

LPI-JAPAN

LPIC 1



LPI-Japan アカデミック認定校
NEC ネットズエスアイ 株式会社
濱田 美奈子



Linux

LPIC

LPIC

1

LPI101

Linux

アジェンダ

◆ LPIC(Linux)

◆ Linux

◆ 101



会社紹介

NEC
NECネットエスアイ

▶ 社名 **NECネットエスアイ株式会社**

<http://www.nesic.co.jp/pr/index.html>



NECネットエスアイは、人・企業・社会をスムーズにつなぐためのネットワークシステムの構築と、その基盤となる通信インフラの工事、そして、システムをより安心・安全にご利用いただくためのお客様サポートを展開しています。

LPIC(Linux技術者認定試験) の概要



「LPIC(エルピック)」は、NPO法人/Linux技術者認定機関「LPI(エルピーアイ)」(本部:カナダ)が実施している世界共通・世界最大・最高品質の「Linux技術者認定制度」です

LPICの特長

1. 世界で通用する資格



LPICは世界で通用するIT資格です。公正なLinuxスキルの判断基準として国際的に認められています。世界共通基準で認定を行っており、150カ国以上の方々に受験されています

2. Linuxの技術力を客観的に評価する資格



LPICはLinuxの技術力を中立公正に判定するIT資格です。製品メーカーや配布元企業に全く依存せず、常に中立公正な立場で、Linux技術力を評価しています。

3. 受験者数でも世界最大規模の資格



世界で43万人以上が受験し、15万人以上の認定者がいる、Linux資格試験の中でも世界最大規模の資格です。日本国内での累計受験者数も25万人を突破しています。(2015年5月時点)



LPIC 3

初級

上級

LPICレベル1 LPIC-1

サーバの
構築、運用・保守

実務に必要なLinuxの
基本操作とシステム管理
が行えるエンジニアで
あることを証明できます。

101試験

102試験

LPICレベル2 LPIC-2

ネットワークを含む、
コンピュータシステムの
構築、運用・保守

Linuxのシステムデザイン、
ネットワーク構築において、
企画、導入、維持、トラブル
シューティング、キャパシ
ティプランニングができる
エンジニアであることを
証明できます。

201試験

202試験

LPICレベル3 LPIC-3

各分野の最高レベルの
技術力を持つ
専門家レベル

LPIC-3 Specialty Mixed
Environment
(300試験)

LPIC-3 Specialty Security
(303試験)

LPIC-3 Specialty Virtualization &
High Availability (304試験)

300試験

303試験

304試験



2015年6月1日より、日本語版のレベル1の出題範囲を改訂します。
改訂後の出題範囲のバージョンは、Ver4.0となります。

■Ver4.0のリリースによる受験方式の変更

新バージョン（Ver4.0）がリリースされてから「6ヶ月間」は、旧バージョン（Ver3.5）も受験することができます。

よって、受験者は、新旧いずれかのバージョンの試験を選択して受験することができます。
またバージョンに関係なく、101試験と102試験に合格すれば、LPICレベル1に認定されます。
例）101試験は旧バージョン（Ver3.5）で合格し、102試験は新バージョン（Ver4.0）で合格した場合など

受験の際は、LPI-ID取得後、試験配信会社（テストセンター）「ピアソンVUE」で受験予約を行って下さい。

試験	試験番号	受験期間
101試験（Ver3.5）	101-350	2012/10/1～2015/11/30
101試験（Ver4.0）	101-400	2015/6/1より【新Version】
102試験（Ver3.5）	102-350	2012/10/1～2015/11/30
102試験（Ver4.0）	102-400	2015/6/1より【新Version】



■Ver3.5 と Ver4.0 の差分詳細

http://www.lpi.or.jp/doc/20150317-101_102.pdf

■出題範囲変更の概要

(1) systemd 【LPI 101】

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する systemd の使用が追加されました。

(2) ログインマネージャ 【LPI 102】

xdm、kdm および gdm は認知レベルに引き下げ、ログインマネージャとして LightDM マネージャが追加されました。

(3) 新しいコマンド

次のコマンドが「重要なファイル、用語、ユーティリティ」に追加されました。

systemctl , wall , grub-mkconfig , xz
pgrep , pkill , screen
gdisk , parted , getent , source , ntpq , logrotate
ip , netcat , ping6 , traceroute6 , tracepath6
fuser , who , w , last

(4) 新しい技術の追加（認知レベル）

次の技術が認知レベルで追加されました。 Btrfs , rsyslog , syslog-ng

※「認知レベル」での出題とは「知っている」を問うことを示し、「操作できる」までは問わないことを示します。



101

- 主題101:システムアーキテクチャ

この範囲の一部を本日解説

- 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

101試験範囲詳細(Ver3.5):

<http://www.lpi.or.jp/lpic1/range/ver3.5/range101.shtml>

101試験範囲詳細(Ver4.0):

<http://www.lpi.or.jp/lpic1/range/ver4.0/range101.shtml>

101試験：出題範囲詳細(一例)

103.1 コマンドラインで操作する	
	4
	bash
	• 1 1 • • •
	. bash echo env exec export pwd set unset man uname history

Linuxの構成・基本操作



Linux

OS

- 1991

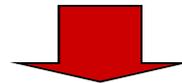
Linux

-

- OS ()

-

CentOS Fedora Ubuntu etc.

