには変更するTelnetdサービスポイントIDを指定します。svpidに0を指定した場合は、Telnetサーバのデフォルト設定を変更します。(デフォルト設定の変更の場合、すでに接続されているTelnetdサービスポイントには影響しません。)

modnameには、TLD\_MOD\_SVP (Telnetサーバ動作モードの設定)、TLD\_MOD\_CBK (コールバックイベントの指定)、TLD\_MOD\_ASC (Telnetサーバテキスト形式の設定)、およびTLD\_MOD\_OPT (Telnetオプションの設定) が指定できます。

指定したパラメータが不正の場合は、エラーコードとしてE\_PARを返します。

無効なTelnetdサービスポイントIDを指定した場合は、エラーコードとしてE\_IDを返します。

(1) modname = TLD\_MOD\_SVP (Telnetサーバ動作モードの設定)

Telnetdサービスポイントの動作モードを指定します。modvalには、T\_TCEP\_MODE型へのアドレスを指定します。T\_TCEP\_MODE型の詳細に関しては、「TCP/IPマネージャ リファレンスマニュアル」の「4.4.18 tcp\_set\_opt TCP通信端点オプション設定」の項を参照してください。

デフォルト設定では、cepemodeがTMD\_ACKNODLY、その他の値はTCP/IPマネージャのデフォルト値と同一となります。

Telnetサーバ動作モードはsvpidに0を指定した場合のみ設定可能です。(すでに接続されているTelnetdサービスポイントには影響しません。)

(2) modname = TLD\_MOD\_CBK (コールバックイベントの設定)

Telnetサーバがコールバックにより通知するイベント (一部) の有効/無効を指定します。デフォルト設定では、cbkmode = 0 (すべて無効) となっています。

コールバックイベントの設定はsvpidに0を指定した場合のみ設定可能です。(すでに接続されているTelnetdサービスポイントには影響しません。)

コールバックルーチンの詳細に関しては「4 コールバック」の項を参照してください。

modvalに指定するパケットの構造:

```
typedef struct{
    UW          cbkmode;          コールバック通知オプション
}T_TLD_MODCBK;
```

コールバック通知オプション(cbkmode)には、次の値を設定します。

ビット1：データ受信イベント

TLD_CBK_RCVOFF (H'00000000)	データ受信イベント通知無し
TLD_CBK_RCVON (H'00000001)	データ受信イベント通知有り

ビット2：送信バッファ空きイベント

TLD_CBK_SNDOFF (H'00000000)	送信バッファ空きイベント通知無し
TLD_CBK_SNDON (H'00000002)	送信バッファ空きイベント通知有り

### (3) modname = TLD\_MOD\_ASC (Telnetサーバテキスト形式の設定)

サーバ側のテキスト形式を設定します。デフォルトではTLD\_ASC\_CRLFに設定されています。

クライアントから送られたデータは、サーバのテキスト形式に自動変換されます。また、クライアントに送られるデータはネットワークの標準形式に自動変換されます。

modvalに指定するパケットの構造：

```
typedef struct{
    UW          ascmode;          テキスト形式
}T_TLD_MODASC
```

テキスト形式(ascmode)には、次の値が指定できます。

TLD_ASC_CRLF (H'00000000)	改行文字はCR+LF
TLD_ASC_LF (H'00000004)	改行文字はLF

### (4) modname = TLD\_MOD\_OPT (Telnetオプションの設定)

サーバ側のTelnetオプションの有効/無効を設定します。

クライアントからのTelnetオプションに対する要求があった場合、有効に設定したオプションに関しては許可する応答を送信し、必要によりコールバックイベントを通知します。無効に設定したオプションに関しては拒否の応答を送信します。

Telnetオプションの設定はsvpidに0を指定した場合のみ設定可能です。(すでに接続されているTelnetdサーバポイントには影響しません。)

以下に示すパケットの各Telnetオプションには、次の値が指定できます。

- 0：サーバのTelnetオプションを無効にする
- 1：サーバのTelnetオプションを有効にする

modvalに指定するパケットの構造

```
typedef struct{
    UB          neg_options[40];
}T_TLD_MODALOPT;
```

neg_options[0]	TRANSMIT-BINARY (バイナリ転送)
neg_options[1]	ECHO (エコーバック)
neg_options[2]	RCP (再接続) ※ <sup>1</sup>
neg_options[3]	SUPPRESS-GA (送受信切り替え抑制)
neg_options[4]	NAMS (メッセージ長の調停) ※ <sup>1</sup>
neg_options[5]	STATUS (Telnetオプション状態通知)
neg_options[6]	TIMING-MARK (タイミングマーク)
neg_options[7]	RCTE (送信&エコーのリモート制御) ※ <sup>1</sup>
neg_options[8]	NAOL (行幅出力の調停) ※ <sup>1</sup>
neg_options[9]	NAOP (ページサイズ出力の調停) ※ <sup>1</sup>
neg_options[10]	NAOCR (キャリッジリターン出力の調停) ※ <sup>1</sup>
neg_options[11]	NAOHTS (水平タブストップ出力の調停) ※ <sup>1</sup>
neg_options[12]	NAOHTD (水平タブキャラクタ処理の調停) ※ <sup>1</sup>
neg_options[13]	NAOFFD (フォームフィード出力処理の調停) ※ <sup>1</sup>
neg_options[14]	NAOVTS (垂直タブストップ出力の調停) ※ <sup>1</sup>
neg_options[15]	NAOVTD (垂直タブ出力処理の調停) ※ <sup>1</sup>

neg_options[16]	NAOLFD (ラインフィード出力の調停) ※1
neg_options[17]	EXTEND-ASCII (拡張ASCII) ※1
neg_options[18]	LOGOUT (強制ログアウト)
neg_options[19]	BM (バイトマクロ) ※1
neg_options[20]	DET (データエントリ端末) ※1
neg_options[21]	SUPDUP (SUPDUPディスプレイプロトコル) ※1
neg_options[22]	SUPDUP-OUTPUT (SUPDUP出力) ※1
neg_options[23]	SEND-LOCATION (ロケーションID送信) ※1
neg_options[24]	TERMINAL-TYPE (ターミナルタイプ)
neg_options[25]	END-OF-RECORD (エンドオブレコード) ※1
neg_options[26]	TUID (TACACSユーザ識別) ※1
neg_options[27]	OUTMRK (マーク出力) ※1
neg_options[28]	TTYLOC (端末ロケーション番号) ※1
neg_options[29]	3270-REGIME (TELNET3270様式) ※1
neg_options[30]	X3-PAD (X.3 PAD) ※1
neg_options[31]	NAWS (ウィンドウサイズの調停)
neg_options[32]	TERMINAL-SPEED (端末スピード)
neg_options[33]	TOGGLE-FLOW-CONTROL (フロー制御)
neg_options[34]	LINEMODE (ラインモード)
neg_options[35]	XDISPLOC (Xディスプレイロケーション) ※1
neg_options[36]	ENVIRON (環境変数) ※1
neg_options[37]	AUTHENTICATION (認証) ※1
neg_options[38]	ENCRYPTION (暗号化) ※1
neg_options[39]	NEW-ENVIRON (新環境変数) ※1

(※1) : 本バージョンでは未サポートのため、0固定

Telnetオプションの詳細に関しては「3.2.9 Telnetオプションを送信する」の項を参照してください。

### 3.2.8 TLd\_get\_mod Telnet サーバの動作モードを取得する

【T】

C 言語インタフェース

```
ER_UINT modlen = TLd_get_mod ( ID svpid , INT modname , VP modval );
```

パラメータ

ID	svpid	TelnetdサービスポイントID
INT	modname	取得する動作モードの種類
VP	modval	動作モードを格納する領域の先頭アドレス

リターンパラメータ

ER_UINT	modlen	リターン値またはエラーコード
---------	--------	----------------

リターン値/エラーコード

正の値	正常終了 (取得した動作モードのデータ長)
E_PAR	パラメータエラー (modnameが不正、modvalが0、または4の倍数以外)
E_ID	不正ID番号 (TelnetdサービスポイントIDは無効)
E_OBJ	オブジェクト状態エラー (サービスが開始していない、または接続状態でない)

