

運用管理装置 PMANモデル50/100

接続サーバのtty設定について

富士通株式会社

改版履歴

No.	文書番号	日付	変更項目	変更箇所
1	POW-FFC-E-0034-R00	2002. 12. 16	初版	全頁
2	POW-FFC-E-0034-R01	2003. 06. 19	補足文を追加	全頁
3	POW-FFC-E-0034-R02	2005. 07. 26	Linux設定対応	全頁
4	POW-FFC-E-0034-R03	2006. 12. 26	公開用書き直し Solaris9, 10 設定対応 Linuxコンソールポート設定追記 対応するPMAN装置側の設定追記	全頁
5	POW-FFC-E-0034-R04	2007. 02. 09	Solaris9 以降のtty端末設定をコマンドライン内でユーザID パスワードを入力するように変更。 ※Solaris 10 6/06 にて認証の仕様が 変わり対話実行では7文字以下のパスワードでは認証不可なため	2. 1. 2
6	POW-FFC-E-0034-R05	2007. 02. 15	「コンソール」についての説明文追記。	第4章 PMAN装置スクリプトについて
7	POW-FFC-E-0034-R06	2011. 05. 31	「3. 1. 2 Red Hat Enterprise Linux 6のtty端末設定」の追加	3. 1. 2 第1章 概要

はじめに

本書は運用管理装置PMANモデル50/100（以降、PMAN装置と称します）における接続サーバ（UNIX系OS搭載のサーバ）側のtty設定について記述した技術文書です。サーバでのtty設定の詳細については、各サーバにインストールされるOSのマニュアルをご参照ください。

■ 本書の目的

本書では、PMAN装置を利用したサーバの電源制御、コンソール透過通信を行うためのサーバ側の設定について記述しています。PMAN装置側の設定は、装置添付のマニュアルをご参照ください。

■ 本書の対象者

- ・ 本装置を使用したシステムの構築を行う方

■ 本書の構成

1. 概要

PMAN装置に接続するUNIX系OS搭載サーバ（Solaris, Linux等）のtty設定を行うための本書のガイドラインについて説明します。

2. Solarisでの設定について

Solarisに特有のtty設定について説明します。

3. Linuxでの設定について

Linuxに特有のtty設定について説明します。

4. PMAN装置スクリプトについて

サーバ側のtty設定によるスクリプトの違いについて説明します。

■ 関連マニュアル

以下の資料を参照する事により、本書をより深く理解する事が出来ます。

- 「POW-FFC-E-0030 運用管理装置PMANモデル50/100 スタートアップガイド」
- 「POW-FFC-E-0031 運用管理装置PMANモデル50/100 ユーザーズガイド」
- 「POW-FFC-E-0032 運用管理装置PMANモデル50/100 コマンドリファレンス」
- 「LA91001-0184 運用管理装置PMANモデル100 取扱説明書」
- 「LA91001-0202 運用管理装置PMANモデル50 取扱説明書」
- 「LA91001-0185 コンセントボックス1 取扱説明書」

お願い

- ・ 本書を無断で複写または他に転写しないでください。
- ・ 本書は予告なく変更することがあります。

UNIXは、米国およびその他の国におけるオープン・グループの登録商標です。

すべてのSPARC商標は、SPARC International, Inc. のライセンスを受けて使用している同社の米国およびその他の国における登録商標です。

OracleとJavaは、Oracle Corporation及びその子会社、関連会社の米国及びその他の国における登録商標です。

Linuxは、Linus Torvalds氏の米国およびその他の国における登録商標または商標です。

Red Hat、RPMおよびRed Hatをベースとした全ての商標とロゴは、Red Hat Inc. の米国およびその他の国における登録商標または商標です。

Microsoft、Windows、Windows Server、Hyper-Vは、米国Microsoft Corporationの米国およびその他の国における登録商標または商標です。

記載されている会社名、製品名は各社の登録商標または商標です。

目次

第1章	概要	6
第2章	Solarisでの設定について	7
2.1	SOLARISサーバの電源制御	7
2.1.1	SOLARIS8 以前のTTY端末設定	7
2.1.2	SOLARIS9 以降のTTY端末設定	9
2.2	SOLARISサーバのコンソール透過通信	13
第3章	Linuxでの設定について	14
3.1	LINUXサーバによる電源制御	14
3.1.1	RED HAT ENTERPRISE LINUX 5 以前のTTY端末設定	14
3.1.2	RED HAT ENTERPRISE LINUX 6 のTTY端末設定	17
3.2	LINUXサーバによるコンソール透過通信	18
3.2.1	BIOSの設定	18
3.2.2	LINUXブートの設定	18
3.2.3	LINUXの設定	19
第4章	PMAN装置スクリプトについて	20

