

LPICレベル1

～技術解説無料セミナー～

2014/3/1

カチシステムプロダクツ株式会社
システムソリューション部
但馬 謙



■カチシステムプロダクツ株式会社

金融業界を中心にアプリケーション開発、環境構築、運用管理などのITサービスを提供しています。

資格取得推進

2009年 ITILファンデーションを全社員取得

2013年 LPICレベル1の全社員資格取得を推進

(同時にOSS-DB Silverの資格取得も推奨)

2014年 LPICおよびOSS-DBの上位資格取得を推進

■講師

システムソリューション部

これまで、金融機関を中心にしたさまざまなアプリケーション開発、環境構築などを担当してきた。現在は、部内の技術研修も担当している。



リナックス
Linuxって?



- 無料で使えるOS(オペレーティング・システム)
- あらゆる種類のオープンソースソフトウェアが無料で使える
- 用途と応用範囲は無限大
ファイルサーバ、メールサーバ、DBサーバ、ネットワーク管理など



- 1 LPICとは
- 2 LPIC-1(レベル1)試験の概要
- 3 試験範囲



■ LPICとは、「Linux技術者認定試験」

LPICには、3段階(レベル1~3)のレベルがある。

LPIC-1は「サーバ運用管理レベル」

LPIC-2は「ネットワークシステム構築レベル」

LPIC-3は「エンタープライズシステム構築レベル」

LPICは、世界標準資格

ベンダーやディストリビューションに依存しない

● ディストリビューションとは

カーネルに加えて、コマンドやアプリケーションを用意し、Linuxをシステムとして、すぐに利用できるようにパッケージ化したもの。

RedHat系: Fedora・Red Hat Enterprise Linux・CentOS

Debian系: Debian GNU/Linux・Ubuntu Linux



<認定要件>

- ・101試験と102試験両方に合格が必要
 - ・試験は、同日、別日どちらでも問題ない。
 - ・受験の順番は、どちらからでもOK！

<101・102試験概要>

- ・問題数:60問
15問～20問がキーボード入力ですが
その他は四者択一方式などの選択問題です。
- ・試験時間:90分
- ・合格基準:得点200～800点のうち500点以上で合格



3 試験範囲(101試験)

主題		内容
101	システムアーキテクチャ	<ul style="list-style-type: none">・ハードウェア設定の決定と構成・システムのブート・ランレベルの変更とシステムのシャットダウンまたはリブート
102	Linuxのインストールとパッケージ管理	<ul style="list-style-type: none">・ハードディスクのレイアウト設計・ブートマネージャのインストール・共有ライブラリを管理する・Debianパッケージ管理を使用する・RPMおよびYUMパッケージ管理を使用する
103	GNUとUnixのコマンド	<ul style="list-style-type: none">・コマンドラインで操作する・フィルタを使ってテキストストリームを処理する・基本的なファイル管理を行う・ストリーム、パイプ、リダイレクトを使う・プロセスを生成、監視、終了する・プロセスの実行優先度を変更する・正規表現を使用してテキストファイルを検索する・viを使って基本的なファイル編集を行う
104	デバイス、Linuxファイルシステム、ファイルシステム階層標準	<ul style="list-style-type: none">・パーティとファイルシステムの作成・ファイルシステムの整合性を保持する・ファイルシステムのマウントとアンマウントをコントロールする・ディスククォータを管理する・ファイルのパーミッションと所有者を管理する・ハードリンクとシンボリックリンクを作成・変更する・システムファイルを見つける、適切な位置にファイルを配置する



3 試験範囲(102試験)

主題		内容
105	シェル、スクリプト、およびデータ管理	<ul style="list-style-type: none">・ シェル環境のカスタマイズと使用・ 簡単なスクリプトをカスタマイズまたは作成する・ SQLデータ管理
106	ユーザインターフェイスとデスクトップ	<ul style="list-style-type: none">・ X11のインストールと設定・ ディスプレイマネージャの設定・ アクセシビリティ
107	管理業務	<ul style="list-style-type: none">・ ユーザアカウント、グループアカウント、および関連するシステムファイルを管理する・ ジョブスケジューリングによるシステム管理業務の自動化・ ローカライゼーションと国際化
108	重要なシステムサービス	<ul style="list-style-type: none">・ システム時刻を維持する・ システムのログ・ メール転送エージェント(MTA)の基本・ プリンターと印刷を管理する
109	ネットワークの基礎	<ul style="list-style-type: none">・ インターネットプロトコルの基礎・ 基本的なネットワーク構成・ 基本的なネットワークの問題解決・ クライアント側のDNS設定
110	セキュリティ	<ul style="list-style-type: none">・ セキュリティ管理業務を実施する・ ホストのセキュリティ設定・ 暗号化によるデータの保護



- 1. システム・アーキテクチャ(主題101)
- 2. 管理業務(主題107)



■ 主題101 システム・アーキテクチャ



■ハードウェア接続形態

●PCI

使用例: ネットワークカード

●IDE

使用例: 内蔵HDD

●SCSI

使用例: 内蔵HDD・CD-ROMドライブ・スキャナ・プリンタ

●SATA (Serial ATA)

