LPICレベル1技術解説無料セミナー

2016/1/30

パナソニック ソリューションテクノロジー(株)
システム三部コンテンツ開発課二係
市川雅士

パナソニック ソリューションテクノロジー(株)

Panasonic
Solution
Technologies

パナソニック ソリューションテクノロジーは、パナソニックグループにあって、B2Bクラウドを基軸にお客様の業務課題に密着したICTソリューションの提供を行う会社です。
最新の技術を用いた信頼性・可用性の高い最適なシステム環境の設計・構築・運用サービス、独自技術で進化を続けるデータ検索エンジンやOCR技術、パナソニックグループでも活用される自社開発のソフトウェア製品群や教育サービス・コンテンツ、各種モバイルソリューションなど、多様な技術とソリューションを保有・展開しています。
わたしたちは、お客様それぞれの業務課題と向き合いながら、25年以上の実績があるこれらのICTソリューションを先進のシステム＆クラウド技術と組み合わせ、オフィスビジネス分野、フィールドモバイル分野、教育ICT分野、ライフサポート分野、コマース分野などの業務シーンに合わせた最適なソリューションを構築し、お客様の事業をサポートし続けます。

社名	パナソニック ソリューションテクノロジー株式会社
英文社名	Panasonic Solution Technologies Co., Ltd.
本社所在地	〒105-0021 東京都港区東新橋2-12-7 住友東新橋ビル2号館
拠点	東京：住友東新橋ビル2号館・浜松町ビル・住友浜松町ビル 大阪：OBPパナソニックタワー 九州：福岡パナソニックビル
代表取締役社長	小河　寿
取締役副社長	福地　孝志
設立	1988年12月1日
資本金	1億円（パナソニック株式会社 全額出資）

パナソニック ソリューションテクノロジー

コーポレート・メッセージ

Linux Professional Institute

ITソリューションラインアップ

▼詳しいソリューションのご紹介
http://panasonic.co.jp/pstc/products/

Panasonic Solution Technologies

オフィス業務プロセス革新ソリューション

コンテンツ管理・活用ソリューション

文書管理
- 機密文書管理
- 原本性保証文書管理

プロジェクト文書管理
- 開発文書
- 契約文書

ワークフロー
- 決裁/承認システム

コンテンツ管理
- エンタープライズコンテンツマネジメントシステム
- Webコンテンツ管理（CMS）

業務効率化ソリューション

OCR
- OCRパッケージソフト
- 帳票OCR／複合機連携OCR

コミュニケーションソリューション

- Web会議
- ポータルサイト
- グループウェア

ソフトウェアエンジン

- OCR
- グループウェア
- 全文検索

ID管理

- 統合ID管理

クラウドソリューション

SaaS

- Web会議サービス
- グループウェアサービス
- CMSサービス
- 特許調査支援サービス

DaaS

- 仮想デスクトップサービス

IaaS

- レンタル仮想サーバサービス

ITインフラソリューション

BCPソリューション（事業継続）

- FalconStor CDP（高速バックアップ・リストア）設計・構築
- 遠隔バックアップ・システム設計・構築
- クラスタリング・システム設計・構築

ITインフラ基盤ソリューション

- インターネット環境 設計・構築
- ディレクトリ設計・構築
- データベース構築・支援

情報セキュリティソリューション

- 情報セキュリティ環境設計・構築
- 統合ログ管理システム設計・構築
- MDM（モバイルデバイス管理）

グリーンITソリューション（TCO削減）

- エンドポイント統合管理設計・構築

仮想化ソリューション

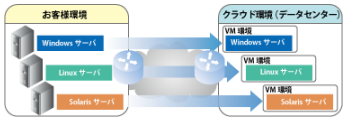
- 仮想サーバー設計・構築
- 仮想デスクトップ
- ネットワーク仮想化設計・構築

ネットワークソリューション

- モバイルアクセスソリューション
- ネットワーク仮想化設計・構築
- 無線LAN設計・構築
- ライフタイム保証

サポート＆サービス

- ワンストップサポートサービス
- DBパフォーマンス診断
- データ消去サービス
- レンタル仮想サーバサービス



Linux Professional Institute

教育事業サービスラインアップ

▼詳しいソリューションのご紹介
http://panasonic.co.jp/pstc/products/

Panasonic Solution Technologies

学習管理システムASPサービス

LMS（eラーニング／eテスト／eアンケート）

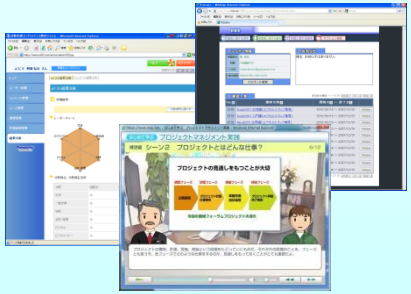
- PaLearn（ASP）
- G-PaLearn（ASP）
- カスタマイズ構築

研修支援

- 研修支援システム（ASP）
- セミナー動画配信システム（ASP）
- カスタマイズ構築

業務システムクラウド開発

アプリケーション開発例
- 学習塾VOD/学習管理
- コールセンタ派遣社員評価管理
- 情報セキュリティ自主点検管理
- 匿名アンケート管理



教材作成サービス

簡易版

- PowerPoint教材をFlash教材へ変換するコンテンツ制作

テンプレート活用版

- オリジナル教材テンプレートを活用したコンテンツ制作

動画版

- 動画収録
- 動画編集、制作

イラスト作成

- 手書きイラスト
- お絵かきソフトによるイラスト

教育研修サービス（eラーニング／集合研修）

人間力育成コース

- 交渉学シリーズ
  - 交渉学エントリー（eL/集合）
  - 交渉学ベーシック（集合）
  - 交渉学アドバンス（集合）
  - 技術者向け交渉学（集合）
  - 営業社員向け交渉学（集合）
- コーチング（集合）
- メンタルヘルス（セルフケア）（eL）