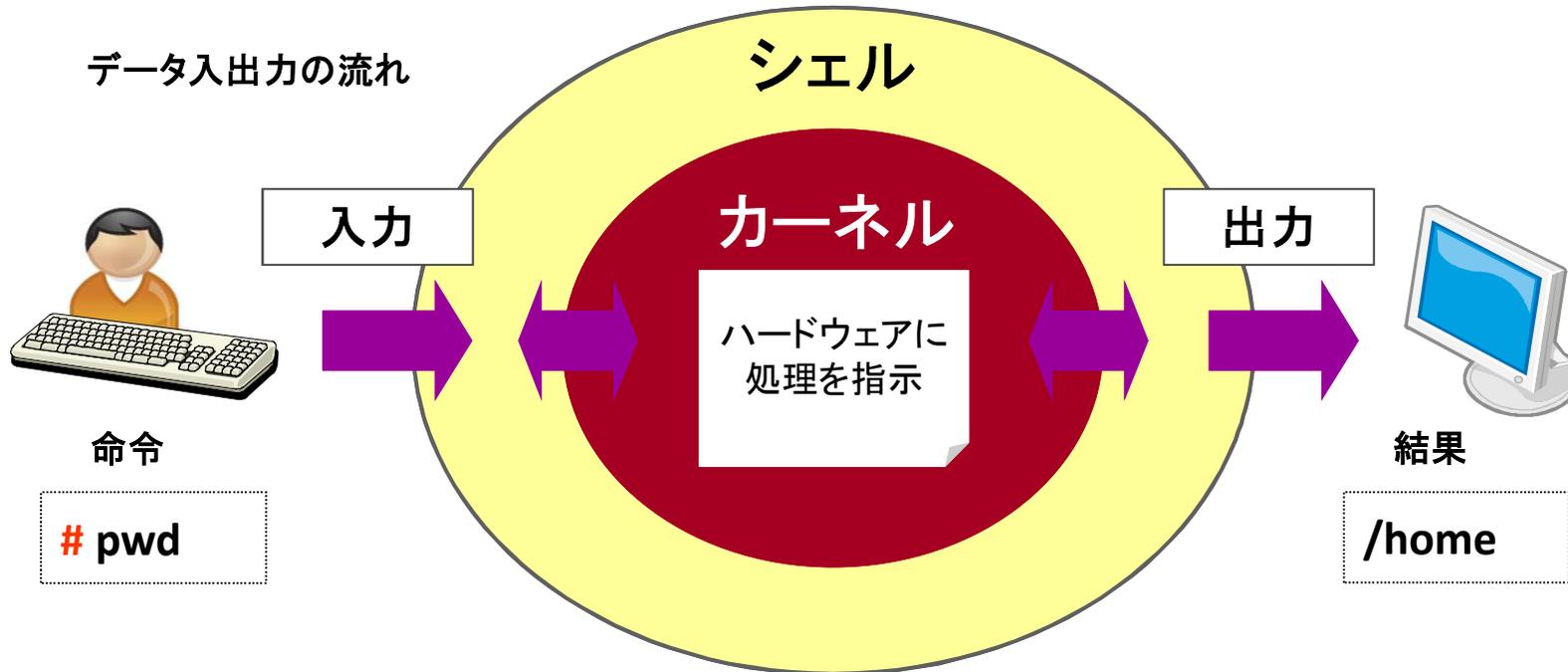


ベンダーに特化したOSではない



LinuxOS

データ入出力の流れ





Linux

Linux

OS

Linuxディストリビューションに含まれるもの

・インストーラ
インストールさせるプログラム

・カーネルモジュール
デバイスドライバなど



・Linuxカーネル
Linux本体 (中核となるプログラム)



・アプリケーション
サーバ機能、コマンドなどの
プログラム



・シェル
ユーザの指示をカーネルに
伝えるプログラム



・オンラインマニュアル
OS上で参照できる
マニュアル





Linux

Red Hat系

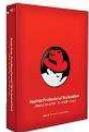
パッケージ管理システムとしてRPM形式を採用

CentOS (無償)



RHELのクローンOS

Red Hat Enterprise Linux



Fedoraをベースにして
安定させた「有償版」のOS
略称：RHEL

Fedora (無償)



レッドハットが支援するコミュニティ
「Fedora Project」によって開発され
最新の技術を積極的に取り込んでいる

Debian系

パッケージ管理システムとしてDebian形式を採用

Debian GNU/Linux (無償)



Debian Project によって開発される

Ubuntu Linux (無償)



使いやすいGUIを目指す
Debian系ではシェアが高い

その他 ディストリビューション

OpenSUSE (無償)