IDEの後継

使用例: 内蔵HDD・CD-ROMドライブ

●SAS (Serial Attached SCSI)

SCSIの後継

使用例: 内蔵HDD・CD-ROMドライブ

●USB (Universal Serial Bus)

使用例: 周辺機器(キーボード・マウス・USBメモリ等)



■ハードウェア情報の確認

システムに接続されるデバイスに接続するには、「/dev」ディレクトリ以下に格納されているデバイスファイルを使ってアクセス可能。

システムが認識しているデバイスの情報は、「/proc」ディレクトリにて確認可能。

説明	ファイル名
USB情報	<code>/proc/bus/usb/devices</code>
CPU情報	<code>/proc/cpuinfo</code>
メモリ情報	<code>/proc/meminfo</code>
I/Oポートアドレス情報	<code>/proc/ioports</code>
IRQ情報	<code>/proc/interrupts</code>
PCI情報	<code>/proc/pci</code>
SCSI情報	<code>/proc/scsi/scsi</code>
DMA情報	<code>/proc/dma</code>



<IRQ>

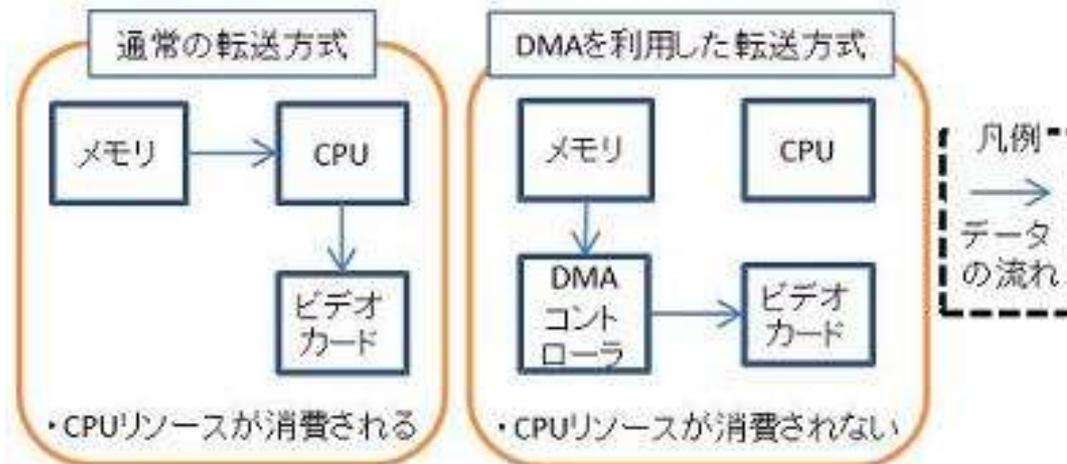
CPUがデバイスにアクセスする際に、どのデバイスからの要求かの識別番号
状況確認には、「/proc/interrupts」ファイル

<I/Oポートアドレス>

CPUがデバイスにデータを送る際に指定するデバイス番号
状況確認には、「/proc/ioports」ファイル

<DMAチャネル>

CPUを中継せずに、周辺機器とメモリをやりとりするDMAの
経路を指定したもの。状況確認には、「/proc/dma」ファイル





・現在のデバイス情報を確認するコマンド

カーネルの初期化時に認識されたデバイスのリストとその設定が表示される。

詳細な情報を確認したい場合は、「-v」オプション

表示する情報	コマンド名
PCIデバイス	lspci
USBデバイス	lsusb

lspciの結果

```
[root@localhost ~]# lspci
00:00.0 Host bridge: Intel Corporation 440BX/ZX/DX - 82443BX/ZX/DX
Host bridge (rev 01)
00:01.0 PCI bridge: Intel Corporation 440BX/ZX/DX - 82443BX/ZX/DX
AGP bridge (rev 01)
00:07.0 ISA bridge: Intel Corporation 82371AB/EB/MB PIIX4 ISA (rev 08)
00:07.1 IDE interface: Intel Corporation 82371AB/EB/MB PIIX4 IDE (rev
01)
00:07.3 Bridge: Intel Corporation 82371AB/EB/MB PIIX4 ACPI (rev 08)
00:07.7 System peripheral: VMware Virtual Machine Communication
Interface (rev 10)
(以下省略)
```

lsusbの結果

```
[root@localhost ~]# lsusb
Protocol spec without prior Class and Subclass spec at line 4297
Bus 001 Device 001: ID 0000:0000
Bus 002 Device 001: ID 0000:0000
```



● デバイスドライバの管理

- ・システムにロードされているデバイスドライバ(制御用モジュール)を確認するには、
【lsmod】コマンドか【/proc/modules】ファイルを確認する。
- ・デバイスドライバを依存関係を考慮して、デバイスドライバのロード・アンロードするには、【modprobe】コマンド