第1章 概要

運用管理装置PMANモデル50/100（以降、「PMAN装置」と称します）にて電源制御やコンソール透過通信を行うために、接続するサーバ側にてtty設定が必要になります。tty設定は、PMAN装置で使用する機能レベルで変わります。

PMAN装置での電源制御、コンソール透過通信ではどのような設定が必要なのかを以下に示します。

項	機能	説明	備考
1	電源制御 (電源切断)	UNIX系OSでの電源切断では、PMAN装置のスク립トを利用してtty経由でサーバに自動ログインし、シャットダウンを実施し、それから電源を切断させます。そのため、サーバのシリアルコネクタにtty端末の設定が行われている必要があります。	
2	コンソール透過通信	UNIX系OSでのコンソールポートの入出力をネットワークに変換します。そのため、サーバのシリアルコネクタにコンソールポートの設定が行われている必要があります。	OS起動後のオペレーションのみであれば1項のtty端末の設定でも可能です。

本書では、SolarisとLinuxについてのtty設定について記述します。次の表を参考に必要な項目を読んでください。

サーバOS	機能	電源制御	コンソール透過通信
	Solaris	8以前 9以降	以下を参照ください。 「 2.1.1 Solaris8以前のtty端末設定 」 以下を参照ください。 「 2.1.2 Solaris9以降のtty端末設定 」
Linux (*)	RHEL 5 以前 RHEL 6	以下を参照ください。 「 3.1.1 Red Hat Enterprise Linux 5以前のtty端末設定 」 以下を参照ください。 「 3.1.2 Red Hat Enterprise Linux 6のtty端末設定 」	以下を参照ください。 「 3.2 Linuxサーバによるコンソール透過通信 」

(*) RHEL: Red Hat Enterprise Linux

第2章 Solarisでの設定について

本章では、Solaris における tty 設定について説明します。Solaris では、OS のバージョンによってシリアルコンソールの設定方法が変更になっています。それぞれのバージョン及びPMAN装置の機能による設定について説明します。

PMAN装置側の設定は、以下の設定を前提にしています。

速度：9600bps
パリティ：なし
データ長：8ビット
ストップビット：1ビット
フロー制御：ソフトウェア

2.1 Solarisサーバの電源制御

SolarisにおいてPMAN装置を利用して電源制御を行うには、シリアルポートをtty端末として設定します。ここでは、Solarisのバージョンによってtty端末としての設定方法が変更になっているため、バージョン毎に説明します。

2.1.1 Solaris8 以前のtty端末設定

Solaris8 以前では、「/etc/inittab」ファイルを編集してtty端末設定を行います。本ファイルの編集にはrootユーザでのログインが必要になります。

「/etc/inittab」ファイルの「co:234:respawn:/usr/lib/saf/ttymon -g -h -p ``uname -n` console login: `` -T vt100 -d /dev/console -l console -m ldterm, ttcompat」の直下に下記1行を追加します。

「c2:234:respawn:/usr/lib/saf/ttymon -g -h -p ``uname -n` console login: `` -T vt100 -d /dev/ttyb -l console -m ldterm, ttcompat」