仕事のプロセス・マネジメントコース

- 仕事の基本
  - QCの基本と応用（eL/集合）
  - 間接部門のQC活動（eL）
- ITの基本
  - Office2003（eL/集合）
  - Office2007（eL/集合）
  - Office2010（eL/集合）
  - Officeバージョンアップ差分（集合）
- プロジェクトマネジメント
  - PM実践（eL/集合）
  - プロジェクトのリスクマネジメント（eL）
  - PMC試験対策（eL/集合）
  - PMC試験予想問題集（eL）
  - PMS・P試験対策（eL）
  - PMS・P試験予想問題集（eL）
- ビジネス戦略
  - マーケティング戦略（eL）
  - 競争戦略（eL）
  - プログラムによる戦略の実践（eL）
  - 戦略実践の基礎（集合）

知財・コンプライアンスコース

- コンプライアンスシリーズ
  - 下請法（基礎編）（eL）
  - 下請法（実践編）（eL）
  - 独禁法（基礎編）（eL）
  - 独禁法（カルテル編）（eL）
  - 契約（基礎編）（eL）
  - 著作権（基礎編）（eL）
  - 管理者向け情報セキュリティ（集合）
  - 情報セキュリティ個人情報保護（eL）
- 知財マネジメントシリーズ
  - 知財マネジメント（特許、実用新案、意匠・商標、著作権）（eL）

その他

- 社内講師育成（集合）
- キャリア・デザイン
- Webシステム構築
- Linuxサーバ構築
- 電気・通信技術
  - 高周波回路（eL）
  - 半導体・IC回路基礎（eL）
  - デジタル信号処理基礎（eL）
  - アナログ信号処理基礎（eL）
  - 統計・設計品質基礎（eL）
  - 機構設計基礎（eL）
  - 品質管理基礎（eL）
  - 電気・電子回路基礎（eL）

Linux Professional Institute

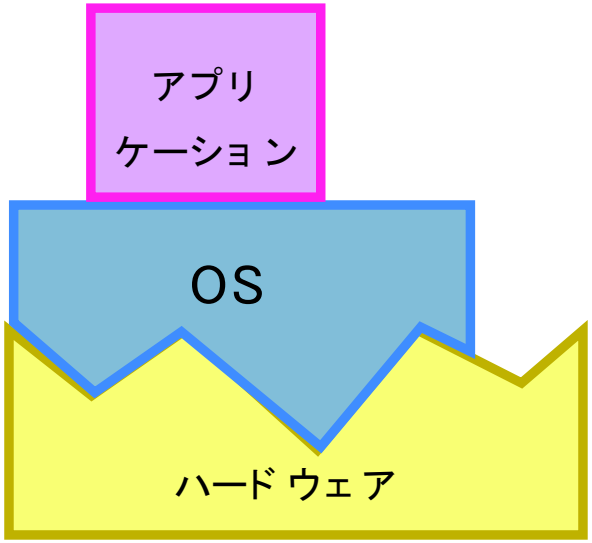
LPICとは

「Linux Professional Institute Certification」の略称で、
特定非営利活動法人/Linux技術者認定機関「LPI」
の実施する、

Linux技術者認定試験

OS（オペレーティング・システム）の役割

- ハードウエアとソフトウエア（アプリケーション）の間を取り持つ
- ハードウエアの違いはOSが吸収
- CPU、メモリ、ディスク、ネットワークなどの管理
- マルチタスク機能
- 他には・・・
  - ユーザー認証・セキュリティ
  - 使い勝手のよい環境（アイコン、メニューなど）
  - プログラム開発環境など



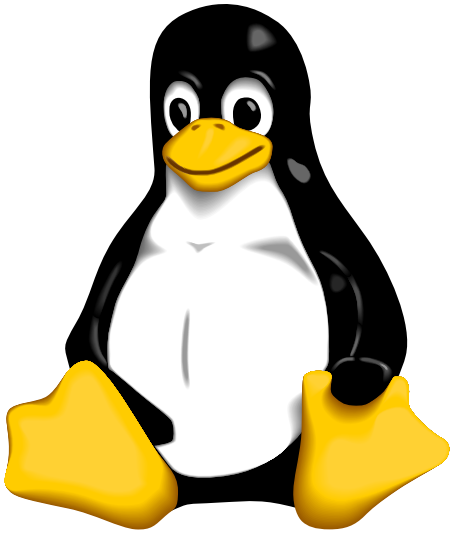
Linux とは

- 1991年 フィンランド人の Linus Torvalds (リーナス・トーバルズ)氏(当時は大学生)が作成
- “UNIXと同じようなOSを、自分のパソコンで動かしたい”のがきっかけ
- ソースコードをインターネットで公開
  - 世界中の研究者、開発者の間で話題に

- 現在では、サーバー系、制御系(組込み機器)を中心に業務で使われるようになってきた

Linuxの特徴

- オープン＆フリー
  - プログラムのソースコードが公開されている
  - 多くの人が協力してプログラムを改良
  - 原則として無償で入手できる(GPL)



- 安定性&互換性
  - 障害が発生しにくい構造
  - 「24時間、365日稼働」にも耐えられる
  - ソースコードが公開されているので、不具合を修正できる
  - UNIXで稼働しているアプリケーションを移植できる