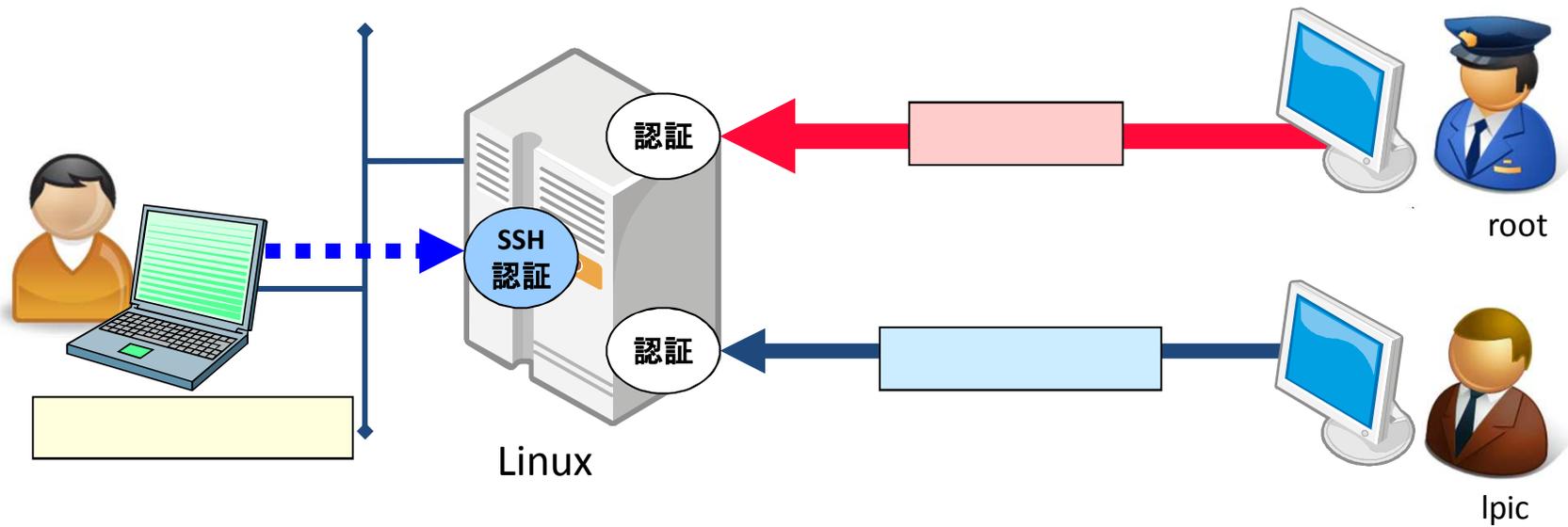
ノベルがスポンサーを務め、コミュニティ
が開発する



Linux

Linux

()
SSH



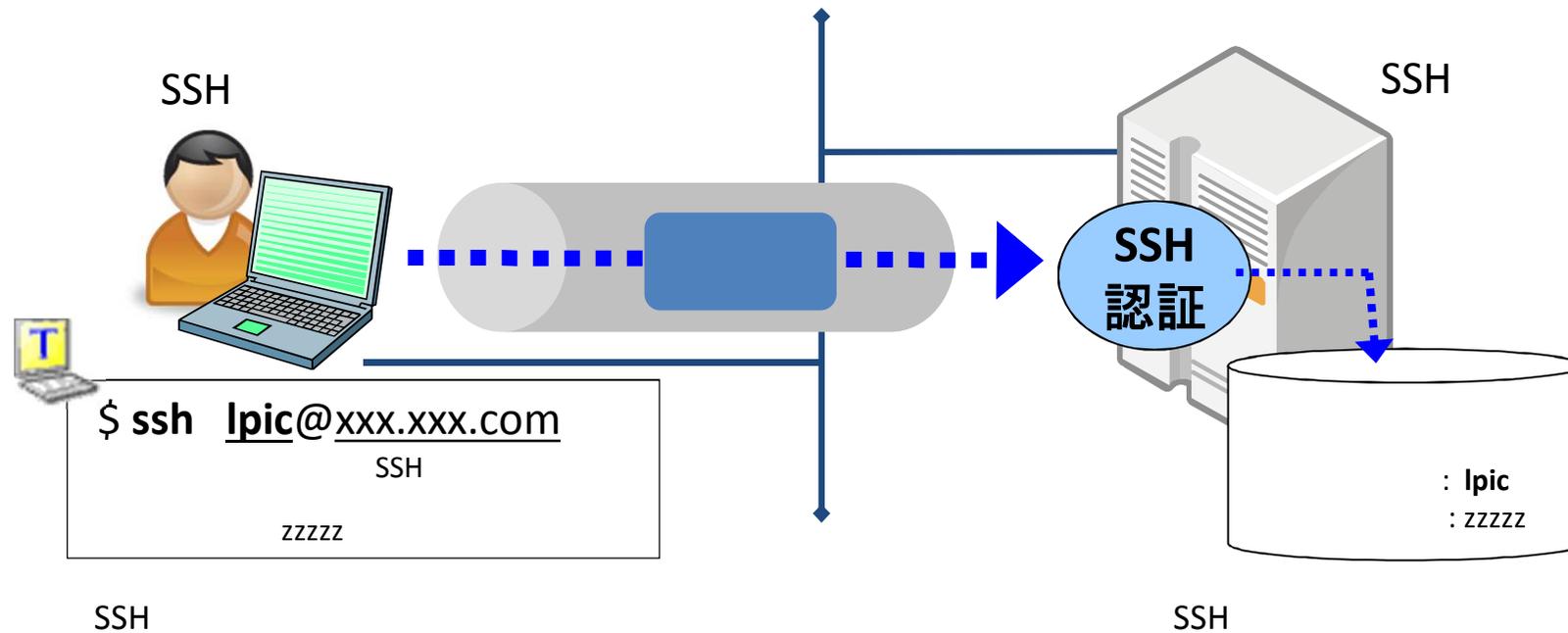


SSH(Secure Shell)

SSH
学びます

LPI102

110.3 暗号化によるデータの保護」で

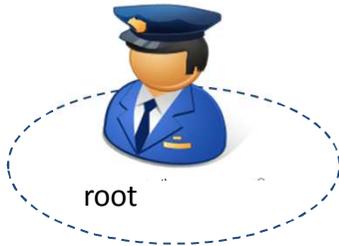




- Linux 2
- 1

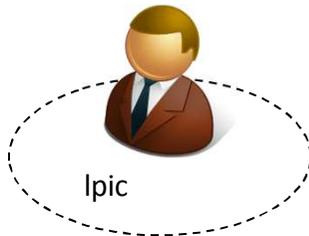
=

= root



1

(lpic)



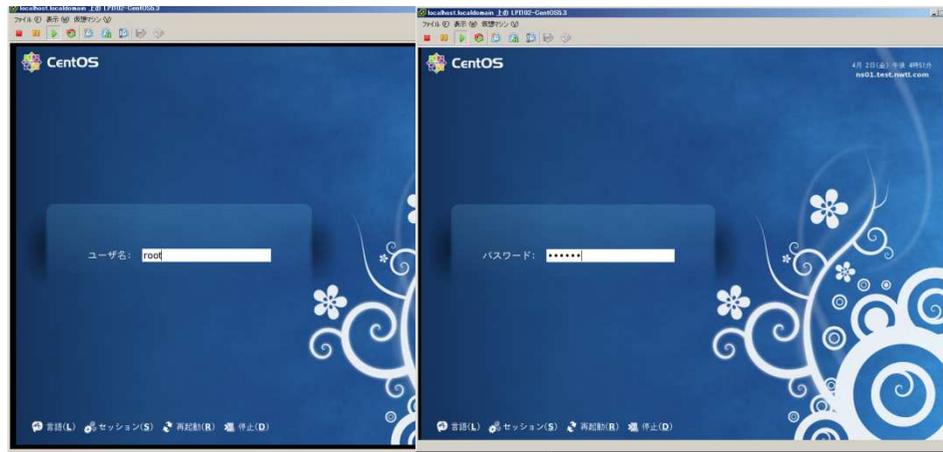
root



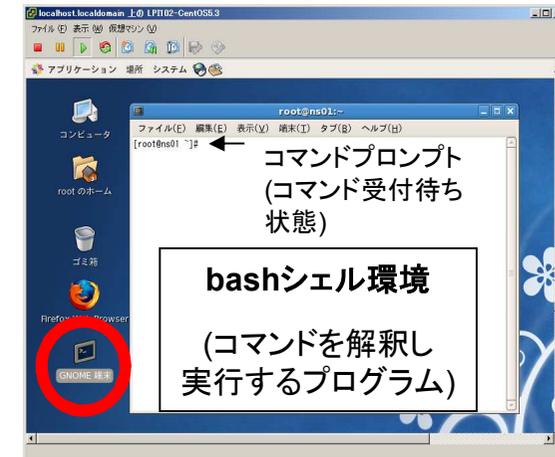
(GUI)

5

NEC
NECネットエスアイ



- ① 「ユーザー名」のあとに **root** を入力して Enter
「パスワード」のあとに **パスワード** を入力して Enter
(※パスワードは表示されません)