例：netraX1 の tty 端末設定(ttyb)を下記に示します。

</etc/inittab>

```
ap::sysinit:/sbin/autopush -f /etc/iu.ap
ap::sysinit:/sbin/soconfig -f /etc/sock2path
fs::sysinit:/sbin/rcS sysinit          >/dev/msglog 2</dev/msglog </dev/console
is:3:initdefault:
p3:s1234:powerfail:/usr/sbin/shutdown -y -i5 -g0 >/dev/msglog 2</dev/msglog
sS:s:wait:/sbin/rcS                    >/dev/msglog 2</dev/msglog </dev/console
s0:0:wait:/sbin/rc0                    >/dev/msglog 2</dev/msglog </dev/console
s1:1:respawn:/sbin/rc1                 >/dev/msglog 2</dev/msglog </dev/console
s2:23:wait:/sbin/rc2                   >/dev/msglog 2</dev/msglog </dev/console
s3:3:wait:/sbin/rc3                    >/dev/msglog 2</dev/msglog </dev/console
s5:5:wait:/sbin/rc5                    >/dev/msglog 2</dev/msglog </dev/console
s6:6:wait:/sbin/rc6                    >/dev/msglog 2</dev/msglog </dev/console
fw:0:wait:/sbin/uadmin 2 0              >/dev/msglog 2</dev/msglog </dev/console
of:5:wait:/sbin/uadmin 2 6              >/dev/msglog 2</dev/msglog </dev/console
rb:6:wait:/sbin/uadmin 2 1              >/dev/msglog 2</dev/msglog </dev/console
sc:234:respawn:/usr/lib/saf/sac -t 300
co:234:respawn:/usr/lib/saf/ttymon -g -h -p "`uname -n` console login: " -T vt100 -d
/dev/console -l console -m ldterm, ttcompat
c2:234:respawn:/usr/lib/saf/ttymon -g -h -p "`uname -n` console login: " -T vt100 -d
/dev/ttyb -l console -m ldterm, ttcompat ←1行追加
```

追加した1行の内容は下記を参照してください。

パラメタ	説明
co:234:respawn:	coは、エントリをコンソールとして指定します。234は、動作 respawn の実行レベルを指定します。つまり失敗した場合、あるいは実行レベル 2、3、および 4 でない場合、このコンソールエントリが再起動されるべきであることを示します。
/usr/lib/saf/ttymon -g -h	-g オプションを使用するため、正しいボーレートと正しい端末設定をポート上で設定でき、SACによる事前構成なしに、ログインサービスに接続できます。-h オプションは、デフォルトまたは指定した速度に設定する前に、回線速度をゼロに設定することにより、回線をハングアップさせます。
-p "`uname -n` console login:	コンソールポート用のプロンプト文字列を指定します。
-t terminal_type	コンソールの端末タイプを指定します。
-d /dev/console -l console -m ldterm, ttcompat	-d オプションは、コンソールデバイスを指定します。-l オプションは、/etc/ttydefs ファイル内の tty 名を指定します。-m オプションは、プッシュする STREAMS モジュールを指定します。

上記設定を変更し、サーバの再起動を行います。再起動後、クライアント端末からtty端末接続(ttyb)を行い、ログインプロンプトが表示されれば成功です。本設定ポートでは、ユーザ権限のみログイン可能ですので、まずはOS側でユーザ権限の適当なユーザIDを作成してログインし、ログイン後"su"コマンドでrootユーザになりシャットダウンを実行するような操作となるように、PMAN装置のスクリプトも変更設定を行ってください（「第4章 PMAN装置スクリプトについて」を参照ください）。

2.1.2 Solaris9 以降のtty端末設定

Solaris9 以降でのtty端末の設定方法には、GUIとCLIの2通りあります。

S series、PRIMEPOWERでは、サーバ本体のttyaにはALOM(Advanced Lights Out Manager)やXSCF(eXtended System Control Facility)等の機能が標準装備されていますので、ttybを使用することを推奨します。

(1) GUIによる設定

本設定方法はttybを使用する例であり、ttyaを設定する場合には、ポート名(Port Name)を b から a に置き換えて下さい。

1. Solaris 管理コンソールが実行していなければ、起動します。

```
% /usr/sadm/bin/smc &
```

Solaris 管理コンソールの起動については、『Solaris のシステム管理(基本編)』の「Solaris Management Console を起動する」を参照してください。

2. ナビゲーション区画で「このコンピュータ(Computer)」アイコンをクリックします。
3. 「Devices and Hardware」->「シリアルポート(Serial Ports)」をクリックします。シリアルポートメニューが表示されます。
4. 端末に使用するポートを選択します。(ここでは仮にcbとします。)

5. 「アクション(Action)」メニューから「構成(Configure)」->「端末(Terminal)」を選択します。
「シリアルポートbを構成(Configure Serial Port)」ウィンドウが「基本(Basic)」モードで表示されます。