LPICの特徴

■世界標準

LPICは世界共通の国際認定制度です。

■中立

LPIC はLinuxの技術力を中立公正に判定する試験です。

■世界最大

LPICは、世界最大のLinux技術者受験者数を有します。

試験概要

試験概要

- 場所　全国各地から選択できます
- 日時　選択できます
- 時間　90分
- 費用　1試験あたり15,000円（消費税別）
- ピアソンVUE http://www.vue.com/japan/index.html

- 形態

  コンピューターを使い、オンラインで約60問の問題を配付します。
  マウスによる選択方式がほとんどですが、キーボード入力問題も
  多少出題されます。

- 合格発表　　試験終了と同時に判明します。

Version 4.0

LPIC-1 Ver4.0 2015年6月1日よりリリース

■主な変更点

メジャーなディストリビューションで使われている「systemd」の採用など、新しく使われるようになっているツール類を取り入れ、各出題範囲の重要度を見直された。

- systemd (ブートプロセス、ブートターゲット、ジャーナルサブシステムを管理)
- ログインマネージャ (LightDM)
- 新しいコマンド

  (systemctl,wall,grub-mkconfig,xz,pgrep,pkill,screen,gdisk,gparted,getent,Source,ntpq,
  logrotate,ip,netcat,ping6,traceroute6,tracepath6,fuser,who,w,last)
- 新しい技術の追加(認知レベル)

  (Btrfs,rsyslog, syslog-ng)

試験範囲

■http://www.lpi.or.jp/lpic1/range/

101試験

主題101：システムアーキテクチャ

主題102：Linuxのインストールと
パッケージ管理
主題103：GNUとUnixのコマンド

主題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

102試験

主題105：シェル、スクリプト、
およびデータ管理
主題106：ユーザインターフェイスと
デスクトップ
主題107：管理業務
主題108：重要なシステムサービス
主題109：ネットワークの基礎

主題110：セキュリティ

ポイント解説

本日解説するポイント 101試験

- コマンドラインの操作
- 基本的なファイル管理の実行
- ストリーム、パイプ、リダイレクトの使用
- プロセスを生成、監視、終了

本日解説するポイント 102試験

- 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の管理
- 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
- ネットワーク関連

103.1 コマンドラインの操作

■ **重要度**: 4

■ **説明**　コマンドラインを使用して、シェルおよびコマンドと対話できる。
この目標は、bashシェルを使用することを想定している。

■ **主要な知識範囲**

- 1つのシェルコマンドおよび1行のコマンドシーケンスを使用して、コマンドラインでの基本的な作業を行う
- 定義することを含めたシェル変数の使用と変更、環境変数の参照とエクスポート
- コマンド履歴の使用と編集
- 定義済みパス内に存在するコマンドおよび存在しないコマンドの呼び出し

■ **重要なファイル、用語、ユーティリティ**

- ✓ bash
- ✓ echo
- ✓ env
- ✓ exec
- ✓ export
- ✓ pwd
- ✓ set
- ✓ unset
- ✓ man
- ✓ uname
- ✓ history
- ✓ .bash_history

シェル変数・環境変数

- set
  - シェル変数・環境変数の両方表示
- env
  - 環境変数のみ表示
- unset
  - 変数の削除
- export
  - 環境変数にする

コマンド履歴

- **history　　履歴一覧**
- **history -c　　履歴の消去**

↑	1つ前に入力したコマンドを表示（続けて押せば遡っていく）
↓	1つ後に入力したコマンドを表示
!n	historyの表示結果でn番のコマンドを実行
!-n	n個前に入力したコマンドを実行
!!	直前に入力したコマンドを再実行
!文字列	指定した文字列で始まるコマンドで、直近に入力したものを実行
!?文字列?	指定した文字列が含まれるコマンドで、直近に入力したものを実行
^文字列1^文字列2^	直前に実行したコマンドの文字列1を文字列2に変えて実行
:p	:pをつけると、コマンドを実行せずに表示のみ行う

環境変数 PATH

- コマンド(プログラム)のある場所を探すPATH(経路)...サーチパス。
- コマンドを実行するには、
  - PATH上のディレクトリに、配置する。
  - 絶対もしくは相対パスで、直接コマンドのありかを指定する。

103.3 基本的なファイル管理の実行

■**重要度**: 4

■**説明**　ファイルおよびディレクトリを管理するための基本的なLinuxコマンドを使用できる。

■**主要な知識範囲**

- 個々のファイルおよびディレクトリをコピー、移動、削除する
- 複数のファイルおよびディレクトリを再帰的にコピーする
- ファイルおよびディレクトリを再帰的に削除する
- 基本的なものから高度なものまで、ワイルドカード規則をコマンドで使用する
- findを使用して、種類、サイズ、または時刻を基にファイルを見つけて操作する
- tar、cpioおよびddの使用方法

重要なファイル、用語、ユーティリティ

- ✓ cp
- ✓ find
- ✓ mkdir
- ✓ mv
- ✓ ls
- ✓ rm
- ✓ rmdir
- ✓ touch
- ✓ tar
- ✓ cpio
- ✓ dd
- ✓ file
- ✓ gzip
- ✓ gunzip
- ✓ bzip2
- ✓ xz
- ✓ ファイル名のパターンマッチ

ファイル操作

- cp　　コピー
- mv　　移動(名前の変更も可能)
- rm　　消去

ディレクトリ操作

- mkdir　ディレクトリ作成
- mv　　ディレクトリ名変更
- rmdir　ディレクトリ削除
  （中身が空であること）

再帰的

- ディレクトリツリーをたどって全てのファイルやディレクトリをなめていく…と云うこと。

- rm -r ディレクトリ名
  で、ディレクトリ内部をすべて削除できる!