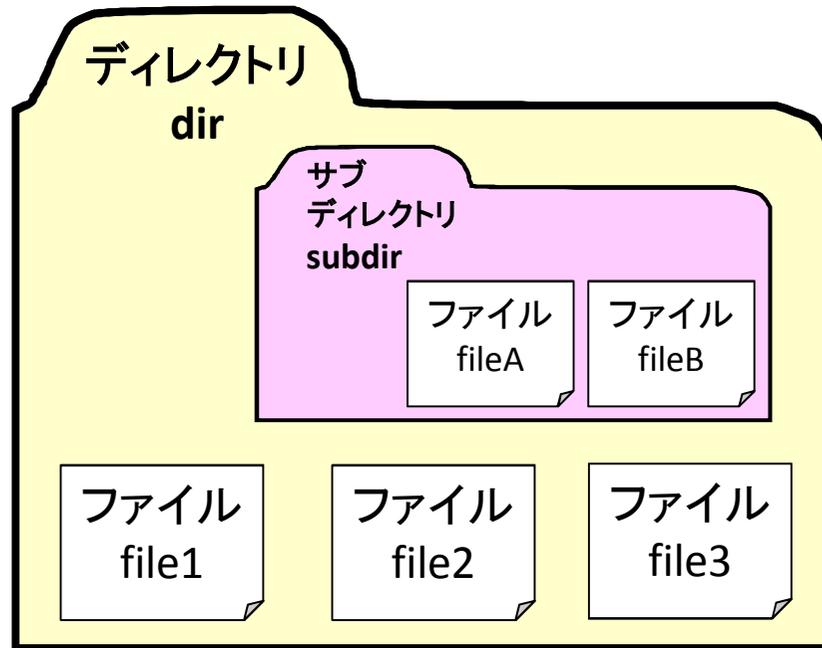


- ② Linuxが起動したら
「GNOME端末」アイコンを
クリックして、コマンド実行の
準備をします

「GNOME端末」を終了し、メニューバーからログアウトを選択する

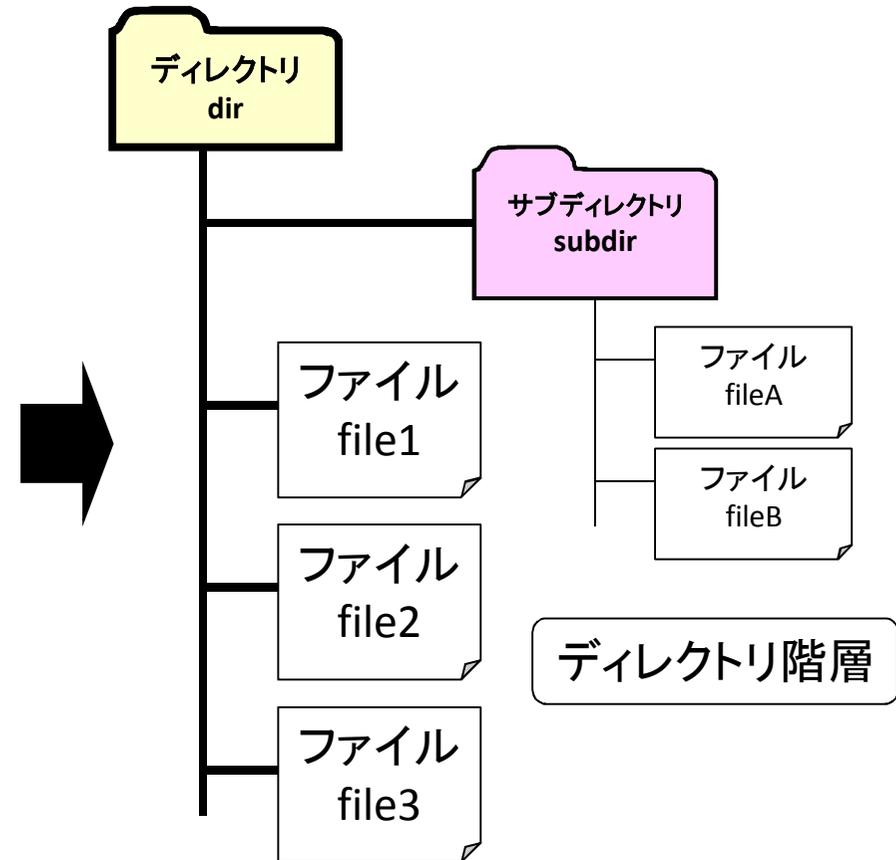


Linux



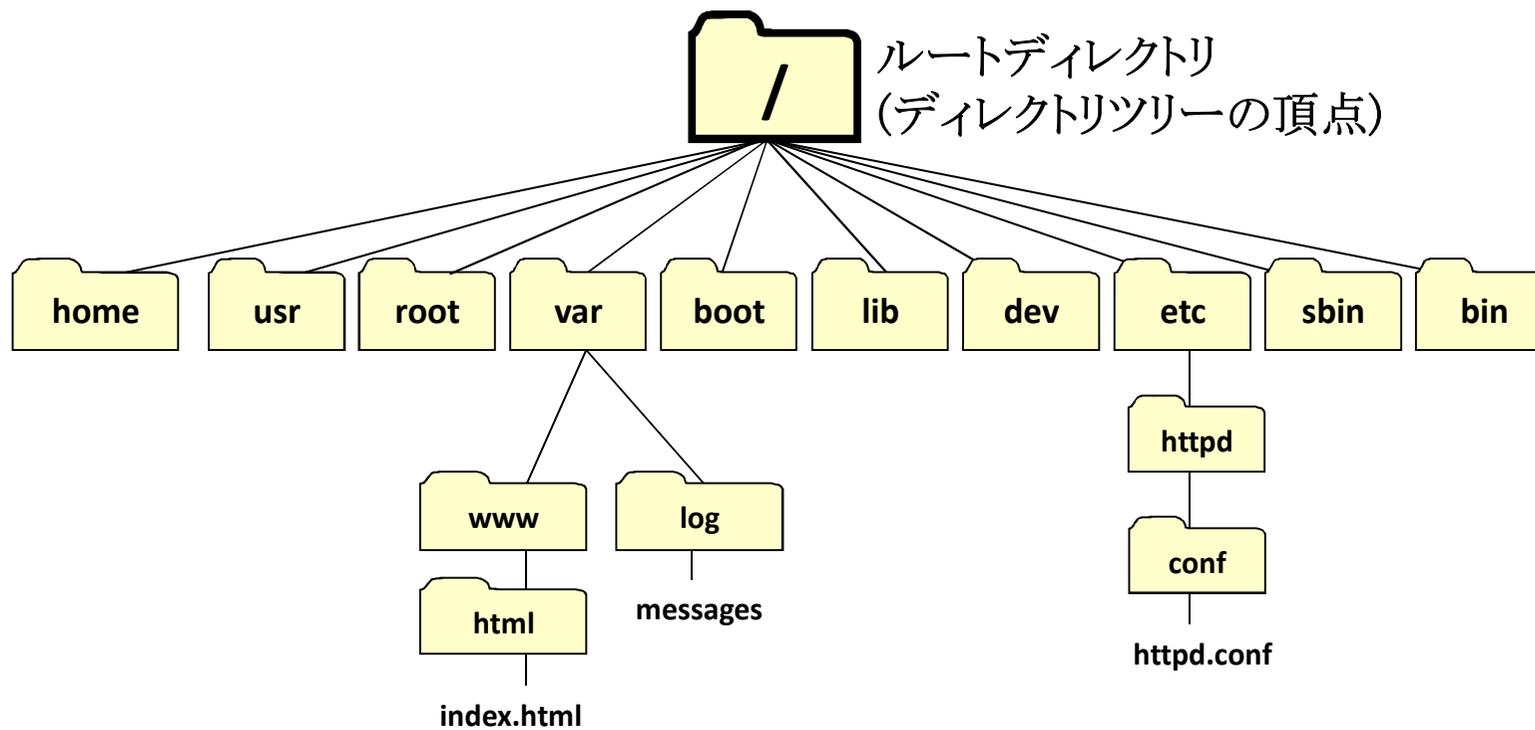
ディレクトリ構成イメージ

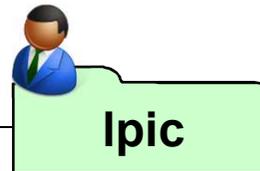
Windows





Linux

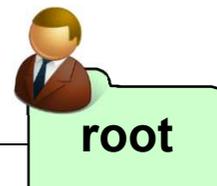




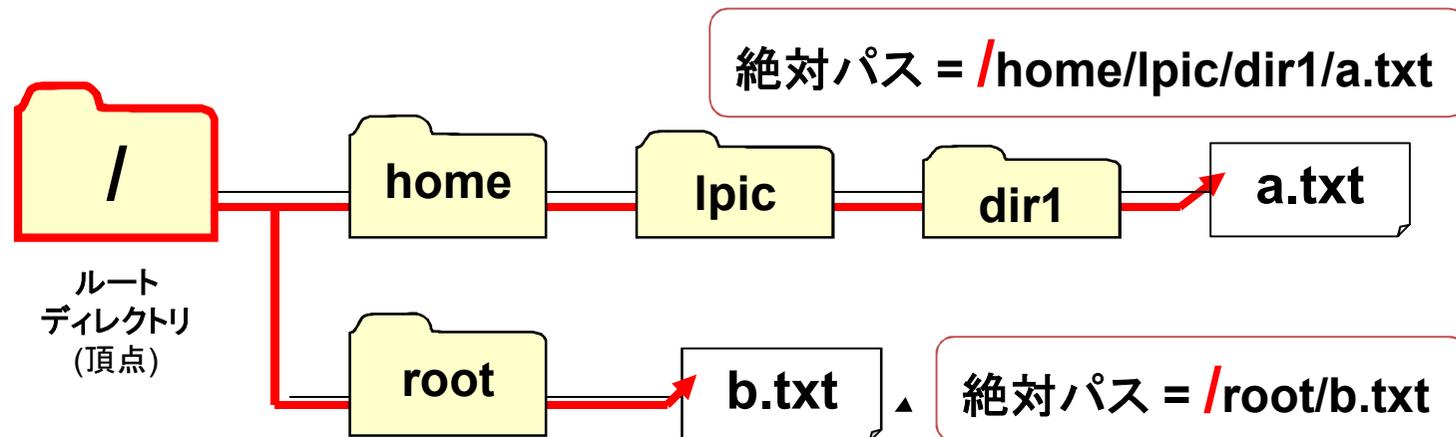
lpicユーザのホームディレクトリ
/home/lpic



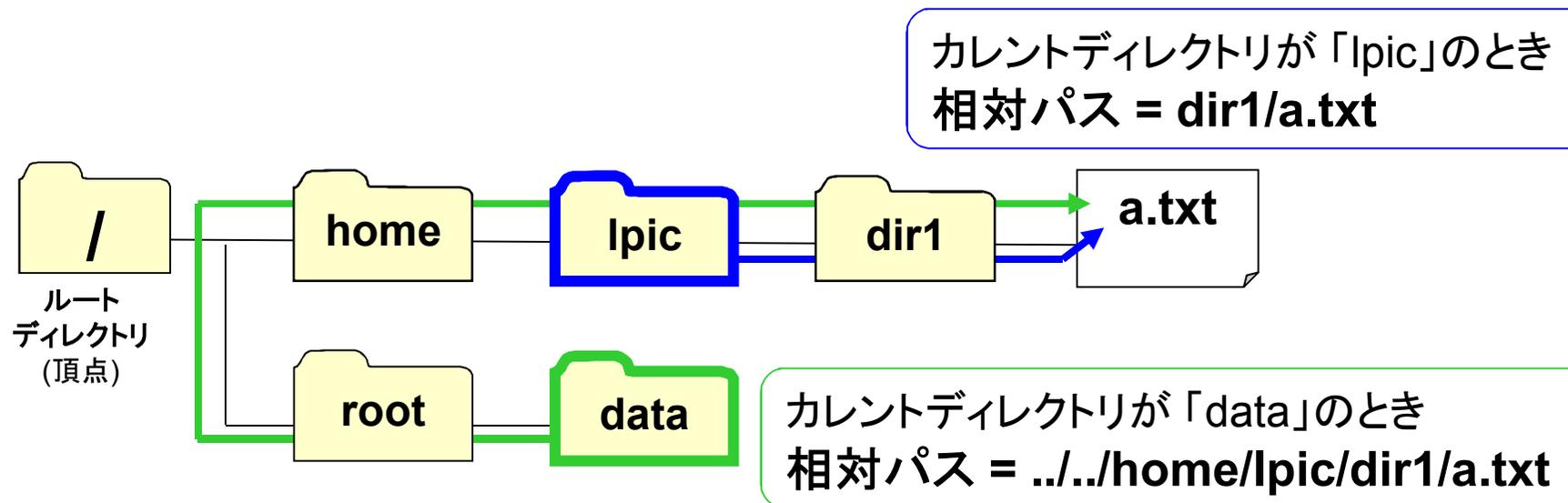
penguinユーザのホームディレクトリ
/home/penguin



rootユーザ(管理者)のホームディレクトリ
/root



`/home/lpic/dir1/a.txt` (`/home/lpic/dir1` は `a.txt` までの経路)



101試験範囲よりポイント解説



◆ 101試験範囲よりポイント解説

- 102.5 RPMおよびYUMパッケージ管理を使用する
- 103.1 コマンドラインで操作する
- 103.3 基本的なファイル管理を行う
- 103.5 プロセスを生成、監視、終了する
- 103.8 viを使って基本的なファイル編集を行う

103.1 コマンドラインで操作する



コマンド

```
# ls  
コマンド
```

コマンド

引数

```
# ls /tmp  
コマンド 引数
```

コマンド

オプション

```
# ls -l  
コマンド オプション
```

コマンド

オプション

引数

```
# ls -l /tmp  
コマンド オプション 引数