「端末(Terminal)」メニュー項目については、以下の通りです。

• 基本(Basic)	
ポート名(Port Name)	: b
備考欄(Description)	: Terminal
サービス状態(Service Status)	: 使用可能
ボーレート(Baud Rate)	: 9600
端末の種類(Terminal Type)	: tvi925
ログインプロンプト(Login Prompt)	: ttyb login:
• 拡張機能(Advanced)	
キャリア検出(Carrier Detection)	: ソフトウェア
キャリア検出時に接続(Connect on Carrier)	: いいえ
オプション: 発着信両用(Bidirectional)	: いいえ
オプション: 初期化のみ(InitializeOnly)	: いいえ
タイムアウト(秒) (Timeout)	: なし
ポートモニター(Port Monitor)	: zsmn
サービスプログラム(ServiceProgram)	: /usr/bin/login

6. 「了解(OK)」をクリックします。
7. 各項目を設定するには、端末として設定されたポートを選択します。次に、「アクション(Action)」メニューから「プロパティ(Properties)」を選択します。
8. 必要な場合は、テンプレートエントリの値を変更します。
9. 「了解(OK)」をクリックしてポートを設定します。

10. 端末サービスが追加されていることを確認します。

```
$ pmadm -l -s ttyb
```

表示例：

```
[root@jedi PMANL]# pmadm -l
PMTAG          PMTYPE          SVCTAG          FLGS ID          <PMSPECIFIC>
zsmon          ttymon          ttya            u   root          /dev/term/a I - /usr/bin/login -
9600 ldterm,ttcompat ttya login: - tvi925 y #
zsmon          ttymon          ttyb            u   root          /dev/term/b -- /usr/bin/login -
9600 ldterm,ttcompat ttyb login: - tvi925 y #Terminal
```

※なお、シリアルポートからはrootユーザでログインできませんので、まずはOS側でユーザ権限の適当なユーザIDを作成してログインし、ログイン後“su”コマンドでrootユーザになりシャットダウンを実行するような操作となるように、PMANのスク립トも変更設定を行ってください（「第4章 PMAN装置スク립トについて」を参照ください）。

(2) CLIによる設定

本設定方法はttybを使用する例であり、ttyaを設定する場合には、smserialportコマンドパラメータの -n b を -n a に置き換えて下さい。

1. Solarisへrootユーザでログインします。
2. シリアルポートの構成を端末にします。

以下の例では、管理者はユーザID : root、パスワード : rootrootとなっています。

```
# /usr/sadm/bin/smserialport configure -u root -p rootroot -- -n b -x
device=terminal

ツール com.sun.admin.serialmgr.cli.SerialMgrCli を hostname から読み込み中
ユーザー root として hostname にログインしました。
hostname から com.sun.admin.serialmgr.cli.SerialMgrCli がダウンロードされました。

#
```

この時点でttybのポートは接続可能になります。但し、スピード等他の設定が必要です。

3. シリアルポートの設定を行います。

```
# /usr/sadm/bin/smserialport modify -u root -p rootroot -- -n b -b 9600 -t vt100 -l "ttyb login:" -c "Terminal" -x service=y -x software_carrier=y -x connect_on_carrier=n -x bidirectional=n -x initializeonly=n -x timeout=never -x service_program=/usr/bin/login
```

ツール com.sun.admin.serialmgr.cli.SerialMgrCli を hostname から読み込み中
ユーザー root として hostname にログインしました。
hostname から com.sun.admin.serialmgr.cli.SerialMgrCli がダウンロードされました。

```
#
```

上記設定は、以下のPMAN装置の設定で動作します。

速度：9600bps

パリティ：なし

データ長：8ビット

ストップビット：1ビット

フロー制御：ソフトウェア

4. pmadmコマンドでシリアルポートの設定を確認します。

```
# pmadm -l
PMTAG          PMTYPE          SVCTAG          FLGS ID          <PMSPECIFIC>
zsmon          ttymon          ttya            u  root          /dev/term/a I - /usr/bin/login -
9600 ldterm, ttcompat ttya login: - tvi925 y #
zsmon          ttymon          ttyb            u  root          /dev/term/b -- /usr/bin/login -
9600 ldterm, ttcompat ttyb login: - vt100 y #Terminal
#
```