ワイルドカード規則

- (ブラケット外部の) '? ' はあらゆる単一の文字にマッチする。
- (ブラケット外部の) '*' はあらゆる文字列にマッチする。空文字列 (empty string) にもマッチする。
- **文字クラス (character class)**
  - "*[...]*" と言う表記は、先頭の '[' に続く最初の文字が '!' で なければ、ブラケットの中に含まれている文字のどれか一つにマッチする。ブラケットの内部に含まれる文字列は空であってはならない。したがって ']' も最初の文字に指定すればブラケットの内部に含めることができる (つまり "*[][!]*" は '[', ']', '!' の 3 文字のどれかにマッチする)。
- **領域指定 (range)**
  - 特殊な表記法が一つ存在する。'-' を挟む二つの文字は領域指定となる。(つまり "*[A-Fa-f0-9]*" は "*[ABCDEFabcdef0123456789]*" と等価となる。)
    '-' 文字そのものを入れたい場合は、ブラケットの先頭または最後の文字に指定する。(つまり "*[]-]*" は二つの文字 ']' と '-' にマッチし、"*[--0]*" は '-', '.', '0' の 3 文字にマッチする。この間の '/' にはマッチしない。)
- **補集合 (complementation)**
  - "*[!...]*" と言う表記は、ブラケットの内部に含まれない単一の文字にマッチする (ただし先頭にある '!' は除外)。
  - (つまり "*[!]a-]*" は ']', 'a', '-' 以外のすべての文字の、どれか一つマッチする。)
  - バックスラッシュ '＼' を前置すれば、'?', '*', '[' は通常の文字として扱われる。またはシェルのコマンドラインの一部に指定する場合は、クォートで囲っても同じ効果が得られる。ブラケットの内部では、これらの文字はその文字自身だけを意味する。すなわち "*[[?*＼]*" は、'[', '?', '*', '＼' のどれか一文字にマッチする。

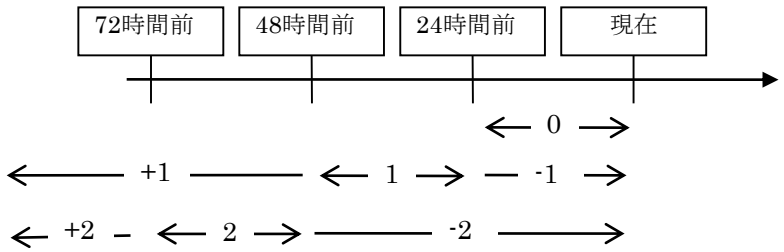
ASCII コード表

\	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
00	NUL	SOH	STX	ETX	EOT	ENQ	ACK	BEL	BS	HT	LF	VT	FF	CR	SO	SI
10	DLE	DC1	DC2	DC3	DC4	NAK	SYN	ETB	CAN	EM	SUB	ESC	FS	GS	RS	US
20	SP	!	”	#	$	%	&	’	(	)	*	+	,	-	.	/
30	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
40	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
50	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
60	‘	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
70	p	q	r	s	t	u	v	w	x	y	z	{	|	}	~	DEL

find 1/2

- find　基点ディレクトリ　検索条件　[アクション]
- [検索条件]
  - –name　ファイル名
    - 指定したファイルを検索。ファイル名にワイルドカードを使う場合は、ファイル名全体を””で囲む。
  - find . –name “sample*”
  - –mtime　n
    - (n+1)×24時間前　から　n×24時間前　の間に変更(書き込み)があったファイル。
    - + をつけるとそれよりも過去、
    - – をつけるとそれより直近になります。

find . –mtime -1 または find . –mtime 0
　カレント・ディレクトリ以下で、
　24時間以内に変更があったファイルを検索
find . –mtime +7
　1週間以上(=168時間以内に)
　変更がなかったファイルを検索



find 2/2

find 基点ディレクトリ 検索条件 [アクション]

- -size サイズ
- 単位を省略するとブロック単位(1ブロック=512バイト)、cはバイト単位、kはキロバイト単位。+をつけるとより大きい、-をつけるとより小さいの意味。
- find . -size +200
  カレント・ディレクトリ以下で、200ブロックより大きいサイズのファイルを検索
- find . -size -200c
  カレント・ディレクトリ以下で、200バイトより小さいサイズのファイルを検索

[アクション]

- -print 検索結果をディスプレイに表示(デフォルト)
- -ls 見つかったファイルの詳細情報(ls -lと同じ内容)を表示
- -exec コマンド {} \; 見つかったファイルに対してコマンドを実行(即座)
- -ok コマンド {} \; 見つかったファイルに対してコマンドを実行(確認後)

tar

tarコマンド（Tape Archive）

- 複数のファイルを1つにまとめる(アーカイブする)
- tar cvf ファイル名.tar ファイル1 ファイル2 (ディレクトリ) …
  - 指定したファイルやディレクトリをtarファイルにまとめる
- tar xvf ファイル名.tar
  - tarファイルからファイルを抽出する
- tar tvf ファイル名.tar
  - tarファイル内のファイルのリストを表示

tar zcvf：アーカイブして圧縮
tar zxvf：解凍して抽出

gzipコマンド

- ファイル名の末尾に自動的に.gzがつく
- gzip ファイル名　　圧縮
- gunzip ファイル名　　解凍　（gzip -d ファイル名　でも同じ）

bzip2コマンド, xzコマンド

- 高圧縮(低速)コマンド。サイズが大きく、解凍頻度が高いものは有利。

cpioおよびdd

cpioコマンド （Copy I/O）

- 複数のファイルを1つにまとめる

- ls | cpio -ov >ファイル名.cpio
  - カレントディレクトリにあるファイルをcpio形式でアーカイブする
- cpio -iv < ファイル名.cpio
  - ファイルを抽出する

ddコマンド

- 1つのファイルを別ファイル（デバイス）に送る

- dd if=入力ファイル of=出力ファイル bs=ブロックサイズ count=回数
  - 例 dd if=boot.img of=/dev/fd0
  - dd if=/dev/zero of=nulldata bs=512 count=1