解 説

Telnetサーバのサービス、または各Telnetdサービスポイントの現在の動作モードを取得します。現在の動作モードを取得できた場合、リターン値として取得した動作モードのデータ長を返します。

svpidには取得するTelnetdサービスポイントIDを指定します。

modnameには、TLD\_MOD\_OPT (Telnetオプションの取得) のみ指定できます。

指定したパラメータが不正の場合は、エラーコードとしてE\_PARを返します。

無効なTelnetdサービスポイントIDを指定した場合は、エラーコードとしてE\_IDを返します。

(1) optname = TLD\_MOD\_OPT (Telnetオプションの取得)

サーバの現在のTelnetオプションの状態を取得します。

以下に示すパケットの各Telnetオプションには、次の値が返されます。

- 0 : サーバ/クライアント共に無効
- 1 : サーバ側のみ有効
- 2 : クライアント側のみ有効
- 3 : サーバ/クライアント共に有効

modvalに指定するパケットの構造

```
typedef struct{
    UB      neg_options[40];
}T_TLD_MODOPT;
```

neg_options[0]	TRANSMIT-BINARY (バイナリ転送)
neg_options[1]	ECHO (エコーバック)
neg_options[2]	RCP (再接続) ※1
neg_options[3]	SUPPRESS-GA (送受信切り替え抑制)
neg_options[4]	NAMS (メッセージ長の調停) ※1
neg_options[5]	STATUS (Telnetオプション状態通知)
neg_options[6]	TIMING-MARK (タイミグマーク)
neg_options[7]	RCTE (送信&エコーのリモート制御) ※1
neg_options[8]	NAOL (行幅出力の調停) ※1
neg_options[9]	NAOP (ページサイズ出力の調停) ※1
neg_options[10]	NAOCD (キャリッジリターン出力の調停) ※1

neg_options[11]	NAOHTS (水平タブストップ出力の調停) ※1
neg_options[12]	NAOHTD (水平タブキャラクタ処理の調停) ※1
neg_options[13]	NAOFFD (フォームフィード出力処理の調停) ※1
neg_options[14]	NAOVTS (垂直タブストップ出力の調停) ※1
neg_options[15]	NAOVTD (垂直タブ出力処理の調停) ※1
neg_options[16]	NAOLFD (ラインフィード出力の調停) ※1
neg_options[17]	EXTEND-ASCII (拡張ASCII) ※1
neg_options[18]	LOGOUT (強制ログアウト)
neg_options[19]	BM (バイトマクロ) ※1
neg_options[20]	DET (データエントリ端末) ※1
neg_options[21]	SUPDUP (SUPDUPディスプレイプロトコル) ※1
neg_options[22]	SUPDUP-OUTPUT (SUPDUP出力) ※1
neg_options[23]	SEND-LOCATION (ロケーションID送信) ※1
neg_options[24]	TERMINAL-TYPE (ターミナルタイプ)
neg_options[25]	END-OF-RECORD (エンドオブレコード) ※1
neg_options[26]	TUID (TACACSユーザ識別) ※1
neg_options[27]	OUTMRK (マーク出力) ※1
neg_options[28]	TTYLOC (端末ロケーション番号) ※1
neg_options[29]	3270-REGIME (TELNET3270様式) ※1
neg_options[30]	X3-PAD (X.3 PAD) ※1
neg_options[31]	NAWS (ウィンドウサイズの調停)
neg_options[32]	TERMINAL-SPEED (端末スピード)
neg_options[33]	TOGGLE-FLOW-CONTROL (フロー制御)
neg_options[34]	LINEMODE (ラインモード)
neg_options[35]	XDISPLOC (Xディスプレイロケーション) ※1
neg_options[36]	ENVIRON (環境変数) ※1
neg_options[37]	AUTHENTICATION (認証) ※1
neg_options[38]	ENCRYPTION (暗号化) ※1
neg_options[39]	NEW-ENVIRON (新環境変数) ※1

(※1) : 本バージョンでは未サポート

Telnetオプションの詳細に関しては「3.2.9 Telnetオプションを送信する」の項を参照してください。

### 3.2.9 TLd\_isu\_opt Telnet オプションを送信する

【T】

C 言語インタフェース

```
ER ercd = TLd_isu_opt ( ID svpid , INT optname , VP optval , TMO tmout );
```

パラメータ

ID	svpid	TelnetdサービスポイントID
INT	optname	送信するTelnetオプション番号
VP	optval	設定内容を格納した領域の先頭アドレス
TMO	tmout	タイムアウト値

リターンパラメータ

ER	ercd	リターン値またはエラーコード
----	------	----------------

リターン値/エラーコード

E_OK	正常終了 (クライアントからの応答)
E_NOSPT	未サポート
E_PAR	パラメータエラー (optnameがマイナス、または256以上、tmout < -1、optvalが不正)
E_ID	不正ID番号 (TelnetdサービスポイントIDは無効)
E_OBJ	オブジェクト状態エラー (サービスが開始していない、または接続状態でない、またはすでに送信済みか送信する必要がない)
E_TMOUT	ポーリング失敗またはタイムアウト
E_RLWAI	処理のキャンセル、待ち状態の強制解除
E_CLS	接続切断
EV_PROT	クライアントが要求を拒否

解 説

指定されたTelnetオプションの要求をクライアントに送信します。

svpidには対象となるにTelnetdサービスポイントIDを指定します。

optnameには送信するTelnetオプションの番号を指定します。

optvalには送信するTelnetオプションの要求内容を指定します。0 (WILL: サーバ側に対してオプションを有効にしたい)、1(DO: クライアント側に対してオプションを有効にしたい)、2(WONT: サーバ側に対してオプションを無効にしたい)、3(DONT: クライアント側に対してオプションを無効にしたい)のみ指定できます。

tmoutにはクライアントからの応答待ち時間を指定します。tmoutに正の値を指定した場合は待ち時間 (単位はms)、TMO\_POL(0)を指定した場合はポーリング、TMO\_FEVR(-1)を指定した場合は永久待ちとなります。ただし、本サービスコールはクライアントからの応答を待つため、必ず待ち状態になります。そのため、ポーリング指定では処理を行わず、直ちにエラーコードとしてE\_TMOUTを返します。

送信したTelnetオプションの要求に対して、それをクライアントが承諾した場合、リターン値としてE\_OKを返します。クライアントが要求に対してそれを拒否した場合、リターン値としてEV\_PROTを返します。

指定したパラメータが不正の場合は、エラーコードとしてE\_PARを返します。

未サポートのTelnetオプション番号を指定した場合は、E\_NOSPTを返します。

接続状態以外のサービスポイントに対して本サービスコールを発行した場合、またはすでに指定した状態となっていてTelnetオプションを送信する必要がない場合、リターン値としてE\_OBJを返します。

Telnetオプションによっては、追加データを受信する場合があります。その内容はcallbackルーチンによって通知されます。コールバックルーチンの詳細に関しては、「4 コールバック」の項を参照してください。

送信できるTelnetオプションを次に示します。

#### Telnetオプション一覧

オプション番号	説明
TLOPT_BINARY (0)	通常の7ビットデータではなく、8ビットバイナリとしてデータを受信する
TLOPT_ECHO (1)	エコーバックを行う
TLOPT_SGA (3)	送受信を切り替えるGO AHEADコマンドの送信を抑制する
TLOPT_STATUS (5)	Telnetオプション状態を送信する
TLOPT_TM (6)	コネクションの双方の同期を取る際に使用される
TLOPT_TTYPE (24)	端末タイプを送信する (クライアント側のみに対して有効)
TLOPT_NAWS (31)	端末ウィンドウの行と列の数を送る (クライアント側のみに対して有効)
TLOPT_TSPEED (32)	端末の送信速度と受信速度を送る (クライアント側のみに対して有効)
TLOPT_LFLOW (33)	フロー制御を行う
TLOPT_LINEMODE (34)	リアルラインモードにてデータを行単位で送る

指定したパラメータが不正の場合は、エラーコードとしてE\_PARを返します。

無効なTelnetサービスポイントIDを指定した場合は、エラーコードとしてE\_IDを返します。

未サポートのTelnetオプション番号を指定した場合は、エラーコードとしてE\_NOSPTを返します。

TCPコネクションが切断された場合は、エラーコードとしてE\_CLSを返します。この場合、受信データがあればTLd\_get\_datにてデータを取得することができますが、TLd\_put\_datにてデータを送信することはできません。