```



bash

ファイル名補完

```
# cat /etc/hosts   ←途中でTabキーを2回押す  
hosts hosts.allow hosts.deny ←候補があった場合表示される  
  
# cat /etc/hosts.a  ←「.a」まで入力しTabキーを押す  
  
# cat /etc/hosts.allow ←残りの部分が補完される
```

コマンド名補完

```
# da  ←途中でTabキーを1回押す  
  
# date ←残りの部分が補完される
```



cd	ディレクトリを移動する
\$ cd /tmp	/tmp ディレクトリへ移動する
\$ cd ~ <u>lpic</u> 移動先ディレクトリ	lpicユーザのホームディレクトリへ移動する

pwd	カレントディレクトリを表示する
\$ pwd /etc	現在のカレントディレクトリを表示する → /etc ディレクトリ上にいることが分かる

特殊記号	説明	使い方
~	ホームディレクトリの指定	# cd Δ ~ <u>lpic</u> lpicユーザのホームディレクトリへ移動する
.	カレントディレクトリの指定	# cat Δ ./a.txt カレントディレクトリ以下の a.txt を参照する
..	1つ上(親)ディレクトリの指定	# cd Δ .. 1つ上の親ディレクトリへ移動する



Linux man

マニュアルは内容毎に、セクションが分かれています。

オンライン
マニュアル
「passwd」



1	
2	
3	
4	
5	
6	
7	
8	
9	



man

man passwd

```
PASSWD(1)                                PASSWD(1)

passwd -

passwd [-f|-s] [name]
passwd [-g] [-r|-R] group
passwd [-x max] [-n min] [-w warn] [-i inact] login
passwd {-l|-u|-d|-S|-e} login

passwd
~
```