103.4 ストリーム、パイプ、リダイレクトの使用

■**重要度**: 4

■**説明**　テキストデータを効果的に処理するためにストリームのリダイレクトや接続ができる。
この作業には標準入力、標準出力、標準エラー出力へのリダイレクト、あるコマンドの出力を別のコマンドの入力にパイプする、あるコマンドの出力を別のコマンドの引数として使用する、出力を標準出力とファイルの両方に送るといったことが含まれる。

■**主要な知識範囲**

- 標準入力、標準出力、標準エラー出力をリダイレクトする
- あるコマンドの出力を別のコマンドの入力にパイプする
- あるコマンドの出力を別のコマンドの引数として使用する
- 出力を標準出力とファイルの両方に送る

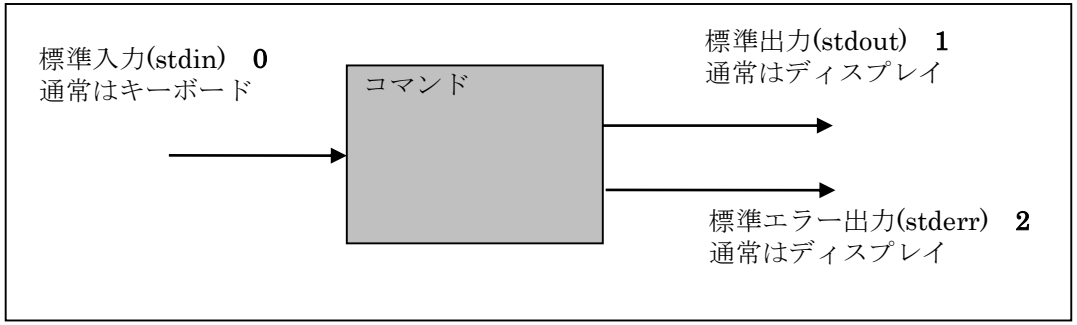
■**重要なファイル、用語、ユーティリティ**

✓ tee

✓ xargs

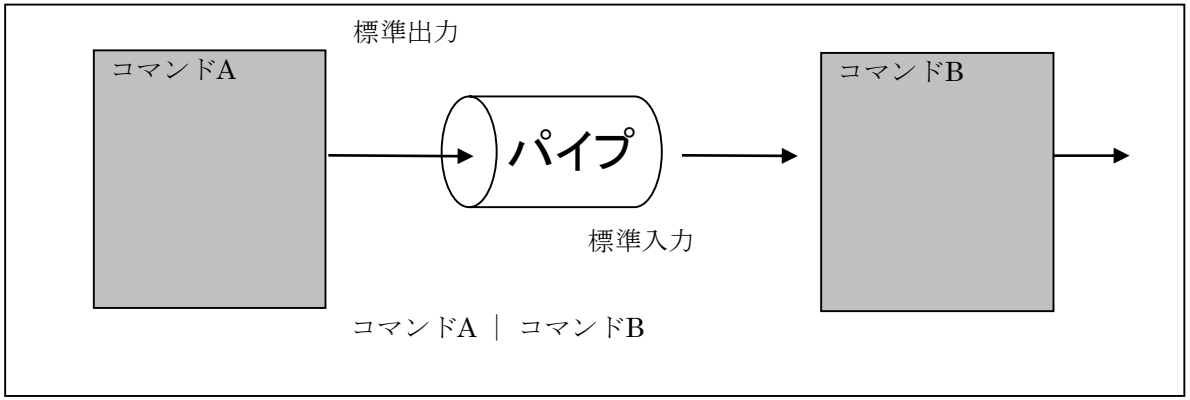
標準入力、標準出力、標準エラー出力

- コマンドを実行する場合、通常は標準入力（キーボード）からデータを読み、処理結果を標準出力（ディスプレイ）に出力します。
- catコマンドで引数にファイル名を指定しなかった場合は、キーボードから読み込んだデータをディスプレイに表示する処理を行います。
- 標準入力、標準出力などを切り替えることを、**リダイレクト**と呼びます。
- 標準入力の切り替え　<
- 標準出力の切り替え　>（上書き）　>>　（追記）



パイプ

■コマンドAの標準出力を
コマンドBの標準入力として使うことができる。



sort と uniq

- ファイルから重複している行を見つける。
  - cat data.txt | sort | uniq -d
- 人気ログインシェルランキング?
  - cat /etc/passwd | cut -d : -f 7 | sort | uniq -c

Tee

(コマンド) | tee ファイル名

- 前のコマンドの実行結果(標準出力)を受け取り、
  それをディスプレイに表示し、かつ指定したファイルに書き込むもの。
  水道のT字管のイメージ。

xargs

- % find . -name “*.c” -exec grep hogehoge {}　;
- % find . -name “*.c” | xargs grep hogehoge\

103.5 プロセスの生成、監視、終了

- **重要度：** 4
- **説明**　基本的なプロセス管理を行える。
- **主要な知識範囲**
  - ジョブをフォアグラウンドやバックグラウンドで実行する
  - ログアウト後にも実行が継続されるようにプログラムにシグナルを送信する
  - 活動中のプロセスを監視する
  - プロセス群を選択し、並び替えて表示する
  - プロセスにシグナルを送信する

- **重要なファイル、用語、ユーティリティ**

✓ &
✓ bg
✓ fg
✓ jobs
✓ kill
✓ nohup
✓ ps
✓ top
✓ free
✓ uptime
✓ pgrep
✓ pkill
✓ killall
✓ screen