(1) TLOPT\_BINARY バイナリ転送オプション

Telnetオプション番号 (optname)

TLOPT\_BINARY (0)

パラメータブロックの構造 (optval)

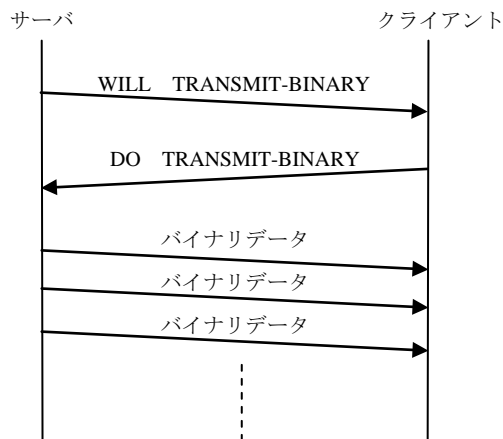
```
typedef struct{
    UW      flag;          バイナリ転送の要求フラグ
} T_TLOPT_COMMON;
```

解 説

クライアントにTelnetオプションのTRANSMIT-BINARY (バイナリ転送) を送信し、バイナリ転送を有効または無効にします。バイナリ転送が有効となると、Telnetコマンドコードが無視され、そのままデータとして送信されます。

optval->flagには送信するTelnetオプションの要求内容を指定します。次の値のみ指定可能です。

- 0 (WILL) : サーバ側がバイナリデータ送信を行うことを通知する。
- 1 (DO) : クライアント側にバイナリデータ送信を行うように要求する。
- 2 (WONT) : サーバ側がバイナリデータ送信を行わないことを通知する。
- 3 (DONT) : クライアント側にバイナリデータ送信を行わないように要求する。





## (2) TLOPT\_ECHO エコーバックオプション

Telnetオプション番号 (optname)

TLOPT\_ECHO (1)

パラメータブロックの構造 (optval)

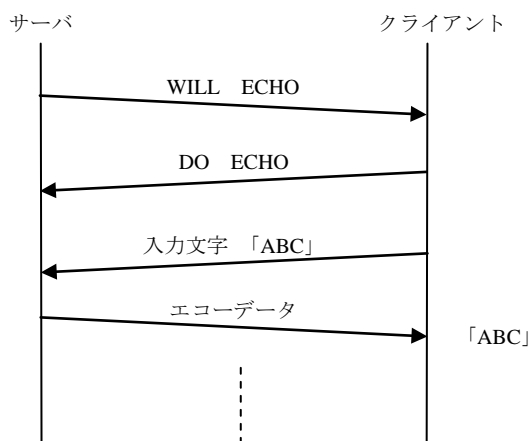
```
typedef struct{
    UW      flag;          エコーバックの要求フラグ
} T_TLOPT_COMMON;
```

### 解 説

クライアントにTelnetオプションのECHO (エコーバック) を送信し、エコーバックを有効または無効にします。

optval->flagには送信するTelnetオプションの要求内容を指定します。次の値のみ指定可能です。

- 0 (WILL) : サーバ側がエコーバックを行うことを通知する。
- 1 (DO) : クライアント側にエコーバックを行うように要求する。
- 2 (WONT) : サーバ側がエコーバックを行わないことを通知する。
- 3 (DONT) : クライアント側にエコーバックを行わないように要求する。



### (3) TLOPT\_SGA GO AHEAD 送信抑制オプション

Telnetオプション番号 (optname)

TLOPT\_SGA (3)

パラメータブロックの構造 (optval)

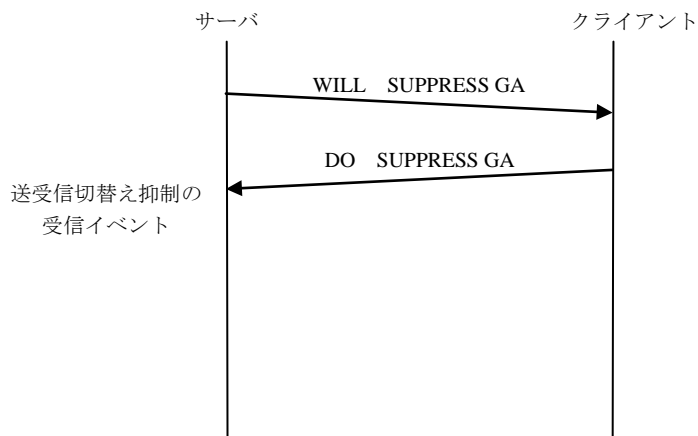
```
typedef struct{
    UW      flag;          GO AHEAD送信抑制の要求フラグ
} T_TLOPT_COMMON;
```

#### 解 説

クライアントにTelnetオプションのSUPPRESS-GA (送受信切り替え抑制) を送信し、送受信を切り替えるGO AHEADコマンド送信の抑制を有効または無効にします。

optval->flagには送信するTelnetオプションの要求内容を指定します。次の値のみ指定可能です。

- 0 (WILL) : サーバ側がGO AHEADコマンドの抑制を行うことを通知する。
- 1 (DO) : クライアント側にGO AHEADコマンドを抑制するように要求する。
- 2 (WONT) : サーバ側がGO AHEADコマンドを抑制を行わないことを通知する。
- 3 (DONT) : クライアント側にGO AHEADコマンドを抑制しないように要求する。



(4) TLOPT\_STATUS ステータス要求オプション

Telnetオプション番号 (optname)

TLOPT\_STATUS (5)

パラメータブロックの構造 (optval)

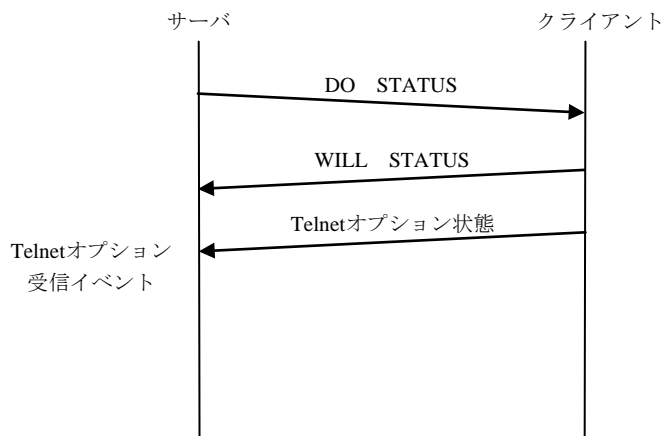
```
typedef struct{
    UW      flag;          ステータス送信の要求フラグ
} T_TLOPT_COMMON;
```

解 説

クライアントにTelnetオプションのSTATUS (Telnetオプションの状態通知) を送信し、Telnetオプションの状態通知を有効または無効にします。

optval->flagには送信するTelnetオプションの要求内容を指定します。次の値のみ指定可能です。

- 0 (WILL) : サーバ側がTelnetオプション状態の送信を行うことを通知する。
- 1 (DO) : クライアント側にTelnetオプション状態の送信を行うように要求する。
- 2 (WONT) : サーバ側がTelnetオプション状態の送信を行わないことを通知する。
- 3 (DONT) : クライアント側にTelnetオプション状態の送信を行わないように要求する。



(5) TLOPT\_TM タイミングマークオプション

Telnetオプション番号 (optname)

TLOPT\_TM (6)

パラメータブロックの構造 (optval)

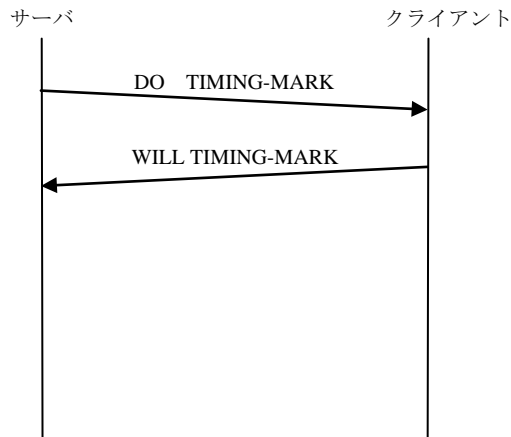
```
typedef struct{
    UW      flag;          タイミングマークの要求フラグ
} T_TLOPT_COMMON;
```