※なお、シリアルポートからはrootユーザでログインできませんので、まずはOS側でユーザ権限の適当なユーザIDを作成してログインし、ログイン後"su"コマンドでrootユーザになりシャットダウンを実行するような操作となるように、PMAN装置のスクリプトも変更設定を行ってください（「第4章 PMAN装置スクリプトについて」を参照ください）。

2.2 Solarisサーバのコンソール透過通信

SolarisにおいてPMAN装置を利用してコンソール透過通信を行うには、シリアルポートをコンソールポートとして設定します。ここでは、Solarisの起動を行うOpenBootPROMにおいてシリアルポートをコンソールポートとして設定する方法を示します。

本設定方法はttybを使用する例であり、ttyaを設定する場合には、ttyaをttyb に置き換えて下さい。

1. OpenBootPROMモードにします。Solaris (OS) が起動していれば、シャットダウン (shutdown) して OpenBootPROMモードにします (詳細は、各サーバのマニュアルを参照ください。)
2. OKプロンプトで、input-device及びoutput-deviceの設定を行います。例ではttyaに設定しています。

```
ok setenv input-device ttya  
ok setenv output-device ttya  
ok
```

これで設定は完了です。OSを起動してクライアント端末をttyaに接続してコンソールログインプロンプトが表示されれば設定成功です。rootユーザでログインできますので、PMAN装置のスクリプトをrootでのログインとして記述してください。

第3章 Linuxでの設定について

本章では、Linuxにおけるtty設定においてPMAN装置の機能による設定を説明します。
PMAN装置側の設定は、以下の設定を前提にしています。

速度：9600bps
パリティ：なし
データ長：8ビット
ストップビット：1ビット
フロー制御：なし

3.1 Linuxサーバによる電源制御

LinuxにおいてPMAN装置を利用して電源制御を行うには、シリアルポートをtty端末として設定します。
ここでは、Linuxのバージョンによってtty端末としての設定方法が変更になっているため、バージョン毎に説明します。

3.1.1 Red Hat Enterprise Linux 5 以前のtty端末設定

Red Hat Enterprise Linux 5 以前では、「/etc/inittab」ファイルを編集してtty端末設定を行います。本ファイルの編集にはrootユーザでのログインが必要になります。

「# Run gettys in standard runlevels」の直下に下記1行を追加します。

```
「co:2345:respawn:/sbin/agetty ttyS0 9600 vt100」
```

※各ディストリビューションについては、下記例の最後を参照ください。

例. </etc/inittab>

```
#
# inittab      This file describes how the INIT process should set up
#              the system in a certain run-level.
#
# Author:      Miquel van Smoorenburg, <miquels@drinkel.nl.mugnet.org>
#              Modified for RHS Linux by Marc Ewing and Donnie Barnes
#

# Default runlevel. The runlevels used by RHS are:
# 0 - halt (Do NOT set initdefault to this)
# 1 - Single user mode
# 2 - Multiuser, without NFS (The same as 3, if you do not have networking)
# 3 - Full multiuser mode
# 4 - unused
# 5 - X11
# 6 - reboot (Do NOT set initdefault to this)
#
id:3:initdefault:

# System initialization.
si::sysinit:/etc/rc.d/rc.sysinit

10:0:wait:/etc/rc.d/rc 0
11:1:wait:/etc/rc.d/rc 1
12:2:wait:/etc/rc.d/rc 2
13:3:wait:/etc/rc.d/rc 3
14:4:wait:/etc/rc.d/rc 4
15:5:wait:/etc/rc.d/rc 5
16:6:wait:/etc/rc.d/rc 6

# Things to run in every runlevel.
ud::once:/sbin/update

# Trap CTRL-ALT-DELETE
ca::ctrlaltdel:/sbin/shutdown -t3 -r now

# When our UPS tells us power has failed, assume we have a few minutes
# of power left.  Schedule a shutdown for 2 minutes from now.
# This does, of course, assume you have powerd installed and your
# UPS connected and working correctly.
pf::powerfail:/sbin/shutdown -f -h +2 "Power Failure; System Shutting Down"
<続く>
```