ジョブ

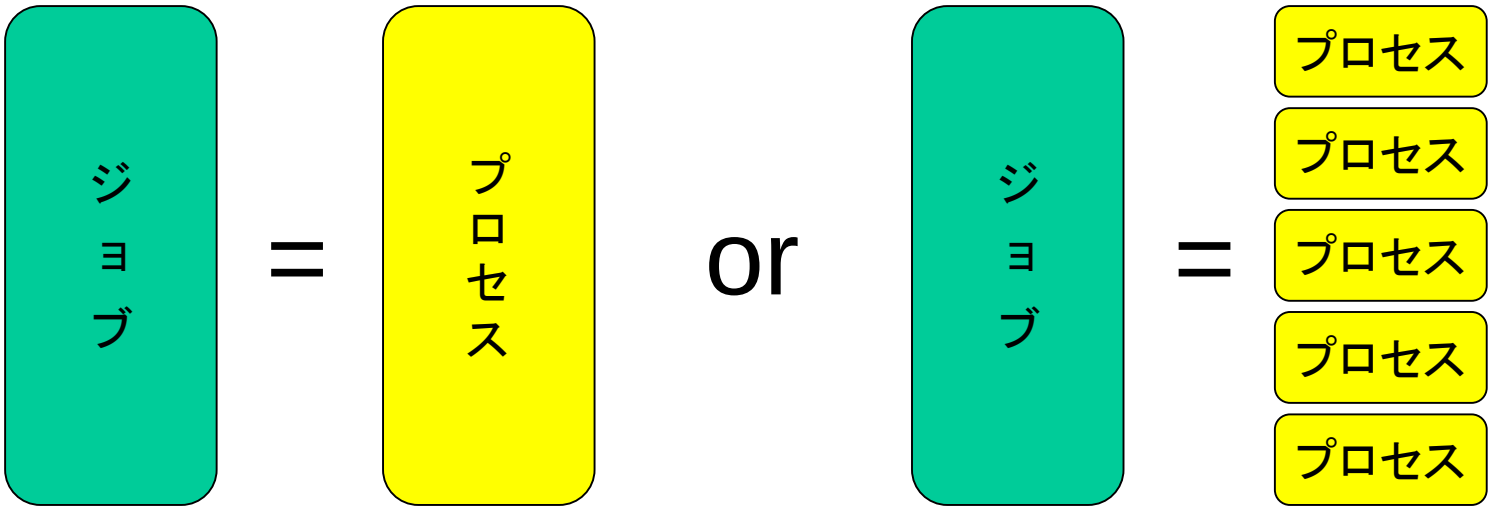
- シェル上でコマンドを実行すると、「ジョブ」という単位で処理される。
  - **フォアグラウンドジョブ**: キーボードからの入力を受け取れる状態のジョブ
  - **バックグラウンドジョブ**: 陰で動いているジョブ
- バックグラウンドジョブとして起動するには、コマンド名の後ろに「**&**」をつける。

ジョブの管理

■コマンド名 &	バックグラウンドでジョブを起動
■jobs	現在稼動中のジョブ一覧を表示 （ジョブ番号、ステータス、ジョブ名）の順
■Ctrl+Z	フォアグラウンドジョブをバックグラウンドに移す ジョブは一時停止する
■bg [%ジョブ番号]	バックグラウンドで停止しているジョブを再開
■fg [%ジョブ番号]	バックグラウンドジョブをフォアグラウンドへ移動

プロセス

■Linux上での、処理の単位。



(ジョブ ≧ プロセス)

■プロセスはプロセスID(pid)で管理される。
(システム内で重複しないように番号がつけられる)

■同じコマンドであっても、実行するたびにプロセスIDは変わる。

プロセスの管理

- **ps** 現在の端末で自分が実行したプロセスを表示
- **ps a** ほかのユーザーのプロセスも表示
- **ps x** 端末を持たないプロセスも表示
- **ps ax** システムで稼動中の全プロセスを表示
- **ps l** 詳細情報を表示（親プロセスIDなど）
- **ps u** 詳細情報を表示（ユーザー名など）
- **ps axlw** 折り返して全体を表示
  （通常は1行80文字で切られる）
- **ps axlf** ツリー状に表示 (≒pstreeコマンド)

- **pgrep (-f) ‘プロセス名’** 指定したプロセス名のプロセスを表示
  ※ -f をつけないと、プロセス名(先頭15文字)しか、マッチしない
- **pgrep -u root sshd** sshd という名前で、かつ root が所有するプロセスのみを表示

プロセスにシグナルを送る

シグナル名	シグナル番号	意味
HUP	1	ハングアップ。デーモンプロセスの初期化
INT	2	割込み。処理の中断 （= Ctrl+C）
KILL	9	強制終了
TERM	15	終了（デフォルト）
TSTP	20	一時停止 （= Ctrl+Z）
CONT	18	再開

- kill -kill 123　　プロセスID123 のプロセスを強制終了
- kill -1 123　　プロセス ID123 のプロセスを再起動
- pkill -f “httpd”　　httpd プロセスを全終了(TERM)

■重要度: 5

■説明 ユーザアカウントの追加、削除、一時停止、変更ができる。

■主要な知識範囲

- ユーザおよびグループを追加、変更、削除する
- パスワード/グループデータベースにある
  ユーザ/グループ情報を管理する
- 特別な目的を持つ制限付きのアカウントの作成、管理する