解 説

コネクション双方の同期をとるために、クライアントにTelnetオプションのTIMING MARK (タイミングマーク) をTCP緊急データで送信します。

optval->flagには送信するTelnetオプションの要求内容を指定します。次の値のみ指定可能です。

- 1 (DO) : クライアント側にTIMING MARKを送信する。
- 2 (WONT) : サーバ側が同期のためのTIMING MARKを送らないことを通知する。



(6) TLOPT\_TTYPE                    ターミナルタイプオプション

Telnetオプション番号 (optname)

TLOPT\_TTYPE (24)

パラメータブロックの構造 (optval)

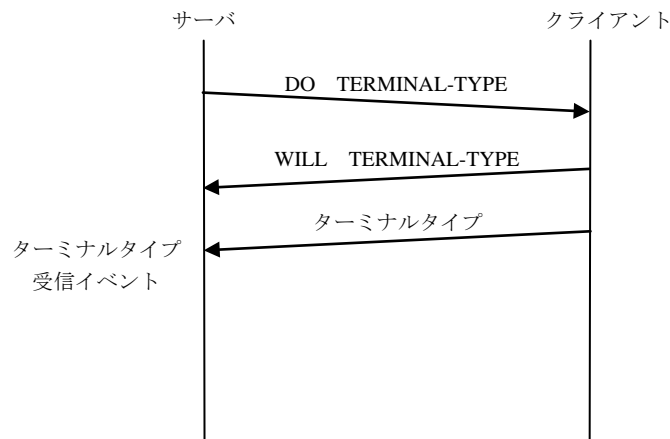
```
typedef struct{
    UW      flag;           ターミナルタイプの要求フラグ
} T_TLOPT_COMMON;
```

解 説

クライアントにTelnetオプションのTERMINAL TYPE (ターミナルタイプ) を送信し、ターミナルタイプの送信を有効または無効にします。

optval->flagには送信するTelnetオプションの要求内容を指定します。次の値のみ指定可能です。

- 1 (DO)        : クライアントにターミナルタイプを送信するように要求する。
- 3 (DONT)     : クライアントにターミナルタイプを送信しないように要求する。



(7) TLOPT\_NAWS ウィンドウサイズオプション

Telnetオプション番号 (optname)

TLOPT\_NAWS (31)

パラメータブロックの構造 (optval)

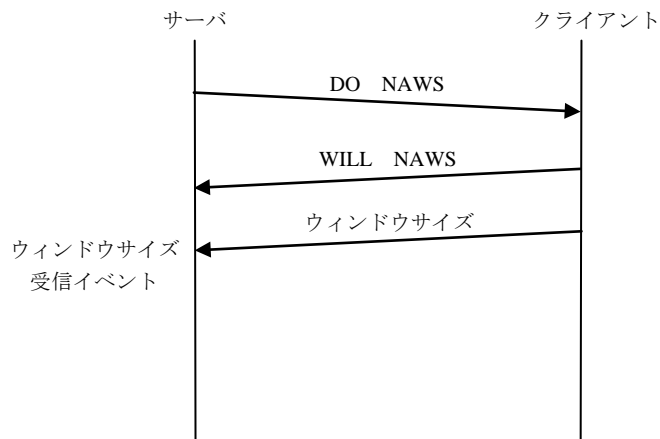
```
typedef struct{
    UW      flag;          ウィンドウサイズの要求フラグ
} T_TLOPT_COMMON;
```

解 説

クライアントにTelnetオプションのNAWS (ウィンドウサイズ) を送信し、ターミナルウィンドウサイズの送信を有効または無効にします。

optval->flagには送信するTelnetオプションの要求内容を指定します。次の値のみ指定可能です。

- 1 (DO) : クライアントにターミナルウィンドウサイズを送信するように要求する。
- 3 (DONT) : クライアントにターミナルウィンドウサイズを送信しないように要求する。



(8) TLOPT\_TSPEED

ターミナル速度オプション

Telnetオプション番号 (optname)

TLOPT\_TSPEED (32)

パラメータブロックの構造 (optval)

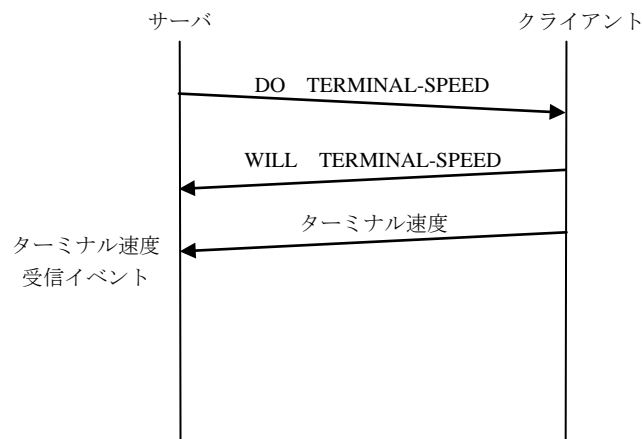
```
typedef struct{
    UW      flag;          ターミナル速度の要求フラグ
} T_TLOPT_COMMON;
```

解 説

クライアントにTelnetオプションのTERMINAL SPEED (ターミナル速度) を送信し、ターミナル速度の送信を有効または無効にします。

optval->flagには送信するTelnetオプションの要求内容を指定します。次の値のみ指定可能です。

- 1 (DO) : クライアントにターミナル速度を送信するように要求する。
- 3 (DONT) : クライアントにターミナル速度を送信しないように要求する。



(9) TLOPT\_LFLOW

リモートフロー制御オプション

Telnetオプション番号 (optname)

TLOPT\_LFLOW (33)

パラメータブロックの構造 (optval)

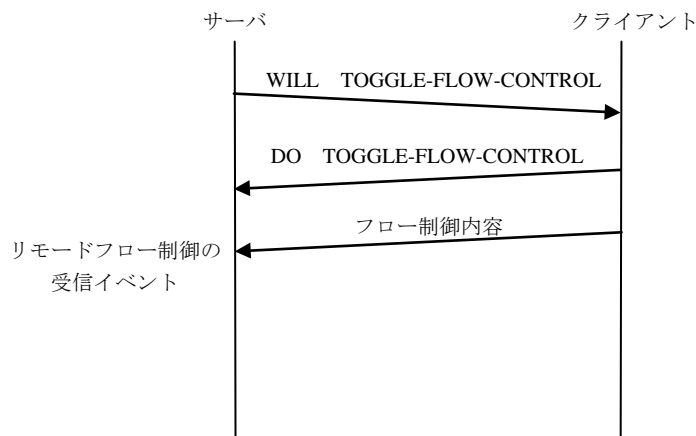
```
typedef struct{
    UW      flag;          リモートフロー制御の要求フラグ
} T_TLOPT_COMMON;
```

解 説

クライアントにTelnetオプションのTOGGLE-FLOW-CONTROL (リモートフロー制御) を送信し、フロー制御コマンドの送信を有効または無効にします。

optval->flagには送信するTelnetオプションの要求内容を指定します。次の値のみ指定可能です。

- 0 (WILL) : サーバ側がフロー制御コマンドの送信を行うことを通知する。
- 1 (DO) : クライアント側にフロー制御コマンドの送信を行うように要求する。
- 2 (WONT) : サーバ側がフロー制御コマンドの送信を行わないことを通知する。
- 3 (DONT) : クライアント側にフロー制御コマンドの送信を行わないように要求する。





(10) TLOPT\_LINEMODE

ラインモードオプション

Telnetオプション番号 (optname)

TLOPT\_LINEMODE (34)

パラメータブロックの構造 (optval)

```

typedef struct{
    UW      flag;           ラインモードの要求フラグ
    UW      fncd;          要求するラインモードの機能コード
    VP      lmpar;         パラメータブロックへの先頭アドレス
} T_TLOPT_LINEMODE;

