

VTTY マニュアル

Version 1.1.2

Rev.	日付	作成者	ヒストリー
1.0.0	2003-12-04	UIS	草稿
1.1.0	2005-07-05	JOJ	Linux カーネル 2.6 サポート追加
1.1.2	2005-07-08	JOJ	変更箇所追加

2005.07.18

目次

1.	はじめに	P.3
2.	コンパイルとインストール	P.3
3.	カーネルドライバ・モジュール	P.4
4.	デバイスファイル	P.4
5.	データベース	P.4
6.	構文	P.5

1. はじめに

vtty(仮想端末)は `sena_vpty`, `ttyd`,および `vtty_manager` により構成されています。

`Sena_vpty`とは、Linux カーネル内部へ動的ロードされるデバイスドライバーモジュールのことです。

`ttyd`は端末サーバーアプリケーションのことです。`vtty_manager` はドライバーモジュール,
`pseudo-tty` デバイスおよび端末サーバーアプリケーションの制御方法を提供するラップ・アップ(要約)アプリケーションです。

2. コンパイルとインストール

お使いの Linux マシン内のどこかに `vtty-1.1.2.tar.gz` をコピーしてください。

```
#tar-xvzf vtty-1.1.2.tar.gz
#cd vtty
#make
#make install
```

注記:

1. 全てのバイナリはインストール時に”`/user/local/vpty`”にインストールされ、そのインストール作業時にコンパイルカーネルモジュールは”`/user/local/vpty`”にコピーされます。初期スクリプト”`vtty init`”は”`/etc/rc.local`”の最後部分に追加されます。
2. いくつかの Linux 配布元はブート時に”`etc/rc.local`”を実行しません。そのような時は、”`vtty init`”毎システムリブート時に手動で実行する必要があります。システムのブートのすぐ後に次のコマンドを入力することにより `rc.local` を実行するかどうかを調べることができます。

```
#lsmod | grep sena_vpty
```

3. カスタム 2.4.x kernel でvttyを使用するには、使用する機器で実行している対応するカーネルソースをもち、なおかつ `/usr/src/linux-2.4` がカーネルソースのトップディレクトリにリンクされている必要があります。
4. Linux カーネル 2.6.x でvttyを使用するには、使用する機器上で実行している設定されたカーネルソースが必要です。カーネルの設定およびコンパイルに関しては、各 Linux ディストリビューションからの関連文書を参照してください。
5. いくつかの Linux 配布元(Fedora のような)のデフォルトの Linux カーネルにはレガシーPTY サポートが含まれていないので、カーネル設定時に次の二つのオプションをリコンパイルしてください。

```
-Device Drivers->Character devices->(*) Legacy (BSD)PTY support
```

```
-Device Drivers->Character devices->(256) Maximum number of legacy PTY in use
```

`/proc/tty/drivers` をチェックすることにより使用しているカーネルがレガシーPTY サポートを

含んでいるかを調べることが可能です。もし `ttyp` および `pty` ノードを使用する二つのドライバがあるなら、レガシーPTY サポートを持つカーネルがある証拠です。

```
#cat /proc/tty/drivers
```

```
...
```

```
pty_slave      /dev/ttyp      3      0-255 pty:slave
pty_master     /dev/pty       2      0-255 pty:master
```

3. カーネルドライバ・モジュール

インストール後、最初の `v tty init` または `v tty add[arguments..]` の実行で、その Pseudo-tty (仮想tty) デバイスのドライバモジュールは `sena_vpty` の名称でカーネルにロードされます。デバイスの主要番号は 226 および 227 が各マスターおよびスレーブノードに個別につけられます。最高 255 デバイスまで追加可能で、それらのデバイスファイルは `v tty_manager` により管理されます。

注記: 新しいカーネルドライバモジュールをインストールするときは、`v tty init` コマンドを実行する前に `rmmod sena_vpty` コマンド (既存のドライバをアンロードする) を実行してください。

4. デバイスファイル

`v tty add [SUFFIX]` コマンドを使用してノードを追加すると、マスターおよびスレーブのペア: マスターに `/dev/vpty[SUFFIX]`、そしてスレーブに `/dev/tty[SUFFIX]`、を生成します。

その `v tty_manager` は 0-255 番までで最小の数値を選択し、そのデバイスに割り当てます。

5. データベース

`v tty_manager` はノード (デバイスファイル)、端末サーバー側のシリアルオプション、およびノードステータスを含むデータ管理用のローカルデータベースを使用します。そのデータベースは `/var/vtty/vtty.db` として生成されます。

6. 構文

```
vtty {add | addnt} TTYSUFFIX ip_address tcp_port [serial_option]
```

```
vtty { enable |enablt | disable | remove } TTYSUFFIX
```

```
vtty status [TTYSUFFIX]
```