■重要なファイル、用語、ユーティリティ

- ✓ /etc/passwd
- ✓ /etc/shadow
- ✓ /etc/group
- ✓ /etc/skel
- ✓ chage
- ✓ groupadd
- ✓ groupdel
- ✓ groupmod
- ✓ passwd
- ✓ useradd
- ✓ userdel
- ✓ usermod

ユーザ管理

- useradd ユーザ名　　ユーザ追加
- userdel -r ユーザ名　　ユーザ削除
- usermod -L ユーザ名　　ユーザー時停止
  (解除は、-U)

- /etc/passwd
- /etc/shadow

} に保存。

/etc/skel

- 新規ユーザのホームディレクトリの
  テンプレート(ドットファイル等を仕込んでおく)

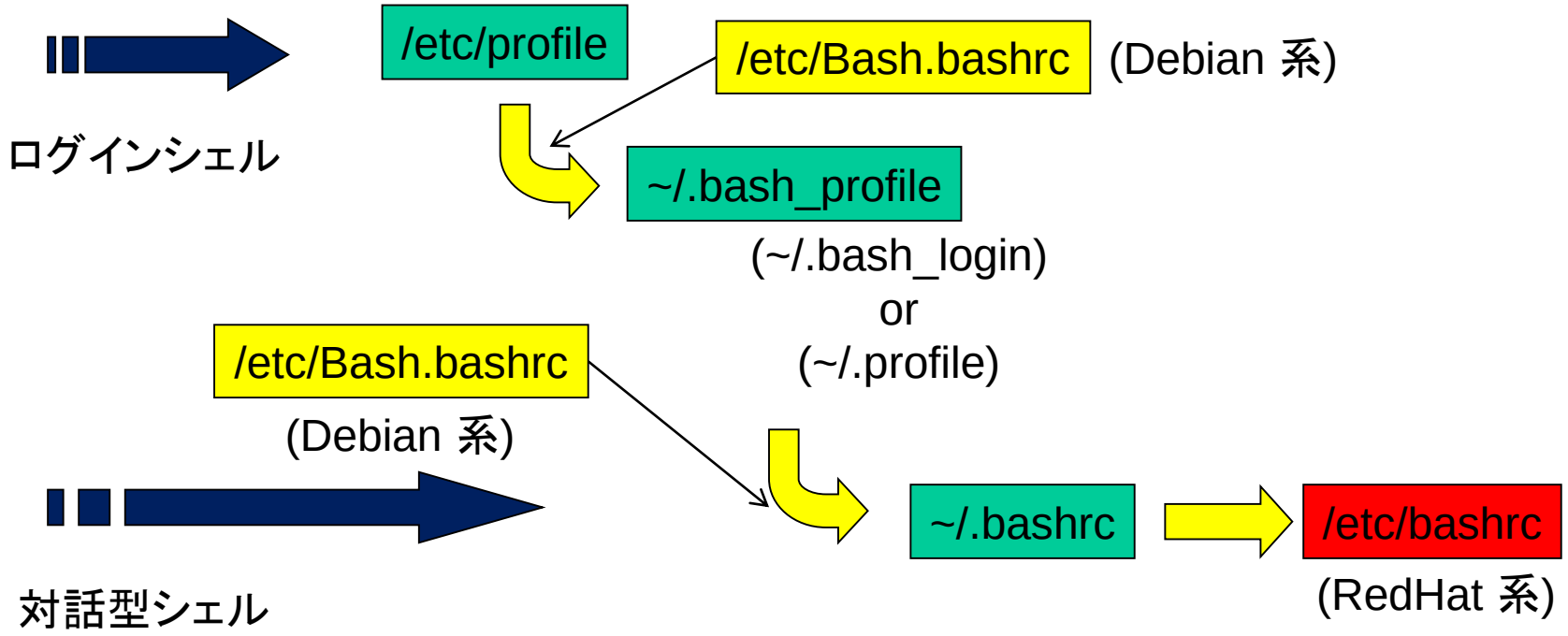
- **重要度**: 4
- **説明** ユーザの要求に応じてシェル環境をカスタマイズできる。
  全体のプロファイルおよびユーザのプロファイルを
  変更する。
- **主要な知識範囲**
  - ログイン時または新しいシェルを生成したときに、
    環境変数(PATHなど)を設定する
  - よく使用する一連のコマンド用に Bash の関数を作成する
  - 新しいユーザアカウント用のスケルトンディレクトリを保守する
  - コマンドサーチパスを適切なディレクトリに設定する

重要なファイル、用語、ユーティリティ

- ✓ .
- ✓ source
- ✓ /etc/bash.bashrc
- ✓ /etc/profile
- ✓ env
- ✓ export
- ✓ set
- ✓ unset
- ✓ ~/.bash_profile
- ✓ ~/.bash_login
- ✓ ~/.profile
- ✓ ~/.bashrc
- ✓ ~/.bash_logout
- ✓ function
- ✓ alias
- ✓ lists

bashの設定ファイル

	ログイン時一度だけ (環境変数の設定等)	bashを起動するたび (alias・関数の設定等)
全ユーザ	/etc/profile	/etc/bashrc
個別ユーザ	~/.bash_profile	~/.bashrc



bashの関数

書式

- function 関数名() { コマンド; }
  alias よりも複雑な”内部コマンド”を定義可
- alias のまね…
  function ls() { command ls -CF --color=tty $@ ; }

カスタマイズ?