```

解 説

クライアントにTelnetオプションのLINEMODE (ラインモード) を送信し、編集/シグナルステータスの送信を有効または無効にします。

optval->flagには送信するTelnetオプションの要求内容を指定します。次の値のみ指定可能です。

- 1 (DO) : クライアントにLINEMODEを送信するように要求する。
- 3 (DONT) : クライアントにLINEMODEを送信しないように要求する。

optval->fncdには、要求するLINEMODEの機能を指定します。flagに1 (DO) がセットされている場合のみ有効で、次の値のみ指定可能です。

- 1 (FUNC\_MODE) : LINEMODEのモード指定
- 2 (FUNC\_SLC) : 特殊文字の定義

(1) FUNC\_MODE (LINEMODEのモード指定)

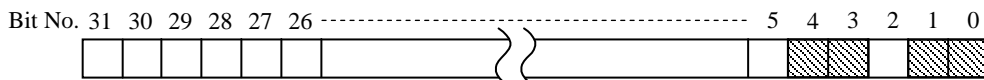
lmparパラメータブロックの構造 :

```

typedef struct{
    UW      modemask;      モード
} T_TLOPT_LMMODE;

```

LINEMODEのモードを指定します。flagに1 (DO) がセットされている場合のみ有効で、modemaskには次の値のみ指定可能です。



- bit0 (MODE\_EDIT) : このビットがセットされる場合、クライアント側にて全ての編集処理を行い、完全な行のみをサーバに送ります。このビットがセットされない場合、クライアント側は、ユーザからの入力を何も処理せずにそのままサーバに送ります。編集処理はサーバ側が行います。
- bit1 (MODE\_TRAPSIG) : このビットがセットされる場合、クライアントは IP、BRK、ATY、ABORT、EOF、SUSP等の割り込み/シグナルを、Telnetオプションに変換してサーバに送ります。このビットがセットされない場合、クライアント側は割り込み/シグナルをTelnetコマンドにてサーバに送ります。
- bit3 (MODE\_SOFTTAB) : このビットがセットされる場合、クライアントは水平タブコード (ASCII : 0x09) を次の水平タブストップまでの適当なスペースに変換してサーバに送ります。このビットがセットされない場合、クライアントは水平タブコードを処理しないでサーバに送ります。
- bit4 (MODE\_LIT\_ECHO) : このビットがセットされる場合、クライアント側がユーザがタイプした印刷不可文字をエコーする場合は、そのままの文字をエコーします。このビットがセットされない場合は、クライアント側が希望する方法で文字をエコーします。(例えば、ASCII 0x01は「^A」をエコーします。)

(2) FUNC\_SLC (特殊文字の定義)

lmparパラメータブロックの構造:

```
typedef struct{
    UW          slcnum;   定義する特殊文字の数
    T_TLOPT_SLC *slc;    定義する特殊文字配列の先頭アドレス
} T_TLOPT_LMSLC;

typedef struct {
    UW  slcfunc;         機能
    UB  slcflag;        機能の修飾子
    UB  slcval;         機能を割り当てる文字コード
} T_TLOPT_SLC;
```

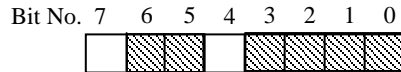
特定の文字コードに機能を割り当てる場合に定義します。特殊文字の定義は、LINEMODEを無効から有効にする場合のみ有効です。指定しない機能には以前の値が使用されます。

lmpar->slcnumには定義する特殊文字の数を設定します。最大30まで指定できます。

lmpar->slc[].slcfuncには機能を指定します。次の値のみ指定可能です。

- 1 (SLC\_SYNCH) : 同期
- 2 (SLC\_BRK) : ブレーク
- 3 (SLC\_IP) : プロセス割り込み
- 4 (SLC\_AO) : 出力停止
- 5 (SLC\_AYT) : 相手存在確認
- 6 (SLC\_EOR) : レコード終端
- 7 (SLC\_ABORT) : プロセス中断
- 8 (SLC\_EOF) : エンド・オブ・ファイル
- 9 (SLC\_SUSP) : プロセスの一時停止
- 10 (SLC\_EC) : 文字消去。入力ストリームから1文字を消去する
- 11 (SLC\_EL) : 行消去。現在の入力行全体を消去する
- 12 (SLC\_EW) : 単語消去。入力ストリームから1単語を消去する
- 13 (SLC\_RP) : 行の再印字
- 14 (SLC\_LNEXT) : 次の文字は処理せず、そのまま送信する
- 15 (SLC\_XON) : 出力開始。ユーザ端末への出力を再開する
- 16 (SLC\_XOFF) : 出力停止。ユーザ端末への出力を停止する
- 17 (SLC\_FORW1) : 送出キャラクタ1。すべてのバッファされたデータと現在の文字を送る
- 18 (SLC\_FORW2) : 送出キャラクタ2。SLC\_FORW1と同様に扱われる別の文字
- 19 (SLC\_MCL) : 1文字左にカーソルを移動する
- 20 (SLC\_MCR) : 1文字右にカーソルを移動する
- 21 (SLC\_MCWL) : 1単語左にカーソル移動する
- 22 (SLC\_MCWR) : 1単語右にカーソル移動する
- 23 (SLC\_MCBOL) : 行の開始にカーソルを移動する
- 24 (SLC\_MCEOL) : 行の終わりにカーソルを移動する
- 25 (SLC\_INSRT) : 挿入モード
- 26 (SLC\_OVER) : 上書きモード
- 27 (SLC\_ECR) : 右側文字の消去
- 28 (SLC\_EWR) : 右側単語の消去
- 29 (SLC\_EBOL) : 行の開始まで消去
- 30 (SLC\_EEOL) : 行の終端まで消去

Impar->slc[].slcflagには、機能の修飾子を指定します。次の値のみ指定可能です。（bit0からbit3の優先度は、bit0が最も低く、bit3が最も高い）



bit0~3=0 (SLC\_NOSUPPORT) : この機能は使用されません。

bit0~3=1 (SLC\_CANTCHANGE) : この機能を使用します。しかし文字コードは変更できません。

bit0~3=2 (SLC\_VALUE) : この機能を使用します。現在の文字コードの値はslcvalに指定されます。

bit0~3=3 (SLC\_DEFAULT) : この機能は相手側のデフォルトを使用します。

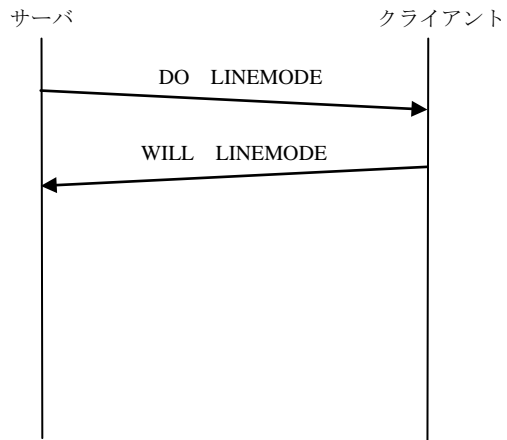
bit5 (SLC\_FLUSHOUT) : このビットがセットされる場合、この機能が送信される際、入力データをフラッシュするために一緒に同期のためのコマンドを送信します。

bit6 (SLC\_FLUSHIN) : このビットがセットされる場合、この機能が送信される際、出力データをフラッシュします。

Impar->slc[].slcvalには、機能を割り当てるASCIIコード (0~255) を指定します。

Telnetサーバの特殊文字のデフォルト値を以下に示します。

機能	修飾子	ASCIIコード
1 (SLC_SYNCH)	SLC_DEFAULT	—
2 (SLC_BRK)	SLC_DEFAULT	—
3 (SLC_IP)	SLC_VALUE   SLC_FLUSHIN   SLC_FLUSHOUT	0x03
4 (SLC_AO)	SLC_VALUE   SLC_FLUSHOUT	0x0f
5 (SLC_AYT)	SLC_DEFAULT	—
6 (SLC_EOR)	SLC_DEFAULT	—
7 (SLC_ABORT)	SLC_VALUE   SLC_FLUSHIN   SLC_FLUSHOUT	0x1c
8 (SLC_EOF)	SLC_VALUE	0x04
9 (SLC_SUSP)	SLC_VALUE   SLC_FLUSHIN	0x1a
10 (SLC_EC)	SLC_VALUE	0x7f
11 (SLC_EL)	SLC_VALUE	0x15
12 (SLC_EW)	SLC_VALUE	0x17
13 (SLC_RP)	SLC_VALUE	0x12
14 (SLC_LNEXT)	SLC_VALUE	0x16
15 (SLC_XON)	SLC_VALUE	0x11
16 (SLC_XOFF)	SLC_VALUE	0x13
17 (SLC_FORW1)	SLC_VALUE	0x00
18 (SLC_FORW2)	SLC_VALUE	0x00
19 (SLC_MCL)	SLC_NOSUPPORT	—
20 (SLC_MCR)	SLC_NOSUPPORT	—
21 (SLC_MCWL)	SLC_NOSUPPORT	—
22 (SLC_MCWR)	SLC_NOSUPPORT	—
23 (SLC_MCBOL)	SLC_NOSUPPORT	—
24 (SLC_MCEOL)	SLC_NOSUPPORT	—
25 (SLC_INSRT)	SLC_NOSUPPORT	—
26 (SLC_OVER)	SLC_NOSUPPORT	—
27 (SLC_ECR)	SLC_NOSUPPORT	—
28 (SLC_EWR)	SLC_NOSUPPORT	—
29 (SLC_EBOL)	SLC_NOSUPPORT	—
30 (SLC_EEOL)	SLC_NOSUPPORT	—