注記: TTYSUFFIX はデバイスファイル名のための suffix であり、同時に vtty database 用の管理キーワードになります。

Serial_option: -b <baudrate> -p <port_setting>

<baudrate>: 以下の数値から選択します。

75,150,200,300,600,1200,1800,2400,4800,9600,19200,38400,57600,115200,230400

<port_setting>:以下の数値を組み合わせます。間にスペースを空けません。

8	8ビット文字
7	7ビット文字
6	6ビット文字
5	5ビット文字
N	パリティ無し
E	偶数パリティ
O	奇数パリティ
C0	ハードウェアフロー制御なし
C1	ハードウェアフロー制御
S0	ソフトウェアフロー制御なし
S1	ソフトウェアフロー制御

シリアルオプションが指定されない場合は、そのシリアルオプションのデフォルト値は:

baudrate=9600、port setting= 8NC0 となります。

例:

```
vtty add mysuffix 192.168.1.1 6001 -b 9600 -p 8NC0
```

```
vtty enable mysuffix
```

```
vtty status
```

Add(Addnt):

```
vtty add mysuffix 192.168.1.1 6001 -b 9600 -p 8NC0
```

ローカルデータベースに1つのノード情報を追加し、2つのデバイスファイルを作成します。

マスター(/dev/vptymysuffix)、およびスレーブ(/dev/ttymysuffix)です。それから、端末サーバプロ

セス“ttyd”を呼び出し、リモートホストへのスレーブ接続が作動可能になります。アプリケーションとしてスレーブを使用することも可能です。新規に追加されたデバイスの状況は「使用可能」です。

vttyを起動しているときに以下のような問題が発生すると、そのコマンドは破棄されます。

- デバイス追加の制限を超過する(新規デバイスが使用可能なマイナー番号がなくなる)
- ローカルデータベース内に suffix(接尾部)がすでに存在する
- ドライバモジュールのロード失敗
- 同じ名前のデバイスファイルが既に存在する(マスターまたはスレーブデバイスファイル内)
- デバイスファイルの作成失敗
- “ttyd”プロセス起動障害

注記:

1. カーネル内にドライバモジュールがロードされていない場合、“init”または“init”コマンドがモジュールをチェックし、そのモジュールをカーネル内にロードします。
2. “add”コマンドで新しいノードを追加すると、いくつかの Telnet オプションが接続開始時にデバイスに送られます。 Super シリーズ、STS シリーズ、および new Pro シリーズ (PS110/410/810)には Telnet サポートを使用可能にするオプションがあります。Telnet プロトコルを使用しない場合は、“addnt”コマンドで Telnet オプションを使用不可にしてください。

Remove:

```
vtty remove my suffix
```

ローカルデータベースからノード情報を削除し、2 つのデバイスファイル、マスター(/dev/vptymysuffix)、およびスレーブ(/dev/ttymysuffix)を削除します。それから“ttyd”プロセスに関連する実行を削除(kill)します。

Enable(Enablent):

```
vtty enable mysuffix
```

ノード状況が“disabled”(使用不可)の時、そのコマンドはノード状況を“enabled”(使用可能)にし、接続用に“ttyd”プロセスを呼び出して作動可能にします。もし disabled 状況でなければ、このコマンドを入力しても何もおきません。

注記: ノードを“enabled”コマンドで使用可能にすると、いくつかの Telnet オプションが接続開始時にデバイスに送られます。 Super シリーズ、STS シリーズ、および new Pro シリーズ (PS110/410/810)には Telnet サポートを使用可能にするオプションがあります。Telnet プロトコルを使用しない場合は、“enablent”コマンドで Telnet オプションを使用不可にしてください。

Disable:

```
vtty disable mysuffix
```

ノード状況が“enabled”の時、そのコマンドはノード状況を”disabled”にし、“ttyd”に関連したプロセスを削除(kill)します。もし enabled 状況でなければ、このコマンドを入力してもなにもおきません。

注記: “disabled”コマンドがデバイスファイルを削除することはありません。

Status:

```
vty status[TTYSUFFIX]
```

ノード状況(ステータス)を表示します。 TTYSUFFIX が与えられると、それは指定したノードの状況を表示します。それ以外は、ローカルデータベースに登録されている全てのノード状況を表示します。

注記: 結果の STATUS フィールドは、CONNECTED、DISCONNECTED、DISABLED、の三種類に分類されます。

Init:

```
vty init
```

ドライバーモジュールをロードすることによりvtyを初期化し、ローカルデータベース内の“enabled”状態の”ttyd”プロセスを実行します。このコマンドは 2 度以上の効果はありません。

注記:

1. このコマンドは“/ect/rc.local”ファイルの最後に追加され、機器のブートアップ時に実行されます。
2. いくつかの Linux 配布元はブート時に”etc/rc.local”を実行しません。そのような時は、”vty init”毎システムリブート時に手動で実行する必要があります。システムのブートのすぐ後に次のコマンドを入力することにより rc.local を実行するかどうかを調べることができます。

```
#lsmod | grep sena_vpty
```

- ・ 本製品の開発、製造はセナ・テクノロジー社です。
- ・ 仕様は断りなく変更されることがあります。
- ・ 製品名、会社名は、各社の商標あるいは登録商標です。