●ホットプラグデバイス

- ・コンピュータの電源が入っている場合にデバイスを取り外すことができるデバイスを「ホットプラグデバイス」
- ・ホットプラグデバイスの制御には、「udev」によって行われる。
- ・「udev」は、「/etc/udev」ディレクトリ内のプログラムを読み込む。
- ・ホットプラグの例:USB

●コールドプラグデバイス

- ・コンピュータの電源が入っていない場合にデバイスを取り外すことができるデバイスを「コールドプラグデバイス」
- ・システム起動時にデバイスを認識する
- ・コールドプラグの例:ネットワークカード



- ・USBデバイスはUSBコントローラ(チップ)によって制御される。
- ・USBコントローラにはモジュール(デバイスドライバ)が必要。

■USBモジュールとコントローラ一覧

USBコントローラ	規格	USBモジュール (カーネル2.4)	USBモジュール (カーネル2.6)
UHCI	USB1.1	Usb-uhci	Uhci_hcd
OHCI	USB1.1	Usb-ohci	Ohci_hcd
EHCI	USB2.0	サポートなし	Ehci_hcd

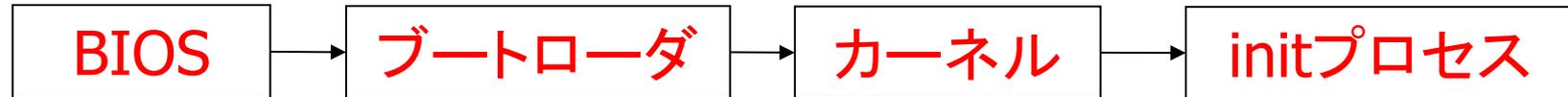
- ・USB1.1規格はカーネル2.4からサポート開始
- ・USB2.0規格はカーネル2.6からサポート開始
- ※カーネル2.2は、USBをサポートしていない。

●USBの特徴

- ・USBコントローラは最大127台のUSBデバイスを制御できる
- ・システムが起動している状態でもUSBデバイスの差し替えが出来る(ホットプラグ)
- ・接続しているホストから電源を供給できる
- ・USB1.1とUSB2.0は、データ転送速度が異なる



1-2-1 システムの起動手順



●BIOSの役割

- ・日時の設定
- ・周辺機器の有効化・無効化
- ・記憶装置(HDD)の最低限の認識
- ・デバイスの起動順序の設定
- ・MBR(マスターブートレコード)を読み込み、ブートローダに制御を移す。



●ブートローダの役割

- ・HDDのカーネルをメモリにロードし、カーネルに制御を移す。
- ・ブートローダーからカーネルに渡される設定は、「**/proc/cmdline**」に記述される。

●カーネルの役割

- ・デバイスを起動するための準備
- ・Linuxを起動するための初期化处理
- ・initプロセスを起動
- ・カーネルモジュールをロードする際は、以下のファイルの設定を参照
 - カーネル2.4系では、「**/etc/modules.conf**」
 - カーネル2.6系では、「**/etc/modprobe.conf**」



●initの役割

- ・システム起動に必要なプログラムを起動。
- ・「**/etc/inittab**」ファイルの設定を読み込んで起動する。
- ・「**/etc/inittab**」ファイルには、「Linuxの起動時の動作モード(ランレベル)」・「サーバ起動時に起動するプログラムの指定」が含まれる。(ランレベルについては、後述)

1-2-2 起動時(ブート時)のイベント確認

- ・【**dmesg**】コマンドを実行する
- ・「**/var/log/messages**」や「**/var/log/boot.log**」や「**/var/log/dmesg**」を確認する。
- ・CPU・ディスク・ネットワークなどの情報を確認可能。



1-3-1 ランレベルの種類

・ランレベルとは、システムの動作モードのこと

●シングルユーザモード

- ・管理者(root)のみが作業可能な動作モード
- ・ルートパーティション以外のファイルシステムは、マウント(認識)しない。
- ・用途としては、**メンテナンス用**で、主にパスワード再設定

●マルチユーザモード

- ・どのユーザでもログインできる
- ・テキストログイン(CUI画面を起動)
- ・グラフィカルログイン(GUI画面を起動)

ディスプレイマネージャプログラムがログイン画面を表示



■ランレベル一覧

ランレベル	機能
0	停止
1/s/S/single	シングルユーザモード
2	マルチユーザモード(テキストログイン・NFS無効)
3	マルチユーザモード(テキストログイン)
4	未使用
5	マルチユーザモード(グラフィカルログイン)
6	再起動