### 3.2.10 TLd\_can\_svp                      ペンディングしている処理をキャンセルする

【T】

C言語インタフェース

```
ER ercd = TLd_can_svp(ID svpid, FN fnccd);
```

パラメータ

ID	svpid	TelnetdサービスポイントID
FN	fnccd	キャンセルするサービスコールの機能コード

リターンパラメータ

ER	ercd	リターン値またはエラーコード
----	------	----------------

リターン値/エラーコード

E_OK	正常終了
E_PAR	パラメータエラー (fnccdが指定可能な機能コードでない)
E_ID	不正ID番号 (TelnetdサービスポイントIDは無効)
E_OBJ	オブジェクト状態エラー (指定したTelnetdサービスポイントIDにfnccdで指定した処理がペンディングしていない)

解説

svpidで示されたTelnetdサービスポイントに対し、fnccdで示されたペンディング中の処理をキャンセルします。

キャンセルされたタスクには、エラーコードとしてE\_RLWAIを返します。

キャンセル可能な処理のサービスコール名と、それを指定する機能コードは次の通りです。また、TFN\_TLD\_ALL(0)を指定すると、指定したTelnetdサービスポイントにペンディングしているすべての処理をキャンセルすることができます。

サービスコール名	機能コード
TLd_sht_srv	TFN_TLD_SHT_SRV (-0x1604)
TLd_put_dat	TFN_TLD_PUT_DAT(-0x1605)
TLd_get_dat	TFN_TLD_GET_DAT(-0x1606)
TLd_isu_opt	TFN_TLD_ISU_OPT(-0x1608)
すべて	TFN_TLD_ALL(0)

---

## 4. コールバック

---

コールバックは、Telnetサーバ内で発生した事象をアプリケーションプログラムに伝えるために用いられます。コールバックルーチンはユーザが作成し、そのアドレスのみがTelnetサーバに渡されるため、モジュール名はユーザ任意となります。ここでは仮称として***callback***とします。

## 4.1 コールバックルーチン

### 4.1.1 *callback* Telnet サーバイベントを通知する

【T】

C 言語インタフェース

ER ercd = *callback* ( ID svpid , UW evcd , VP parblk );

パラメータ

ID	svpid;	TelnetdサービスポイントID
UW	evcd;	コールバックイベント番号
VP	parblk;	パラメータを格納した領域の先頭アドレス

リターンパラメータ

ER	ercd	リターン値またはエラーコード (未使用)
----	------	----------------------

解 説

Telnetサーバのイベントが発生した場合にコールされます。コールバックルーチンはTelnetサーバタスクレベルにてサブルーチンコールされます。

コールバックルーチン内にてTelnetクライアントのサービスコールは発行できません。コールバックルーチン内にてTelnetクライアントのサービスコールを発行した場合の動作は保証しません。

svpidにはイベントが発生したTelnetdサービスポイントIDが渡されます。

コールバックイベントには以下の種類があります。

#### コールバックイベント一覧

イベント種類	イベント名称	イベント番号	説明
システム イベント	TEV_TLD_CONN	0x1700	クライアントの接続
	TEV_TLD_DISCONN	0x1701	クライアントの接続切断
	TEV_TLD_RCVDAT	0x1702	データの受信
	TEV_TLD_SNDEMP	0x1703	送信バッファ空き
Telnetオプション 受信イベント	TEV_TLD_BINARY	0x1600	バイナリモードの変更を受信
	TEV_TLD_ECHO	0x1601	エコーバックの変更を受信
	TEV_TLD_SGA	0x1603	送受信切替の抑制を受信
	TEV_TLD_STATUS	0x1605	Telnetオプション状態を受信
	TEV_TLD_LOGOUT	0x1612	強制LOGOUTを受信
	TEV_TLD_TTYPE	0x1618	端末タイプ情報を受信
	TEV_TLD_NAWS	0x161F	ウィンドウサイズ情報を受信
	TEV_TLD_TSPEED	0x1620	端末スピード情報を受信
	TEV_TLD_LFLOW	0x1621	リモートフロー制御の変更を受信
	TEV_TLD_LINEMODE	0x1622	ラインモードの変更を受信
	TEV_TLD_EDIT	0x1704	編集モードの変更を受信
	TEV_TLD_SIGNAL	0x1705	割り込み/シグナルの変更を受信
	TEV_TLD_SOFTTAB	0x1706	ソフトタブの変更を受信
	TEV_TLD_LITECHO	0x1707	リテラルエコーの変更を受信
	TEV_TLD_SLC	0x1708	特殊文字の変更を受信

(1) TEV\_TLD\_CONN

Telnet クライアントの接続イベント

コールバックイベント番号 (evcd)

TEV\_TLD\_CONN (0x1700)

パラメータブロックの構造 (parblk)

```
typedef struct{
    UW    ipaddr;      TelnetクライアントのIPアドレス
    UH    portno;     Telnetクライアントのポート番号
} T_IPV4EP;
```

解 説

Telnetクライアントが接続されたことを通知します。

svpidには接続されたTelnetdサービスポイントIDが渡されます。

parblkには接続したTelnetクライアントのIPアドレス、ポート番号が渡されます。



(2) TEV\_TLD\_DISCONN

Telnet クライアントの接続切断イベント

コールバックイベント番号 (evcd)

TEV\_TLD\_DISCONN (0x1701)

パラメータブロックの構造 (parblk)

無し

解 説

Telnetサーバからクライアントが切断されたことを通知します。  
svpidには切断されたTelnetdサービスポイントIDが渡されます。

このイベントでのparblkは無効のため、0を渡します。

(3) TEV\_TLD\_RCVDAT                      データの受信イベント

コールバックイベント番号 (evcd)

TEV\_TLD\_RCVDAT (0x1702)

パラメータブロックの構造 (parblk)

```
typedef struct{
    INT      rtncd;          受信データの長さ
} T_TLDPB_RCVDAT;
```

解 説

クライアントからのデータを受信したことを通知します。  
svpidには受信したTelnetdサービスポイントIDが渡されます。  
本イベント通知は、Tld\_set\_modにてコールバックイベント通知の設定を行った場合のみ発生します。また、Tld\_rcv\_datがペンディング状態の場合は発生しません。

コールバックルーチン内にてTld\_rcv\_datをコールし、受信データを取得することも可能です。

(4) TEV\_TLD\_SNDEMP 送信バッファの空きイベント

コールバックイベント番号 (evcd)

TEV\_TLD\_SNDEMP (0x1703)

パラメータブロックの構造 (parblk)

```
typedef struct{
    INT      rtncd;          空き送信バッファのサイズ
} T_TLDPB_SNDEMP;
```

解 説

送信バッファに空きが発生したことを通知します。

svpidには空きが発生したTelnetdサービスポイントIDが渡されます。

本イベント通知は、Tld\_set\_modにてコールバックイベント通知の設定を行った場合のみ発生します。また、Tld\_snd\_datがペンディング状態の場合は発生しません。

コールバックルーチン内にてTld\_snd\_datをコールし、データを送信することも可能です。

(5) TEV\_TLD\_BINARY

バイナリモード要求の受信イベント

コールバックイベント番号 (evcd)