man 5 passwd

```
PASSWD(5)                                PASSWD(5)

passwd -

passwd

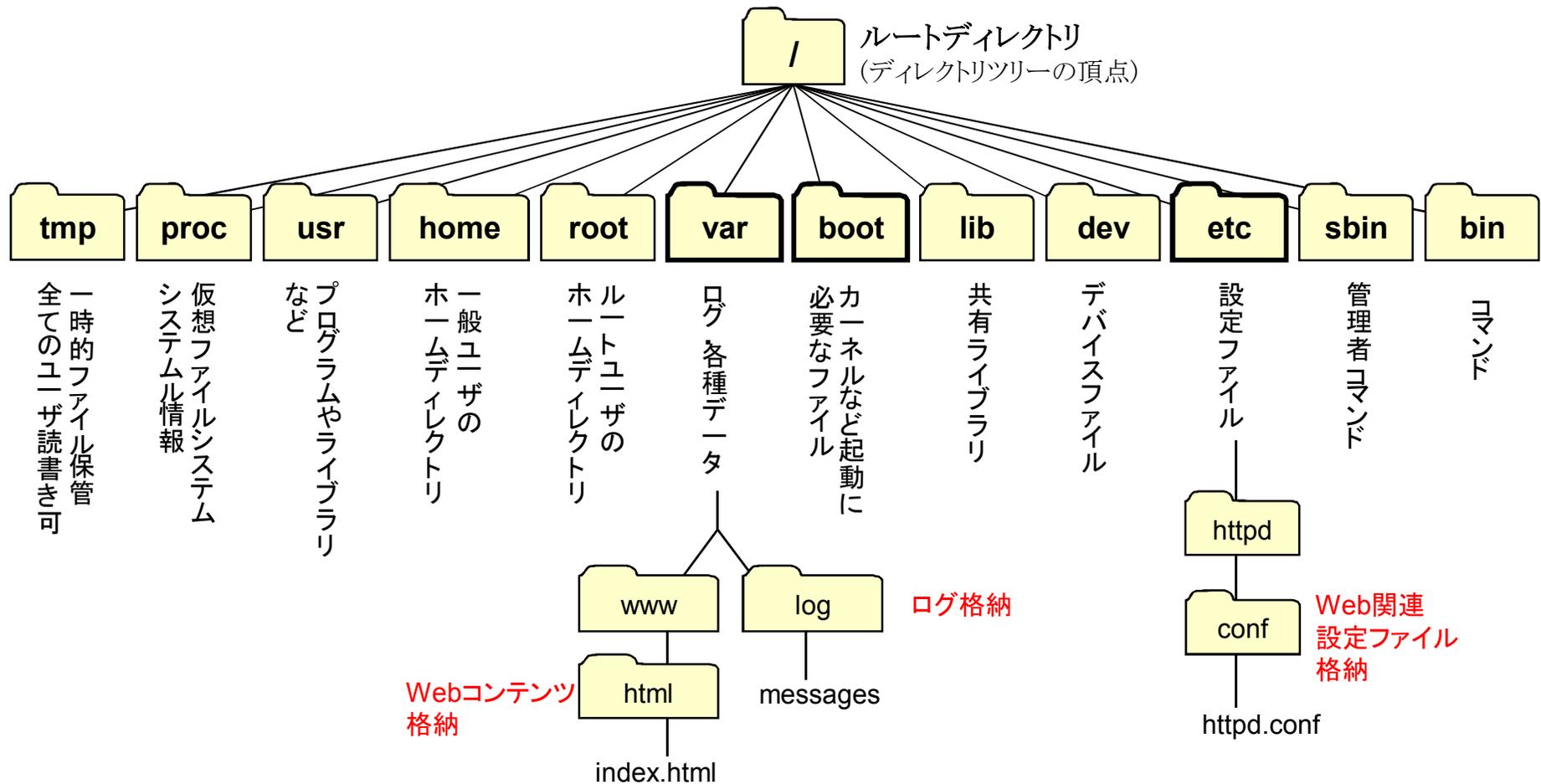
ID ( )
ID

~
```

103.3 基本的なファイル管理を行う



FHS()





(2)

mkdir

ディレクトリを作成する

mkdir △ /root/data

/root の下に data ディレクトリを作成する

\$ mkdir △ **-p** △ ~lpic/dir1/test

親ディレクトリも同時に作成

lpicユーザのホームディレクトリ以下にて
親ディレクトリも同時に作成する

※ test を作成するにはひとつ上の親ディレクトリ dir1 が存在する必要があります
親ディレクトリも同時に作成する場合は **-p** オプションを使います

rm

ディレクトリを削除する

rm △ **-r** △ /root/data

ディレクトリ内全て削除

/root の下の data ディレクトリを削除する

※ディレクトリ削除のときは **-r** オプションを使います

rm: remove directory `/root/data'? **y** 削除するか聞かれるので y を入力する

rm

ファイルを削除する

rm △ /root/b.txt

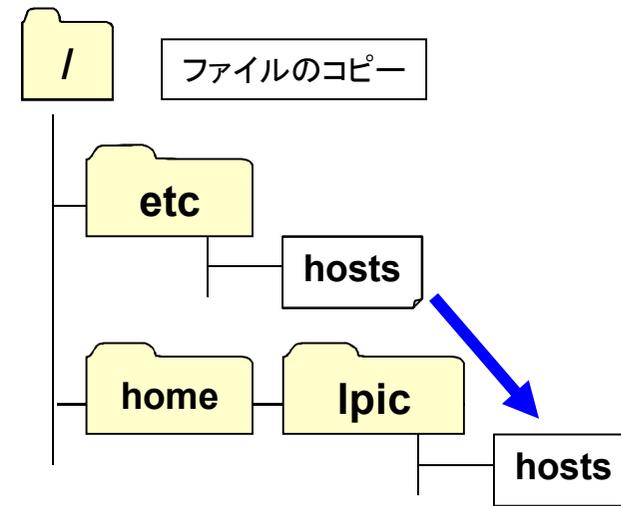
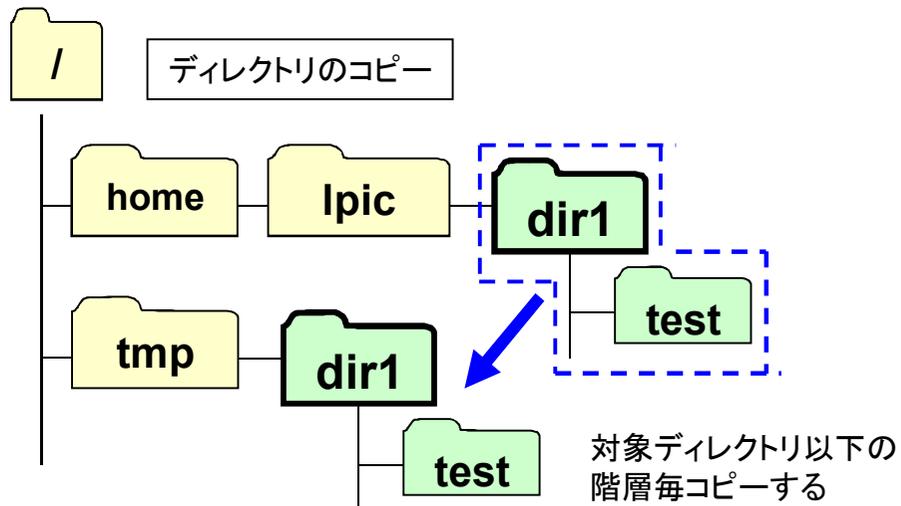
/root の下の b.txt ファイルを削除する
(オプション必要なし)

rm: remove 通常の空ファイル `/root/b.txt'? **y**



(3)