<続き>

```
# If power was restored before the shutdown kicked in, cancel it.
pr:12345:powerokwait:/sbin/shutdown -c "Power Restored; Shutdown Cancelled"

# Run gettys in standard runlevels
co:2345:respawn:/sbin/agetty ttyS0 9600 vt100
#1:2345:respawn:/sbin/mingetty tty1
#2:2345:respawn:/sbin/mingetty tty2
#3:2345:respawn:/sbin/mingetty tty3
#4:2345:respawn:/sbin/mingetty tty4
#5:2345:respawn:/sbin/mingetty tty5
#6:2345:respawn:/sbin/mingetty tty6

# Run xdm in runlevel 5
# xdm is now a separate service
x:5:respawn:/etc/X11/prefdm -nodaemon
```

■各ディストリビューションの設定は以下の通りになります。

(1) Turbo Linuxの場合は

```
「co:2345:respawn:/sbin/mingetty ttyS0 9600 vt100」
```

を設定します。

(2) OpenLinuxServer3.1.1の場合は

```
「co:2345:respawn:/sbin/getty ttyS0 9600 vt100」
```

を設定します。

■サーバマネジメントポートを有する装置に関してはCOM2に接続し、下記設定を行います。

PRIMERGY TX200, RX300※1

```
「co:2345:respawn:/sbin/agetty ttyS1 9600 vt100」
```

※1 Red Hat Enterprise Linux ES V2.1 はttyS0 にします。

設定が完了したらサーバを再起動するか、「init q」により設定を反映させます。

3.1.2 Red Hat Enterprise Linux 6 の tty 端末設定

Red Hat Enterprise Linux 6 では、以下のように tty 端末設定を行います。本ファイルの編集には root ユーザでのログインが必要になります。

「/etc/init」ディレクトリ配下に、下記内容の下記新規ファイルを作成します。

- ファイル名：「serial-ttyS0.conf」
- ファイル内容：

```
# This service maintains a getty on /dev/ttyS0.  
  
start on stopped rc RUNLEVEL=[2345]  
stop on starting runlevel [016]  
  
respawn  
exec /sbin/agetty /dev/ttyS0 9600 vt100-nav
```

設定が完了したらサーバを再起動し設定を反映させます。

3.2 Linuxサーバによるコンソール透過通信

Linuxによるコンソール透過通信の設定は、サーバのBIOS設定、Linuxのブート設定、Linuxの設定に分けられます。以下にそれぞれについて説明します。

3.2.1 BIOSの設定

サーバのBIOSは入出力をシリアルで行うことが可能です。PMAN装置のコンソール透過通信を使用しBIOSの保守まで行う場合に設定を行ってください。「シリアルポートの機能」の設定は「System」を選択してください。また、「コンソールリダイレクション」の設定は、PMAN装置の通信設定に合わせてください。設定の詳細は、サーバのマニュアルを参照ください。

3.2.2 Linuxブートの設定

ここではLinuxのブートローダであるgrubの設定について説明します。ここでの設定は、grubの入出力をシリアルポートで行うように設定します。PMAN装置のコンソール透過通信を使用しLinuxのブートローダも制御する場合に設定を行ってください。

「/boot/grub/grub.conf」ファイルを編集してLinuxのブートローダの設定を行います。本ファイルの編集にはrootユーザでのログインが必要になります。

- ・「splashimage=(hd0,1)/boot/grub/splash.xpm.gz」行をコメントアウト又は削除します。
- ・下記2行を追加します。

```
「serial --unit=0 --speed=9600 --word=8 --parity=no --stop=1」
```

```
「terminal serial」
```

- ・kernel文の後に以下の文を追加

```
「console=tty0 console=ttyS0,9600」
```

例. </boot/grub/grub.conf> (ディスプレイの表示と同時使用の場合)

```
# grub.conf generated by anaconda
#
# Note that you do not have to rerun grub after making changes to this file
# NOTICE: You have a /boot partition. This means that
#         all kernel and initrd paths are relative to /boot/, eg.
#         root (hd0,0)
#         kernel /vmlinuz-version ro root=/dev/sda2
#         initrd /initrd-version.img
#boot=/dev/sda
default=0
timeout=10
serial --unit=0 --speed=9600 --word=8 --parity=no --stop=1
terminal serial
# splashimage=(hd0,0)/grub/splash.xpm.gz
title Red Hat Enterprise Linux AS (2.6.9-5.EL)
        root (hd0,0)
        kernel /vmlinuz-2.6.9-5.EL ro root=/dev/VolGroup00/LogVol100 rhgb quie
t console=tty0 console=ttyS0,9600
        initrd /initrd-2.4.21-4.0.1.ELsmp.img
```