TEV\_TLD\_BINARY (0x1600)

パラメータブロックの構造 (parblk)

```
typedef struct{
    UW      flag;          バイナリモード要求フラグ
} T_TLDPB_COMMON;
```

解 説

クライアントから、バイナリモードの要求を受信したことを通知します。  
svpidには受信したTelnetdサービスポイントIDが渡されます。  
parblk->flagには次の値が渡されます。

- 0 (WILL) : クライアント側のバイナリデータ送信の実行要求を受信した。
- 1 (DO) : サーバ側のバイナリデータ送信の実行要求を受信した。
- 2 (WONT) : クライアント側のバイナリデータ送信の禁止要求を受信した。
- 3 (DONT) : サーバ側のバイナリデータ送信の禁止要求を受信した。

(6) TEV\_TLD\_ECHO

エコーバック要求の受信イベント

コールバックイベント番号 (evcd)

TEV\_TLD\_ECHO (0x1601)

パラメータブロックの構造 (parblk)

```
typedef struct{
    UW      flag;           エコーバック要求フラグ
} T_TLDPB_COMMON;
```

解 説

クライアントから、エコーバック要求を受信したことを通知します。TelnetオプションのLINEMODEが無効の場合に通知されます。

svpidには受信したTelnetdサービスポイントIDが渡されます。

parblk->flagには次の値が渡されます。

- 0 (WILL) : クライアント側のエコーバックの実行要求を受信した。
- 1 (DO) : サーバ側のエコーバックの実行要求を受信した。
- 2 (WONT) : クライアント側のエコーバックの禁止要求を受信した。
- 3 (DONT) : サーバ側のエコーバックの禁止要求を受信した。

(7) TEV\_TLD\_SGA 送受信切り替え抑制の受信イベント

コールバックイベント番号 (evcd)

TEV\_TLD\_SGA (0x1603)

パラメータブロックの構造 (parblk)

```
typedef struct{
    UW      flag;          送受信切り替え抑制要求フラグ
} T_TLDPB_COMMON;
```

解 説

クライアントから、送受信切り替え抑制 (Suppress-Go-Ahead) の要求を受信したことを通知します。  
svpidには受信したTelnetdサービスポイントIDが渡されます。  
parblk->flagには次の値が渡されます。

- 0 (WILL) : クライアント側の送受信切り替え抑制の実行要求を受信した。
- 1 (DO) : サーバ側の送受信切り替え抑制の実行要求を受信した。
- 2 (WONT) : クライアント側の送受信切り替え抑制の禁止要求を受信した。
- 3 (DONT) : サーバ側の送受信切り替え抑制の禁止要求を受信した。

## (8) TEV\_TLD\_STATUS

## Telnet オプション状態の受信イベント

コールバックイベント番号 (evcd)
---------------------

TEV\_TLD\_STATUS (0x1605)

パラメータブロックの構造 (parblk)
-----------------------

```
typedef struct{
    UB      neg_options[40];
}T_TLD_MODOPT;

neg_options[0]    TRANSMIT-BINARY (バイナリ転送)
neg_options[1]    ECHO (エコーバック)
neg_options[2]    RCP (再接続) ※1
neg_options[3]    SUPPRESS-GA (送受信切り替え抑制)
neg_options[4]    NAMS (メッセージ長の調停) ※1
neg_options[5]    STATUS (Telnetオプション状態通知)
neg_options[6]    TIMING-MARK (タイミングマーク)
neg_options[7]    RCTE (送信&エコーのリモート制御) ※1
neg_options[8]    NAOL (行幅出力の調停) ※1
neg_options[9]    NAOP (ページサイズ出力の調停) ※1
neg_options[10]   NAOCRD (キャリッジリターン出力の調停) ※1
neg_options[11]   NAOHTS (水平タブストップ出力の調停) ※1
neg_options[12]   NAOHTD (水平タブキャラクタ処理の調停) ※1
neg_options[13]   NAOFFD (フォームフィード出力処理の調停) ※1
neg_options[14]   NAOVTS (垂直タブストップ出力の調停) ※1
neg_options[15]   NAOVTD (垂直タブ出力処理の調停) ※1
neg_options[16]   NAOLFD (ラインフィード出力の調停) ※1
neg_options[17]   EXTEND-ASCII (拡張ASCII) ※1
neg_options[18]   LOGOUT (強制ログアウト)
neg_options[19]   BM (バイトマクロ) ※1
neg_options[20]   DET (データエントリ端末) ※1
neg_options[21]   SUPDUP (SUPDUPディスプレイプロトコル) ※1
neg_options[22]   SUPDUP-OUTPUT (SUPDUP出力) ※1
neg_options[23]   SEND-LOCATION (ロケーションID送信) ※1
neg_options[24]   TERMINAL-TYPE (ターミナルタイプ)
neg_options[25]   END-OF-RECORD (エンドオブレコード) ※1
neg_options[26]   TUID (TACACSユーザ識別) ※1
neg_options[27]   OUTMRK (マーク出力) ※1
neg_options[28]   TTYLOC (端末ロケーション番号) ※1
neg_options[29]   3270-REGIME (TELNET3270様式) ※1
neg_options[30]   X3-PAD (X.3 PAD) ※1
neg_options[31]   NAWS (ウィンドウサイズの調停)
neg_options[32]   TERMINAL-SPEED (端末スピード)
neg_options[33]   TOGGLE-FLOW-CONTROL (フロー制御)
neg_options[34]   LINEMODE (ラインモード)
neg_options[35]   XDISPLOC (Xディスプレイロケーション) ※1
neg_options[36]   ENVIRON (環境変数) ※1
neg_options[37]   AUTHENTICATION (認証) ※1
neg_options[38]   ENCRYPTION (暗号化) ※1
neg_options[39]   NEW-ENVIRON (新環境変数) ※1
```

Telnetオプションの詳細に関しては「3.2.9 Telnetオプションを送信する」の項を参照してください。

## 解 説

クライアントから現在のTelnetオプション状態を受信したことを通知します。  
svpidには受信したTelnetdサービスポイントIDが渡されます。  
parblkの各オプションには次の値が渡されます。

- 0 : サーバ/クライアント共に無効
- 1 : サーバ側のみ有効
- 2 : クライアント側のみ有効
- 3 : サーバ/クライアント共に有効

(※1) : 本バージョンでは未サポートのため、常に0が渡されます。



(9) TEV\_TLD\_LOGOUT 強制ログアウトの受信イベント

コールバックイベント番号 (evcd)

TEV\_TLD\_LOGOUT (0x1612)

パラメータブロックの構造 (parblk)

```
typedef struct{
    UW      flag;          強制ログアウト要求フラグ
} T_TLDPB_COMMON;
```

解 説

クライアントから、強制ログアウトの要求を受信したことを通知します。  
svpidには受信したTelnetdサービスポイントIDが渡されます。  
parblk->flagには次の値が渡されます。

- 1 (DO) : ログアウトの実行要求を受信した。

接続を切断する場合は、TDL\_sht\_srvをコールする必要があります。

(10) TEV\_TLD\_TTYPE

ターミナルタイプ情報の受信イベント

コールバックイベント番号 (evcd)

TEV\_TLD\_TTYPE (0x1618)

パラメータブロックの構造 (parblk)

```
typedef struct{
    UB      *ttype;          ターミナルタイプ文字列への先頭アドレス
    INT     len;            ターミナルタイプ文字列のバイト長
} T_TLDPB_TTYPE;
```

解 説

クライアントから、ターミナルタイプ情報を受信したことを通知します。TelnetオプションのTTYPE (ターミナルタイプ) が有効の場合に通知されます。

svpidには受信したTelnetdサービスポイントIDが渡されます。

parblkにはクライアントのターミナルタイプ文字列 (終端はヌルコード(0)) への先頭アドレスと文字列のバイト長 (ヌルコードは含まない) が渡されます。

(11) TEV\_TLD\_NAWS

ウィンドウサイズの変更イベント

コールバックイベント番号 (evcd)

TEV\_TLD\_NAWS (0x161F)

パラメータブロックの構造 (parblk)

```
typedef struct{
    UH    col;           ターミナルウィンドウの行数
    UH    row;          ターミナルウィンドウの列数
} T_TLDPB_NAWS;
```