- 環境変数PATHに追加
  - PATH=$PATH:/home/...
- alias の設定
  - 設定
    - alias ls='ls -l'
  - 解除
    - unalias ls
  - 一時解除
    - ＼ls

■**重要度：** 4

■**説明**　既存のスクリプトをカスタマイズしたり、簡単なBashスクリプトを新規作成できる。

■**主要な知識範囲**

- 標準的なshの構文（ループ、テスト）を使用する
- コマンド置換を使用する
- コマンドによって返される、成功または失敗を示す戻り値やその他の情報をテストする
- 条件に応じて、スーパーユーザにメールを送信する
- 先頭行(#!)を利用して、適切なスクリプトインタープリタを選択する
- スクリプトの位置、所有権、実行権、SUID権を管理する

■**重要なファイル、用語、ユーティリティ**

- ✓ for
- ✓ while
- ✓ test
- ✓ if
- ✓ read
- ✓ seq
- ✓ exec

制御構造

- if *条件文*
  then
  実行文
  elif *条件文*
  実行文
  else
  実行文
  fi

- for *識別子* in *リスト*
  do
  $*識別子を使う文*
  done

- while *条件文*
  do
  実行文
  done

条件文

- [$? -eq 0] 手前のコマンドが成功なら

- s1 = s2　　文字列 s1 と s2 が同じであれば真
- s1 != s2　　文字列 s1 と s2 が同一でなければ真
- s1 < s2　　文字列 s1 が文字列 s2 に対し、ASCII 順で前なら真
- s1 > s2　　文字列 s1 が文字列 s2 に対し、ASCII 順で後なら真

- n1 -eq n2　　整数 n1 と n2 が等しければ真
- n1 -ne n2　　整数 n1 と n2 が等しくなければ真
- n1 -gt n2　　整数 n1 が n2 がより大きければ真
- n1 -ge n2　　整数 n1 が n2 より大きいか等しければ真
- n1 -lt n2　　整数 n1 が n2 より小さければ真
- n1 -le n2　　整数 n1 が n2 より小さいか等しければ真

シェルスクリプト

- コマンド置換
  - 「‘」(バッククォート: jisキーボードでは、Shift + @) で、もしは、$() で、コマンドを囲むと、実行結果と置き換わる。
    - filesize=`cat $1 | wc -c`
    - filesize=$(cat $1 | wc -c)
- 直前コマンドの結果
  - $?
    - 正常終了した場合の戻り値 : 0
    - 異常終了した場合の戻り値 : 0以外
- シバン
  - #!
    - #!/bin/bash 等、以下のスクリプトを処理するインタープリタを指定。

ネットワーク関連 1/2

109.1 インターネットプロトコルの基礎

- ネットワークマスクとCIDR表記法について理解していることを示す
- プライベートとパブリックのドット区切り形式のIPアドレスの違いを知っている
- 一般的なTCPおよびUDPのポート（20、21、22、23、25、53、80、110、123、139、143、161、162、389、443、465、514、636、993、995）について知っている
- UDP、TCP、およびICMPの違いや主な機能について知っている
- IPv4とIPV6の主な違いについて知っている
- IPV6の基本的な機能について知っている

ネットワーク関連 2/2

109.2 基本的なネットワーク構成

- ネットワークインターフェイスの設定を手作業および自動で行う
- ホストの基本的なTCP/IP設定
- デフォルトルートを設定する

109.3 基本的なネットワークの問題解決

- ネットワークインターフェイスおよびルーティングテーブルを手作業および自動的に設定できる。これには、ネットワークインターフェイスの追加、起動、停止、再起動、削除、および再設定が含まれる
- ルーティングテーブルを変更、参照、設定し、不適切なデフォルトルート設定を手作業で訂正する
- ネットワーク構成に関連する問題をデバッグする

ホストの基本的なTCP/IPを構成(1/2)

- ホスト名
  - /etc/HOSTNAME ないし /etc/hostname
  - (RedHat系は、/etc/sysconfig/network)
- 名前解決
  - 問い合わせ順
    - /etc/nsswitch.conf もしくは、/etc/host.conf
  - ファイルによる指定
    - /etc/hosts
  - DNS 参照先の設定
    - /etc/resolv.conf

ホストの基本的なTCP/IPを構成(2/2)

ネットワークインターフェース

- 設定ファイル
  - /etc/sysconfig/network-scripts (RedHat 系)
  - /etc/network/interfaces (Debian)

DHCP設定は、上記ファイルに
BOOTPROTO=dhcp(RedHat系)
iface eth0 inet dhcp（Debian) 等と、書く

- 確認
  - Ifconfig もしくは、ip a
- 一時設定
  - ifconfig eth0 192.168.0.20 netmask 255.255.255.0
  - (nmcli connection modify eth0 ipv4.addresses ″192.168.0.20/24 “)

※ CentOS7 の設定。LPIC 102(Ver.4.0) の、「重要なユーティリティ」ではない。

ルーティングテーブル

- route もしくは、ip r (ip route)
  - ルーティングテーブルの表示
- route add もしくは、ip route add
  - ルーティングの追加
    - route add -net 192.168.10.0 netmask 255.255.255.0 gw 192.168.2.254
    - route add default gw 192.168.2.1
    - ip route add 192.168.10.0/24 via 192.168.2.254 dev eth1
- route del もしくは、ip route del
    - route del -net 192.168.10.0 netmask 255.255.255.0 gw 192.168.2.254
    - ip route del 192.168.10.0/24

診断コマンド

- ping
  - 疎通確認(トラブルの切り分け)
- traceroute
  - 経路確認
- netstat
  - ネットワーク情報
    (DNSトラブルのときは、-n で名前解決を抑止)

その他実際のトラブルシューテイング

- iptables によるパケットフィルタ
  - iptables -L -n などで確認
- 各サーバの設定ファイル、ログ等の確認
- inetd 使用の場合
  - /etc/hosts.allow
  - /etc/hosts.deny 設定の確認
- netstat でソケットの使用状況を確認
  - netstat -at
- tcpdump で、通信をモニタ
  - tcpdump -X -i eth0 port 80