1-3-2 /etc/init.dディレクトリ

・initプロセスは、ランレベルに従い「/etc/rc[ランレベル].dディレクトリ」の起動スクリプトが実行される。



KとSの скриптがあり、Kから始まる скриптは、ランレベルで起動した際に停止します。

Sから始まる скриптは、ランレベルで起動した際に実行します。

/etc/rc2.dにランレベル2の起動 скриптが記述される。

■ /etc/rc2.dの一覧

```
# ls /etc/rc2.d
K03samba*          S20syssetup*      S72slpd*          S88utmpd*
K06mipagent*       S21perf*          S73cachefs.daemon* S89PRESERVE*
K07dmi*            S30sysid.net*     S73nfs.client*    S89bdconfig@
K07snmpdx*         S40llc2*          S74autofs*        S90wbem*
K16apache*         S42ncakmod*       S74syslog*        S91afbinit*
K21dhcp*           S47pppd*          S74xntpd*         S91gfbinit*
K26sshd*           S69inet*          S75cron*          S91ifbinit*
K27boot.server*   S70sckm*          S75flashprom*     S92volmgt*
K28kdc*            S70uucp*          S75savecore*      S93cacheos.finish*
K28kdc.master*    S71ldap.client*   S76nsd*            S94ncalogd*
K28nfs.server*    S71rpc*           S77sf880dr*       S95IIim*
README             S71sysid.sys*     S80lp*             S95svm.sync*
S01MOUNTFSYS*     S72autoinstall*  S80spc*            S98efcode*
S05RMTMPFILES*    S72directory@    S85power*          S99audit*
S10lu*            S72inetsvc*      S88sendmail*      S99dtlogin*
```



1-3-3 デフォルトのランレベルの変更

- ・「/etc/inittab」ファイルの「id:5:inittdefault:」の数字の部分を変更することで起動時のデフォルトランレベルの変更可能
- ・デフォルトのランレベルとしては、**停止(0)**・**未使用(4)**・**再起動(6)**は、利用できない。
- ・「/etc/inittab」では、**「Ctrl+Alt+Delete」**を押した際の再起動を無効化可能。

1-3-4 ランレベルの確認と変更

- ・ランレベルを確認するには、【runlevel】コマンド
実行すると一つ前のランレベルと現在のランレベルが確認できる。
「N」と表示した場合、起動してから一度もランレベルを変更していないとわかる。

```
[root@localhost ~]# runlevel  
N 3
```

- ・ランレベルを変更するには、【telinit】・【init】コマンド
ランレベルを変更できるのは、管理者(root)のみ
システム稼働中でも必要に応じてランレベルは変更でき、再起動は不要。



1-3-5 システムのシャットダウン

- ・システムの再起動やシャットダウンは、【shutdown】コマンド
- ・シャットダウンの際に、ログインしているユーザにメッセージを送信することが可能。

shutdown [オプション] 時間 [メッセージ]

オプション	説明
シャットダウン後にシステムを停止	-h
シャットダウン後にシステムを再起動	-r
シャットダウンせずにメッセージの通知のみを行う。	-k
実行中のシャットダウンをキャンセル	-c

時間指定の方法

オプション	説明
HH時MM分に実行	HH:MM
M分後に実行	+M
今すぐに実行	now

- ・再起動の方法としては、【reboot】コマンドでも可能。
(ただし、時間指定・メッセージ送信はできない)

※ログインしているユーザにメッセージを送信できるコマンドに【wall】コマンド



■ shutdownコマンドの例

● shutdown -h +5

今から5分後にシステムを停止する。

● shutdown -r 22:00 "System will restart 22:00"

ログインしているユーザに「System will restart 22:00」のメッセージを送り、22:00にシステムを再起動する。

● shutdown -r now

今すぐにシステムを再起動する。



1-3-6 Upstartとsystemd

- ・これまでは、SysVinitという起動の仕組みを使っていたが、最近では、Upstart とSystemdが使われるようになった。
- ・理由は、SysVinitは、サービスの並列起動ができないので、起動が遅い。
- ・initプログラムは、「/etc/inittab」を設定ファイルとして使用するが、Upstart とsystemdは、「/etc/inittab」は使わない。
- ・「Upstart」とは、「init」の改良版で、SysVinitと互換性はある。初期化を並行実行するなど、高速化が図られた。ただし、初期化時の依存関係を定義する必要があり、依存関係の連鎖によって、不要なプログラムを起動してしまう欠点がある。
- ・「systemd」とは、Upstartの欠点の改良版。依存関係のプログラムが必要か否かを判断し、必要に応じて起動する。Upstartよりも高速。



■ 主題107 管理業務



2-1-1 ユーザ用設定ファイル

- ・ユーザの設定は、「/etc/passwd」ファイルに格納される。

```
hidakak:x:199:300:Kensuke Hidaka :/home/hidakak:/bin/bash
```

ユーザ名	パスワード	UID	GID	コメント	ホームディレクトリ	ログインシェル
hidakak	x	199	300	Kensuke Hidaka	/home/hidakak	/bin/bash