解 説

クライアントから、ターミナルウィンドウサイズの変更を受信したことを通知します。TelnetオプションのNAWS（ウィンドウサイズ）が有効の場合で、クライアントのウィンドウサイズに変更があった場合に通知されます。

svpidには受信したTelnetdサービスポイントIDが渡されます。

parblkにはクライアントのターミナルウィンドウのサイズ（行と列の数）が渡されます。

(12) TEV\_TLD\_TSPEED

ターミナル速度情報の受信イベント

コールバックイベント番号 (evcd)

TEV\_TLD\_TSPEED (0x1620)

パラメータブロックの構造 (parblk)

```
typedef struct{
    UW    tspeed;        ターミナルの送信速度
    UW    rspeed;        ターミナルの受信速度
} T_TLDPB_TSPEED;
```

解 説

クライアントから、ターミナル速度情報を受信したことを通知します。TelnetオプションのTSPEED（ターミナル速度）が有効の場合に通知されます。

svpidには受信したTelnetdサービスポイントIDが渡されます。

parblkにはクライアントのターミナル速度の送信速度、および受信速度が渡されます。

(13) TEV\_TLD\_LFLOW

リモートフロー制御要求の受信イベント

コールバックイベント番号 (evcd)

TEV\_TLD\_LFLOW (0x1621)

パラメータブロックの構造 (parblk)

```
typedef struct{
    UW      flag;          XON/XOFFフロー制御要求フラグ
} T_TLDPB_COMMON;
```

解 説

クライアントから、リモートフロー制御の変更要求を受信したことを通知します。TelnetオプションのTOGGLE-FLOW-CONTROL (リモートフロー制御) が有効で、フロー制御に変更があった場合に通知されます。

svpidには受信したTelnetdサービスポイントIDが渡されます。

parblkには次の値が渡されます。

- 0 : フロー制御の禁止
- 1 : フロー制御の許可
- 2 : XOFF以外のキャラクタにて送信再開
- 3 : XONキャラクタにて送信再開

(14) TEV\_TLD\_LINEMODE                      ラインモード変更要求の受信イベント

コールバックイベント番号 (evcd)

TEV\_TLD\_LINEMODE (0x1622)

パラメータブロックの構造 (parblk)

```
typedef struct{
    UW      flag;          LINEMODE要求フラグ
} T_TLDPB_COMMON;
```

解 説

クライアントから、LINEMODEの変更要求を受信したことを通知します。TelnetオプションのLINEMODEが有効 (DO LINEMODEを発行) で、LINEMODEに変更があった場合に通知されます。

svpidには受信したTelnetdサービスポイントIDが渡されます。

parblkには次の値が渡されます。

- 0 (WILL) : クライアントはLINEMODEを有効にする
- 2 (WONT) : クライアントはLINEMODEを無効にする

詳細については、「3.2.9(10) TLOPT\_LINEMODE                      ラインモードオプション」の項を参照してください。

(15) TEV\_TLD\_EDIT

編集モード変更要求の受信イベント

コールバックイベント番号 (evcd)

TEV\_TLD\_EDIT (0x1704)

パラメータブロックの構造 (parblk)

```
typedef struct{
    UW      flag;          編集モード要求フラグ
} T_TLDPB_COMMON;
```

解 説

クライアントから、LINEMODEの編集モードの変更要求を受信したことを通知します。TelnetオプションのLINEMODEが有効 (DO LINEMODEを発行) で、編集モードに変更があった場合に通知されます。

svpidには受信したTelnetdサービスポイントIDが渡されます。

parblkには次の値が渡されます。

- 1 : クライアントは編集モードを有効にする
- 0 : クライアントは編集モードを無効にする

詳細については、「3.2.9(10) TLOPT\_LINEMODE

ラインモードオプション」の項を参照して

ください。

(16) TEV\_TLD\_SIGNAL

割り込み/シグナル通知変更要求の受信イベント

コールバックイベント番号 (evcd)

TEV\_TLD\_SIGNAL (0x1705)

パラメータブロックの構造 (parblk)

```
typedef struct{
    UW      flag;           割り込み/シグナル要求フラグ
} T_TLDPB_COMMON;
```

解 説

クライアントから、LINEMODEの割り込み/シグナル変更要求を受信したことを通知します。TelnetオプションのLINEMODEが有効 (DO LINEMODEを発行) で、割り込み/シグナルに変更があった場合に通知されます。

svpidには受信したTelnetdサービスポイントIDが渡されます。

parblkには次の値が渡されます。

- 1 : クライアントは割り込み/シグナルを有効にする
- 0 : クライアントは割り込み/シグナルを無効にする

詳細については、「3.2.9(10) TLOPT\_LINEMODE                      ラインモードオプション」の項を参照してください。



(17) TEV\_TLD\_SOFTTAB

ソフトタブ変更要求の受信イベント

コールバックイベント番号 (evcd)

TEV\_TLD\_SOFTTAB (0x1706)

パラメータブロックの構造 (parblk)

```
typedef struct{
    UW      flag;          ソフトタブ要求フラグ
} T_TLDPB_COMMON;
```

解 説

クライアントから、LINEMODEのソフトタブ変更要求を受信したことを通知します。TelnetオプションのLINEMODEが有効 (DO LINEMODEを発行) で、ソフトタブに変更があった場合に通知されます。

svpidには受信したTelnetdサービスポイントIDが渡されます。

parblkには次の値が渡されます。

- 1 : クライアントはソフトタブの変換を有効にする
- 0 : クライアントはソフトタブの変換を無効にする

詳細については、「3.2.9(10) TLOPT\_LINEMODE

ラインモードオプション」の項を参照して

ください。

(18) TEV\_TLD\_LITECHO リテラルエコー変更要求の受信イベント

コールバックイベント番号 (evcd)

TEV\_TLD\_SLC (0x1707)

パラメータブロックの構造 (parblk)

```
typedef struct{
    UW      flag;          リテラルエコー要求フラグ
} T_TLDPB_COMMON;
```

解 説

クライアントから、LINEMODEのリテラルエコーの変更要求を受信したことを通知します。TelnetオプションのLINEMODEが有効 (DO LINEMODEを発行) で、リテラルエコーに変更があった場合に通知されます。

svpidには受信したTelnetdサービスポイントIDが渡されます。

詳細については、「3.2.9(10) TLOPT\_LINEMODE ラインモードオプション」の項を参照してください。

(19) TEV\_TLD\_SLC 特殊文字変更要求の受信イベント

コールバックイベント番号 (evcd)

TEV\_TLD\_SLC (0x1708)

パラメータブロックの構造 (parblk)

```
typedef struct {
    UW    slcfunc;      機能
    UB    slcflag;     機能の修飾子
    UB    slcval;      機能を割り当てる文字コード
} T_TLOPT_SLC;
```

解 説

クライアントから、LINEMODEの特殊文字の変更要求を受信したことを通知します。TelnetオプションのLINEMODEが有効 (DO LINEMODEを発行) で、特殊文字に変更があった場合に通知されます。

svpidには受信したTelnetdサービスポイントIDが渡されます。

slcfunc、slcflag、slcvalの内容については、「3.2.9(10) TLOPT\_LINEMODE ラインモードオプション」の項を参照してください。

---

## 5. サービスコール エラーコード一覧

---

表5.1 Telnetdサービスコール エラーコード一覧

エラーコード	説明
E_SYS	システムエラー
E_RSFN	拡張サービスコールが登録されていない
E_PAR	パラメータエラー
E_ID	不正ID番号
E_NOMEM	メモリ不足
E_NOID	ID番号不足
E_OBJ	オブジェクト状態不正
E_NOEXS	オブジェクト未生成
E_RLWAI	処理のキャンセル、待ち状態の強制解除
E_TMOUT	ポーリング失敗、またはタイムアウト
E_CLS	接続できない、または接続が切断された
E_BOVR	バッファオーバーフロー
E_ILUSE	不正なサービスコール呼び出し
EV_PROT	プロトコルエラー

Telnetサーバ  
リファレンスマニュアル  
CE7000TLD02J-7

発行年月 2013年 9月 第7版  
発行 株式会社ルネサス北日本セミコンダクタ  
編集 株式会社ルネサス北日本セミコンダクタ

©株式会社ルネサス北日本セミコンダクタ 2013