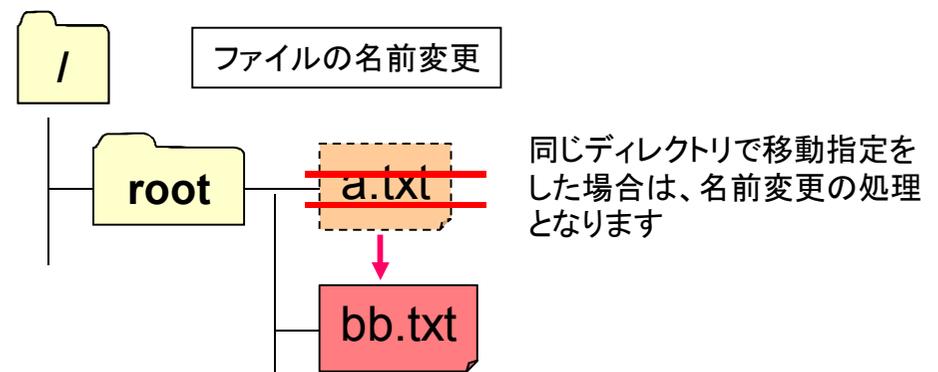
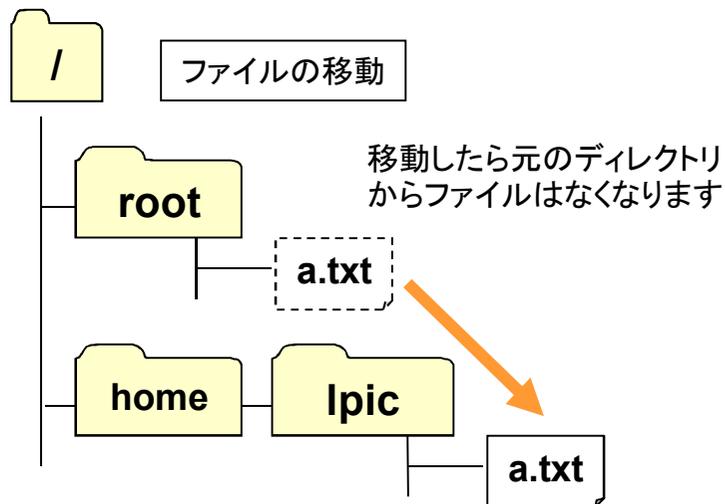
cp	ディレクトリをコピーする
# cp Δ -r Δ ~lpic/dir1 Δ /tmp ディレクトリのコピー コピー元ディレクトリ コピー先	~lpic/dir1ディレクトリを /tmpディレクトリ以下へコピーする
cp	ファイルをコピーする
\$ cp Δ /etc/hosts Δ ~lpic コピー元ファイル コピー先	/etc/hostsファイルをlpicユーザのホームディレクトリへコピーする





(4)

mv	ファイルの移動
# mv Δ /root/a.txt Δ ~lpic <small>移動元ファイル 移動先ディレクトリ</small>	/tmp 以下の a.txt ファイルを lpicユーザのホームディレクトリへ移動する
mv	ファイルの名前変更
# mv Δ /root/a.txt Δ /root/bb.txt <small>変更前ファイル名 変更後ファイル名</small>	a.txt ファイルを bb.txt へ名前変更する ※同一ディレクトリで移動した場合は、ファイル名の変更となります。





cat	ファイルの中身を表示
\$ cat Δ <u>/etc/hosts</u> ファイル名	/etc/hostsファイルの内容を表示する (cat コマンドは行数がたくさんあると表示が流れてしまう)
less	ファイルの中身を先頭から表示 (ページャ)
\$ less Δ <u>/etc/inittab</u> ファイル名 q を入力して終了	/etc/inittabファイルの内容を表示する ← lessは実行すると閲覧モードに入ってしまうため、q(quit)で終了する
head	ファイルの先頭部分を表示
\$ head Δ <u>/etc/inittab</u> ファイル名	/etc/inittabファイルの先頭部分を表示する (デフォルトは 10行)
tail	ファイルの末尾部分を表示
\$ tail Δ <u>/etc/inittab</u> ファイル名	/etc/inittabファイルの末尾部分を表示する (デフォルトは 10行)
\$ tail Δ -f Δ <u>/var/log/messages</u> リアルタイム表示 ファイル名	ログファイルをリアルタイム表示する

103.8 viを使って基本的なファイル 編集を行う



vi

vi エディタの起動

vi △ /etc/resolv.conf →

vi コマンドで viエディタを起動します。

ファイルが既存にあれば
「ファイル編集」,
ファイルが存在しなければ
「ファイル新規作成」となります。

vi エディタ コマンドモード
(メニュー操作)

```
search lpic.biz
nameserver 192.168.60.1
nameserver 192.168.50.252
~
~
~
"/etc/resolv.conf" 3L, 111C
```

vi 起動直後はコマンドモード

vi エディタ インサートモード
(文字の入力)

```
search lpic.biz
nameserver 192.168.60.1
nameserver 192.168.50.252
hama ← 新しい文字を入力できる
~
~
-- INSERT -- ← インサートモード
```

文字入力はインサートモード

インサートモードへの
切り替え

i a →

← Esc 文字入力の終了



vi

- 基本的なコマンドモードでの操作

インサートモードへの切り替え	
コマンド	説明
i	カーソル前にテキスト入力
a	カーソル後にテキスト入力
I	行頭にテキスト入力
A	行末にテキスト入力
o	カーソルの下に行挿入、テキスト入力
O	カーソルの上に行挿入、テキスト入力

カーソル移動	
コマンド	説明
0	行の先頭へ移動する
\$	行の末尾へ移動する
G	ファイル内の最終行へ移動する
:n	ファイル内 n行へ移動する

編集コマンド	
コマンド	説明
x	カーソル位置の文字削除 (Delete)
X	カーソル位置手前の文字削除 (Backspace)
dd	行の削除 (切り取り) 2行分→ 2dd
yy	行のコピー 5行分→ 5yy
p	カーソルの下に貼り付け
P	カーソルの上に貼り付け
u	Undo 直前操作のやり直し

vi の終了	
コマンド	説明
:q!	内容を保存せずに終了する
:wq 又は ZZ	内容を保存して終了する
:W △ファイル名	ファイルを指定して保存する



vi

検索コマンド	
コマンド	説明
/ パターン	カーソル位置から下に向かってパターンを検索する
? パターン	カーソル位置から上に向かってパターンを検索する
n	次を検索する
N	次を検索する (逆方向)
:noh	候補のハイライト表示を解除
:%s/AA/BB/	文字列AAを文字列BBに1つ置換する
:%s/AA/BB/g	文字列AAを文字列BBにすべて置換する

vi の設定変更、シェルコマンドの実行、viの終了	
コマンド	説明
:set nu	行番号を表示する
:set nonu	行番号を非表示する
! Δ コマンド	vi を終了せずにシェルコマンドを実行する
:w!	終了せず保存のみ実行 (上書き保存)

```

1 LPIC Level 1
2 spring
3 summer
4 fall
5 winter
6 LPIC Level 1
6 LPIC Level 1 hamada
7 spring
7 spring hamada
8 summer
8 summer hamada
9 fall
9 fall hamada
10 winter
10 winter hamada
~
~
/hamada ←下方向へ検索

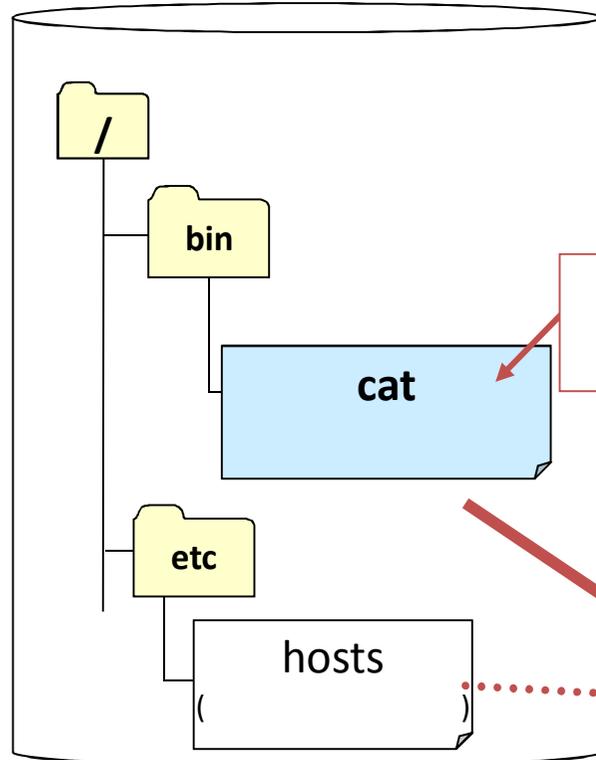
```