もともとはGECOSを実行する汎用機が使用するログイン情報が格納されていたため、GECOSフィールドとも呼ばれる

rootは必ず0、1~99はシステム管理用、一般ユーザは通常100以降の番号を割り当てる

パスワード
シャドウパスワードが使われている場合、実際のパスワードは/etc/shadowに格納されるため、/etc/passwdファイルにはxと記述する



2-1-2 グループ設定ファイル

- ・グループの設定は、「/etc/group」ファイルに格納される。

```
students:x:501:suzuki,yamamoto
```

グループ名 ↑ GID グループのメンバ

パスワード

シャドウパスワードが使われている場合、実際のパスワードは
/etc/gshadowに格納されるため、/etc/groupファイルにはxと記述する

- ・ユーザは、複数のグループメンバになれる

■プライマリグループ

ファイルやディレクトリを作成した時、所有グループになる。

「/etc/passwd」のGIDフィールドで確認できる。

■セカンダリグループ

プライマリグループ以外に参加しているグループ



2-1-3 シャドーパスワード

- ・「/etc/passwd」ファイルは、パスワードを格納するファイルだが、以下の点が問題
 - 1 管理者以外でも読める
 - 2 パスワードが暗号化されない
- ・「/etc/shadow」ファイルは、管理者しか読めない・パスワードを暗号化(シャドウ化)される。
- ・「/etc/passwd」「/etc/shadow」それぞれのファイルでアクセス権が異なる
 - rw-r--r-- /etc/passwd
 - r----- /etc/shadow
- ・暗号化されていないパスワードを暗号化(シャドウ化)するには、**【pwconv】**コマンド
暗号化したパスワードを暗号化から解除するには、**【pwunconv】**コマンド
管理者(スーパーユーザ)のみ実行可能。



2-1-4 ユーザアカウント管理

- ・ユーザアカウント管理のコマンドは、管理者しか実行できない。
- ・ユーザを新規で登録するには、【useradd】コマンド

利用用途	オプション
デフォルトの値の設定あるいは、表示	-D
パスワードの使用期限が切れてから、アカウントが失効になるまでの日数(パスワードの変更周期の設定)	-f(-Dと併用)
アカウントの有効期限	-e(-Dと併用)

■ useraddの例(アカウント作成時の設定)

- パスワードの変更周期を60日に設定する

```
useradd -D -f 60
```

- アカウントが2012年12月31日に失効(アカウントの有効期限切れ)する設定

```
useradd -D -e 2012/12/31
```



- ・ **【useradd】**コマンドを実行すると「`/etc/passwd`」・「`/etc/shadow`」へのユーザ追加ならびに、ホームディレクトリを作成する。
パスワードを設定しないとパスワードフィールドが「!!」と表示される。
解消するには、**【passwd】**コマンドを利用して、パスワード設定が必要。
- ・ **【useradd】**コマンドを実行すると「`/etc/skel`」ディレクトリ内のファイルをコピーする
全ユーザに共通の設定ファイル(ドライバ)・ソフトウェア等を置いておく際に利用する。



- ・ユーザ情報を変更する場合は、【usermod】コマンド
ユーザ名・ユーザID・パスワードが変更可能。
その他、アカウントのロックが可能。一時的に使用不可にできる。

利用用途	オプション
ホームディレクトリを指定	-d
ログインシェルを指定	-s
パスワードをロック	-L
パスワードのアンロック	-U

- ・ユーザアカウントを削除する場合は、【userdel】コマンド
-rオプションで、ホームディレクトリも削除
- ・ログインパスワードを変更する場合は、【passwd】コマンド
【usermod】と同様で、アカウントのロック・アンロックが可能。
(passswd -l、passswd -u)



- ・パスワードの有効期限(パスワードを最後に更新してから、次に変更しなければならぬ期日までの日数)を確認・変更する場合は、【chage】コマンド
【chage -E】【passwd -x】【usermod -e】コマンドでも有効期限を変更可能。
有効期限になると、アカウントがロックされる。
【chage】コマンドでは、パスワードの変更はできない。

利用用途	オプション
パスワードの有効期限に関する情報 [ls]	-l
パスワード変更の間隔の最短日数 [minimum] (パスワードが変更できるまでの日数)	-m
パスワードが有効な最長日数を指定[Max] (何日ごとに必ずパスワードを変更させるか)	-M
パスワードの有効期限切れ後にアカウントがロックされるまでの日数[Inactive]	-I
パスワードの有効期限が切れる前に警告を表示する日数を指定 [Warning]	-W
ユーザーアカウントの有効期限を指定[Expire]	-E



- ・ユーザのユーザID・グループID・グループ名を確認する場合は、**【id】**コマンド

利用用途	オプション
ユーザIDと所属するすべてのグループIDを表示	なし
ユーザIDを表示	-u
プライマリグループIDを表示	-g
所属するすべてのグループIDを表示	-G

- ・一般ユーザのログインを禁止するには、「**/etc/nologin**」または、「**/etc/false**」ファイルを作成する



2-1-5 グループアカウント管理

- ・グループアカウント管理のコマンドは、管理者しか実行できない。
- ・ユーザの所属しているグループを表示するには、【groups】コマンド
- ・グループを新規で登録するには、【groupadd】コマンド
- ・すでに存在しているグループアカウントの情報を変更するには、
【groupmod】コマンド
- ・グループアカウントを削除する場合は、【groupdel】コマンド
削除するグループがプライマリグループにしている人がいたら削除できない。