3.2.3 Linuxの設定

Linuxの設定では、先に説明した「/etc/inittab」を設定し、さらに以下のファイルの設定変更が必要です。

- ・「/etc/sysconfig/kudzu」：「SAFE=yes」に変更
→起動時にttyS0 の新規デバイスが見つかったメッセージが表示された場合は、「Ignore」を選択してください。
- ・「/etc/securetty」：「ttyS0」を追加
- ・「/etc/sysconfig/init」：「BOOTUP=serial」を設定 (推奨)

上記設定によりコンソールからrootユーザによるログインが可能になります。

第4章 PMAN装置スクリプトについて

PMAN装置のスクリプトは、サーバ側のシリアルポートがtty端末設定の場合とコンソール設定の場合では自動ログイン部分が異なります。

コンソール設定はOSのデフォルト入出力先として設定されている状態です。OSメッセージ（ブート時の起動メッセージやシャットダウン時の各デーモンの停止処理メッセージ等）の出力先となります。Solarisの場合、OpenBootPROMモード（okプロンプト）はこのコンソール設定に指定されている入出力先でのみ使用できます。

通常、サーバのデフォルト状態としてはコンソールはCRT、もしくはttyaに割り当てられています。

（コンソールの割り当てを変更する場合には「2.2 Solarisサーバのコンソール透過通信」、「3.2 Linuxサーバのコンソール透過通信」をご参照ください。）

コンソール設定の場合は、自動ログイン部分（ログオンスクリプト）にてrootユーザでログインできるため、そのまま「root」ユーザでログインします。

tty端末設定（本書の内容に従い新規に追加したシリアルコンソール。コマンドシェル機能のみをもつ。）の場合は、rootユーザでのログインはできません。一度一般ユーザにてログインし、ログイン後に「su」コマンドにてrootユーザになり、「shutdown」コマンドを実行する手順となります。従って、自動ログイン部分（ログオンスクリプト）は、一般ユーザでのログインまでのスクリプトを記述し、シャットダウンスクリプトにて「su」コマンドによるrootユーザへの移行、及びシャットダウンコマンドの実行までを記述します。

なお、PMAN装置のスクリプトで使用できる文字は”半角英数字のみ”ですので、ロケール設定は英語環境としてください

例1. tty端末設定時のスクリプト例

(1) ログオンスクリプト例

timeout=60	: タイムアウト時間の設定（60秒）
retry=3	: リトライ回数の設定（3回）
interval=60	: リトライ間隔の設定（60秒）
sleep=1	: 1秒待ち
send=	: 改行入力
sleep=1	: 1秒待ち
send=	: 改行入力
sleep=1	: 1秒待ち
wait=ogin:	: login: プロンプトを受信するまで待機※
sleep=5	: 5秒待ち
send=<ユーザ名>	: ユーザ名の送信
wait=assword:	: password: プロンプトを受信するまで待機※
sleep=5	: 5秒待ち
send=<パスワード>	: パスワードの送信
wait=\$: シェルのプロンプトを受信するまで待機
	: ※UNIX系サーバの処理系により、“login:” “password:”
	: の先頭文字が大文字になっている場合があるため、2文字目以下
	: の文字列の受信待ちをしています。

(2) シャットダウンスクリプト例

timeout=60	タイムアウト時間の設定 (60秒)
sleep=10	10秒待ち
interval=10	リトライ間隔の設定 (10秒)
retry=3	リトライ回数設定 (3回)
sleep=1	1秒待ち
send=	改行入力
sleep=1	1秒待ち
send=	改行入力
sleep=1	1秒待ち
wait=\$	シェルのプロンプトを受信するまで待機
send=export LANG=C	言語を英語に変更
sleep=1	1秒待ち
wait=\$	シェルのプロンプトを受信するまで待機
send=su -	"su" コマンドの投入
wait=assword:	password: プロンプトを受信するまで待機※1
send=<rootユーザのパスワード>	パスワードの送信
wait=#	rootユーザのシェルのプロンプトを受信するまで待機
sleep=5	5秒待ち
send=shutdown -y -g0 -i0	UNIX系サーバのシャットダウンコマンドを送信※2※3
	※1 UNIX系サーバの処理系により、“login:” “password:” の先頭文字が大文字になっている場合があるため、2文字目以降の文字列の受信待ちをしています。
	※2 最後に改行を追加しないとシャットダウンコマンドは実行されません。
	※3 シャットダウンコマンドは、Linux, S series の場合 「shutdown -y -g0 -i0」、PRIMEPOWERの場合「shutdown -y -g0 -i5」になります。