2-2-1 ジョブスケジューリング

・決まった時間にコマンドやシェルスクリプト(ジョブ)を実行する。

■ cron

定期的に行う

(具体的には、cronデーモンにより/usr/sbin/anacronが実行)

crontabファイルにスケジュールを記述

■ at

一回だけ実行する

■ batch

システムの負荷が低くなったときに一回だけ実行する。



2-2-2 crontabファイル・crontabコマンド

・ジョブスケジュールを記述するファイルをcrontabファイル

■ユーザごと

/var/spool/cronディレクトリ

crontabコマンド(一般ユーザが実行可能)

■システムごと

/etc/crontabファイル(システム管理者のみ設定権限あり)

/etc/cron.*ディレクトリを実行

毎時・毎日・毎週・毎月自動タスク設定が可能

```
# run-parts
01 * * * * root run-parts /etc/cron.hourly
02 4 * * * root run-parts /etc/cron.daily
22 4 * * 0 root run-parts /etc/cron.weekly
42 4 1 * * root run-parts /etc/cron.monthly
```

※/etc/cron.dayというディレクトリは存在しません。

/etc/cron.dディレクトリには、上記以外の自動タスク設定ファイルを置く。



2-2-3 crontabコマンドの設定・crontabファイルの変更

・crontabタブコマンドの設定

【crontab -e】コマンド

各ユーザが設定可能。

・crontabファイルの変更

/etc/crontabを編集

管理者のみが設定可能

01	23	*	*	*	/usr/local/bin/backup.sh
分	時	日	月	曜日	コマンド

・曜日は、0～7で0または、7が日曜日

・複数の値を指定するには、「,」で区切る

● 毎時15分と45分に/usr/local/bin/backup.shを起動

```
15,45 * * * * /usr/local/bin/backup.sh
```

・〇分(〇時)おきに実行

● 20分おきに/usr/local/bin/backup.shを起動

```
*/20 * * * * /usr/local/bin/backup.sh
```



2-2-4 crontabのアクセス制御

- ・crontabを利用できるユーザは、以下のとおり
 1. /etc/cron.allowがある
記述されたユーザは、crontab利用可能
記述されていないユーザは、利用不可
 2. /etc/cron.allowがない。かつ/etc/cron.denyがある
記述されたユーザは、crontab利用不可
記述されていないユーザは、利用可能
 3. /etc/cron.allow・/etc/cron.denyどちらもない
全員crontab利用可能



2-2-5 ジョブの登録

- ・一回だけ実行するジョブをスケジューリングするには、【at】コマンド
- ・書式は「at [オプション...] [実行日時]」
- ・日付は、**MMDDYY**かMM/DD/YY形式で指定
- ・時刻は、hh:mmで指定

2-2-6 ジョブの確認

- ・atコマンドで設定されたジョブを確認するには、【atq】コマンド
【at -l】コマンドでも可能

2-2-7 ジョブの削除

- ・atコマンドで設定されたジョブを削除するには、【atrm】コマンド
【at -d】コマンドでも可能



2-2-8 atコマンドのアクセス制御

・atを利用できるユーザは、以下のとおり

1. /etc/at.allowがある

記述されたユーザは、atコマンド利用可能

記述されていないユーザは、利用不可

2. /etc/at.allowがない。かつ/etc/at.denyがある

記述されたユーザは、atコマンド利用不可

記述されていないユーザは、利用可能

3. /etc/at.allow・/etc/at.denyどちらもない

管理者(スーパーユーザ)だけがatコマンドを実行可能(crontabとの違い)



2-2 ジョブスケジューリングによる システム管理業務の自動化(重要度:4)

ここで時間の関係から、重要度4「ジョブスケジューリングによるシステム管理業務の自動化」のまとめについてのみ、講師の学習の進め方について例示いたします。よろしければ参考にしてください。



「ジョブスケジューリングによるシステム管理業務の自動化」についての学習は(LPIC全般について言えますが)ユーザーがLinuxに対して行いたい業務に対して、OSのインターフェースがどう対応するかを理解することになります。

業務側は、このインターフェースに対応すべく、業務内容を以下のような概念で細分化します。

- ・種類
- ・使用属性

種類は、業務などにおける内容を示すための種類の事です。また、**属性**は、例えば、当該コマンドの実行にあたり、行われるべきことの種類の事です。そして、この業務内容の細分化は直観的に理解できます。

この細分化された業務内容に対応する形で、以下のようなOSのインターフェースがあることになります。

- ・コマンドによるもの
- ・ファイル(およびその**記述内容**)によるもの
- ・さらに該当のコマンドやファイルに対して、**アクセス制御**を伴うもの。

また、**コマンド**には、その実行の使用**属性**を表現する**オプション**と**デフォルト**があります。**デフォルト**とは**オプション**を記述しないことです。これらの概念は、直観的とは限りませんので、記憶する対象になります。

以下では、上記の**青い文字**の概念が含まれる項目については、**赤い文字**でテキストボックスに記述します。**緑の文字**は、業務における項目が細分化されないままインターフェースの記述に取り込まれることを示します。また、テキストボックスの枠の色で、**青**が直観的に把握できるもの、**オレンジ**が直観+記憶が必要なもの、**赤**が記憶しなければならないものとして表記します。



2-2 ジョブスケジューリングによる システム管理業務の自動化(重要度:4)

■ 業務内容

「ジョブスケジューリング」とは「決まった**タイミング**に**コマンド**や**シェルスクリプト**(ジョブ)を実行する」ことです。

■ **タイミング**には当然以下の**種類**があります。

- ・**毎回**: 月,日,時,分,曜のどれかの組み合わせにより、確定できる。
- ・**1回**: 年,月,日,時,分の全てを組み合わせ、確定できる。

■ 実行する**コマンド**等

- ・**毎回**のタイミング⇒「cron」(≒「**crontab**」)
- ・**1回**のタイミング ⇒システム負荷に関し条件が任意の場合は「**at**」
システム負荷が低い場合にのみ起動したい場合は「**batch**」

■ 「**crontab**」コマンド等には、以下の**種類**があります。

- ・**一般ユーザーが実行するもの**。
- ・**システムごとに実行するもの**。

■ **一般ユーザーが実行するもの(コマンド)**

「**crontab -e**」コマンドで「/var/spool/cron/」ディレクトリに設定を格納します。

■ **システムごとに実行するもの(ファイル設定)**

システム管理者が「/etc/crontab」**ファイル**を設定することで実施されます。

■ 「**at**」コマンドの使用**属性**には、以下のものがあります。

デフォルトと**オプション**で表現されるもの

- ・**登録**: デフォルト「at <タイミング>」
 - ・**確認**: オプション([l]ist)「at -l」 ジョブ番号も確認できます。
 - ・**削除**: オプション([d]elete)「at -d ジョブ番号」
- 「**at**」コマンドの別**コマンド**で表現されるもの

- ・**登録**: -
- ・**確認**: 「atq」
- ・**削除**: 「atrm ジョブ番号」

実行する**コマンド**や**シェルスクリプト**は標準入力で指定します。
何かを設定するコマンドの一つの使用**属性**は以下になります。

<**登録**, **確認**, **削除**>



■ **毎回のタイミング**には、以下のような**種類**があります。
当該タイミングで**実行するファイル**(以下のディレクトリに格納)

- ・**毎時** :「/etc/cron.hourly/」
- ・**毎日** :「/etc/cron.daily/」
- ・**毎週** :「/etc/cron.weekly/」
- ・**毎月** :「/etc/cron.monthly/」
- ・**その他**:「/etc/cron.d/」

「/etc/crontab」**記述内容例**

- ・**毎時** :「01 * * * *」(つまり時から「*」とする)
- ・**毎日** :「01 4 * * *」(つまり日から「*」とする)
- ・**毎週** :「01 4 * * 0」(つまり上記に曜日を指定する)
- ・**毎月** :「01 4 1 * *」(つまり上記に日を指定する)
- ・**その他**: (つまり上記以外の場合である)

■ 「at」コマンドのタイミングについての**記述形式(書式)**
年月日:「MMDDYY」または「MM/DD/YY」(アメリカ式)
時分 :「hh:mm」です。
年月日の表し方が「cron」のイギリス式とは異なりますので、注意してください。

■ 「crontab」コマンドにおける**記述形式(書式)**

「<タイミング> <コマンドやシェルスクリプト>」

<タイミング>についての留意事項(ヒント)

- ・月日の並びはイギリス式(発音時ofを補完)になっているために、この並びである。
- ・4つ目の「*」が月を意味するから、毎月になる。

■ crontabのタイミングにおける**略記の種類と記述形式(書式)**

- ・**範囲** :「A-B」と記述します。
- ・**リスト**:「A,B,C」と記述します。
- ・**~毎** :「/ A」と記述します。

したがって、時に「1-5/2」と指定した場合は1時、3時、5時に処理を実行します。



■ 「crontab」コマンドの一般ユーザーが実行するものに関するアクセス制御

(ファイルとその記述内容によって実現)

・cronの一般ユーザ利用可

- 「/etc/cron.allow」ファイルに記述があるユーザー
- 「/etc/cron.allow」ファイルがない場合、「/etc/cron.deny」ファイルに記述がないユーザー
- 「/etc/cron.allow」ファイルと「/etc/cron.deny」ファイルが共にない場合、全てのユーザー

・cronの一般ユーザ利用不可

- 「/etc/cron.allow」ファイルに記述がないユーザー
- 「/etc/cron.allow」ファイルがない場合、「/etc/cron.deny」ファイルに記述があるユーザー

■ 「at」コマンドの一般ユーザに関するアクセス制御[つまり、システム管理者は、いつでも「at」コマンドを使用できる]

(ファイルとその記述内容によって実現)

・atの一般ユーザ利用可

- 「/etc/at.allow」ファイルに記述があるユーザー
- 「/etc/at.allow」ファイルがない場合、「/etc/at.deny」ファイルに記述がないユーザー