(3) 停電時のシャットダウンスクリプト例

timeout=60	タイムアウト時間の設定 (60秒)
sleep=10	10秒待ち
interval=10	リトライ間隔の設定 (10秒)
retry=3	リトライ回数設定 (3回)
sleep=1	1秒待ち
send=	改行入力
sleep=1	1秒待ち
send=	改行入力
sleep=1	1秒待ち
wait=\$	シェルのプロンプトを受信するまで待機
send=export LANG=C	言語を英語に変更
sleep=1	1秒待ち
wait=\$	シェルのプロンプトを受信するまで待機
send=su -	"su" コマンドの投入
wait=assword:	password: プロンプトを受信するまで待機※1
send=<rootユーザのパスワード>	パスワードの送信
wait=#	シェルのプロンプトを受信するまで待機
sleep=5	5秒待ち
send=shutdown -y -g0 -i0	UNIX系サーバのシャットダウンコマンドを送信※2※3

※1 UNIX系サーバの処理系により、“login:” “password:” の先頭文字が大文字になっている場合があるため、2文字目以降の文字列の受信待ちをしています。

※2 最後に改行を追加しないとシャットダウンコマンドは実行されません。

※3 シャットダウンコマンドは、Linux, S series の場合「shutdown -y -g0 -i0」、PRIMEPOWER の場合「shutdown -y -g0 -i5」になります。

例2. コンソールポート設定時のスクリプト例

(1) ログオンスクリプト例

timeout=60	タイムアウト時間の設定 (60秒)
retry=3	リトライ回数設定 (3回)
interval=60	リトライ間隔の設定 (60秒)
sleep=1	1秒待ち
send=	改行入力
sleep=1	1秒待ち
send=	改行入力
sleep=1	1秒待ち
wait=ogin:	login: プロンプトを受信するまで待機※
sleep=5	5秒待ち
send=<ユーザ名>	ユーザ名の送信
wait=assword:	password: プロンプトを受信するまで待機※
sleep=5	5秒待ち
send=<パスワード>	パスワードの送信
wait=#	シェルのプロンプトを受信するまで待機

※UNIX系サーバの処理系により、“login:” “password:” の先頭文字が大文字になっている場合があるため、2文字目以降の文字列の受信待ちをしています。

(2) シャットダウンスクリプト例

timeout=60	: タイムアウト時間の設定 (60秒)
sleep=10	: 10秒待ち
interval=10	: リトライ間隔の設定 (10秒)
retry=3	: リトライ回数の設定 (3回)
sleep=1	: 1秒待ち
send=	: 改行入力
sleep=1	: 1秒待ち
send=	: 改行入力
wait=#	: シェルのプロンプトを受信するまで待機
sleep=5	: 5秒待ち
send=shutdown -y -g0 -i0	: UNIX系サーバのシャットダウンコマンドを送信※1※2
	: ※1最後に改行を追加しないとシャットダウンコマンドは実行されません。
	: ※2シャットダウンコマンドは、Linux,S seriesの場合「shutdown -y -g0 -i0」、PRIMEPOWERの場合「shutdown -y -g0 -i5」になります。

(3) 停電時のシャットダウンスクリプト例

timeout=60	: タイムアウト時間の設定 (60秒)
sleep=10	: 10秒待ち
interval=10	: リトライ間隔の設定 (10秒)
retry=3	: リトライ回数の設定 (3回)
sleep=1	: 1秒待ち
send=	: 改行入力
sleep=1	: 1秒待ち
send=	: 改行入力
wait=#	: シェルのプロンプトを受信するまで待機
sleep=5	: 5秒待ち
send=shutdown -y -g0 -i0	: UNIX系サーバのシャットダウンコマンドを送信※1※2
	: ※1最後に改行を追加しないとシャットダウンコマンドは実行されません。
	: ※2シャットダウンコマンドは、Linux,S seriesの場合「shutdown -y -g0 -i0」、PRIMEPOWERの場合「shutdown -y -g0 -i5」になります。