・atの一般ユーザ利用不可

- 「/etc/at.allow」ファイルに記述がないユーザー
- 「/etc/at.allow」ファイルがない場合、「/etc/at.deny」ファイルに記述があるユーザー
- 「/etc/at.allow」ファイルと「/etc/at.deny」ファイルが共にない場合、システム管理者のみ

■ アクセス制御についての留意事項(ヒント)

- ・一般ユーザ利用可不可の内容は、「記述がある」「記述がない」を反転させたものになる。
- ・許可ファイルと拒否ファイルが共にない場合の扱いについて、「cron」と「at」で逆転する。



2-3-1 ローカライゼーション

- ・表示言語や通貨単位・時刻などの書式を国や地位によって合わせることをローカライゼーションという。
- ・最近のOSは、ローカライゼーションを使って、複数の言語や地域に対応している。

2-3-2 ロケール

- ・利用者の地域情報のことをロケールという。
- ・設定するロケール毎に環境変数が定まっている。



2-3-3 ロケールの環境変数

環境変数	説明
LANG	言語の設定 LC_から始まる環境変数のデフォルト値。 LCから始まる環境変数に指定がない場合は、LANGの値が適用される。
LC_CTYPE	文字の種類を指定
LC_MESSAGE	出力されるメッセージの言語を指定
LC_MONETARY	通貨管理を指定
LC_NUMERIC	数値の入出力に使用する形式を指定
LC_TIME	時刻形式に各国語を指定
LC_ALL	ロケールの環境変数すべて ※この設定をするとLC_から始まる環境変数は、すべてこの値に設定され、他の設定に変更できなくなる。 それぞれ設定したい場合は、この設定はしないこと。



2-3-4 ロケールの確認

- ・現在のロケール設定を表示するには、【locale】コマンド
ロケールに設定できる値を確認するには、【locale -a】コマンド

```
[root@localhost ~]# locale
LANG=ja_JP.UTF-8
LC_CTYPE="ja_JP.UTF-8"
LC_NUMERIC="ja_JP.UTF-8"
LC_TIME="ja_JP.UTF-8"
LC_COLLATE="ja_JP.UTF-8"
(以下省略)
```

ロケールの値	説明
ja_JP.UTF-8	日本語 (UTF8)
ja_JP.shiftJIS	日本語 (ShiftJIS)
ja_JP.eucJP	日本語 (EUC-JP)
C	英語
fr_FR.ISO8859-1	フランス語
pt_PT.ISO8859-1	ポルトガル語



2-3-5 文字コード

文字コード	説明
ASCII	7ビットで英数字を表現。128種類の文字
ISO-8859	ASCIIを拡張し、8ビットで英数字を表現。256文字
UTF-8	Unicodeを使った文字コードで、Linuxの標準文字コード
日本語EUC(EUC-JP)	UNIX環境で標準の日本語文字コード
Shift_JIS	WindowsやMacOSで標準の日本語文字コード
ISO-2022-JP	電子メールで利用される。JISで定められた日本語文字コード
Unicode	1つの文字コード体系で多国語処理を可能にするための文字コード 当初はすべての文字を16ビットで表現していたが、現在32ビットで表現。 LinuxやWindowsなど、さまざまなOSで利用可能。

2-3-6 文字コードの変換

- ・文字コードの変換するには、【iconv】コマンド

説明	オプション
変換前の文字コードの指定	-f 入力文字コード
変換後の文字コードの指定	-t 出力文字コード
取扱い可能な文字コードの表示	-l

- EUC-JPで保存されたaaa.txtをUTF-8に変換して、bbb.txtを作成

```
[root@localhost~]# iconv -f eucjp -t utf8 aaa.txt > bbb.txt
```



2-3-7 タイムゾーン

- ・地域ごとに区切られた標準時間帯をタイムゾーンという。
- ・日本標準時は、グリニッジ標準時間(協定世界時:UTC)より9時間早い。
- ・タイムゾーン情報は、**/usr/share/zoneinfo**ディレクトリ以下のバイナリファイルに格納

2-3-8 タイムゾーンの変更

- ・タイムゾーンの変更するには、**/usr/share/zoneinfo**ディレクトリのファイルを**/etc/localtime**ファイルにコピーする
 - 日本のタイムゾーンファイル

```
[root@localhost ~]# cp /usr/share/zoneinfo/Asia/Tokyo /etc/localtime
```

```
[root@localhost ~]# ln -s cp /usr/share/zoneinfo/Asia/Tokyo /etc/localtime
```

- ・ディストリビューションによっては、環境変数TZをみる
- ・**/etc/timezone**に記述しておけば、全ユーザが環境変数TZの値が同じになる。
- ・**【tzselect】**コマンドで、対話形式で一覧からタイムゾーンを選択できる。
- ・**【tzconfig】**コマンドで、対話形式で「**/etc/localtime**」と「**/etc/timezone**」をまとめて変更できる。



ご清聴ありがとうございました。