カーソル位置から
下方向に検索

103.5 プロセスを生成、監視 終了する



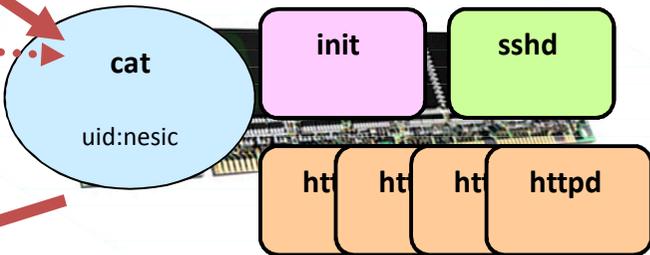
```
$ cat /etc/hosts nesc
```



nesc

ID(PID)

```
[nesc@ns01 ~]$ cat /etc/hosts
127.0.0.1 localhost.localdomain
192.168.25.50 ns01.test.nwtl.com
192.168.25.50 ns01.test.nwtl.com
```





ps	現在実行されているプロセス表示
\$ ps	ユーザ自身が起動しているプロセスを表示
\$ ps <u>△ aux</u>	システム全体で実行されているプロセスを表示
	a .. 他のユーザのプロセスも表示 u .. 実行ユーザ名も表示 x .. 制御端末のないプロセスも表示する
<pre>USER PID %CPU %MEM VSZ RSS TTY STAT START TIMECOMMAND root 1 0.0 0.0 2060 584 ? Ss Mar25 0:00 init [3] root 2 0.0 0.0 0 0 ? S< Mar25 0:00 [migration/</pre>	

aux

オプション `aux` はハイフン(-)を付けずに指定します。

pstree	プロセスの階層構造を表示
\$ pstree -p	ID
<pre>init(1)--acpid(4798) --atd(5083) --login(4422)---bash(4424)---ls(4425) --sshd(4430)</pre>	

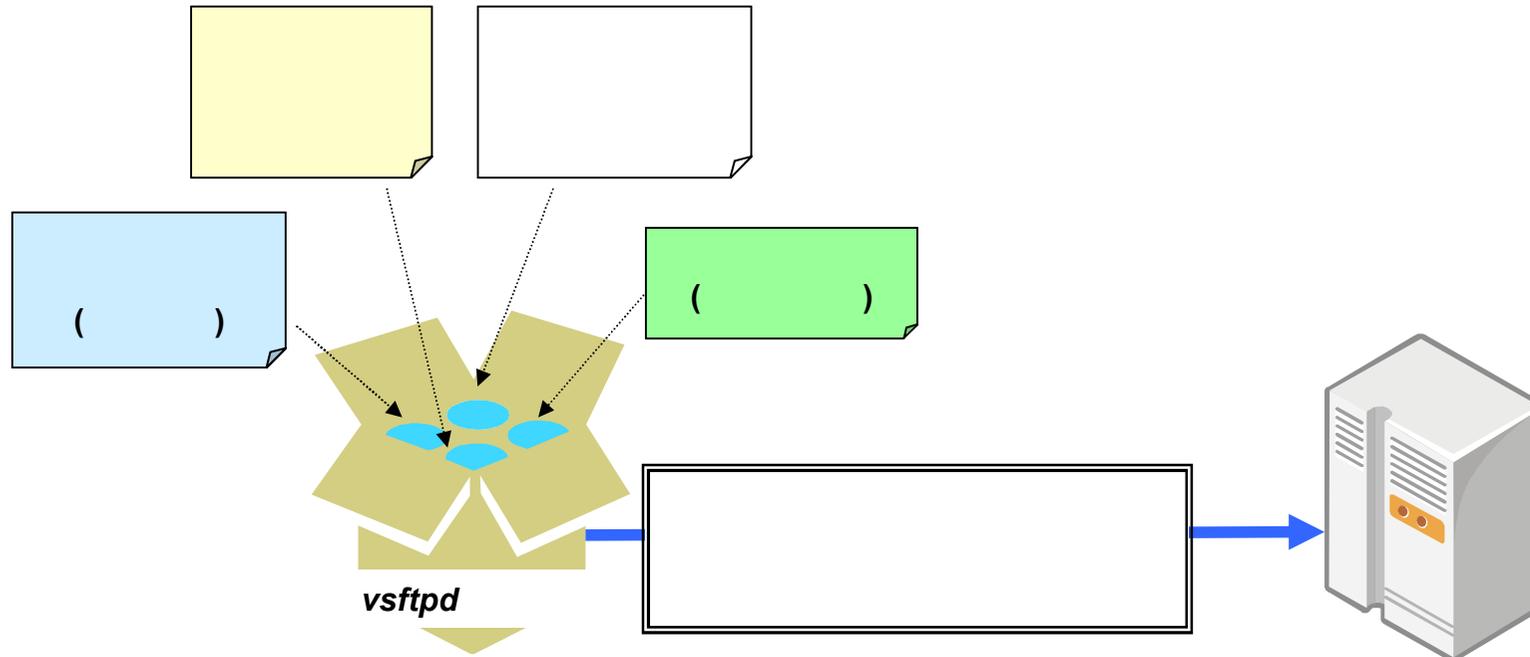


()

kill	指定したプロセスにシグナルを送信
# kill <u>500</u> プロセスID	プロセスID 500番のプロセスを正常終了する (シグナル指定がない場合のデフォルト動作)
# kill -9 100	} プロセスID100番のプロセスを 強制終了する シグナルはどの指定でも動きは同じ
# kill -KILL 100	
# kill -SIGKILL 100	

シグナル		シグナル番号	動作
HUP	SIGHUP	1	設定再読み込み、プロセスの再起動
INT	SIGINT	2	Ctrl + Cキー押下など割り込み通知
KILL	SIGKILL	9	プロセスの強制終了 ※使用注意
TERM	SIGTERM	15	プロセスの正常終了(デフォルト)

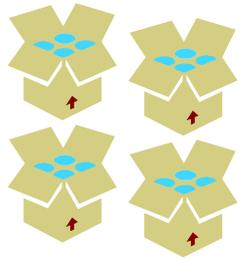
102.5 RPMおよびYUM パッケージ管理を使用する



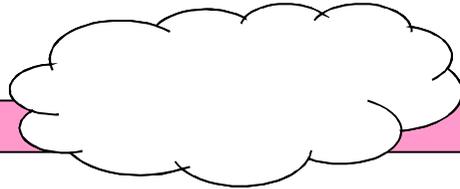
ディストリビューション	Debian系 (Debian, Ubuntu)	RPM系 (RHEL, CentOS, fedora)
パッケージ管理コマンド (パッケージを入手し手動インストール)	dpkg	rpm
パッケージ管理ツール (ネットワーク経由で自動インストール)	APT	YUM
パッケージファイル名	vsftpd-1-2.3.i386.<u>deb</u>	vsftpd-1-2.3.i386.<u>rpm</u>



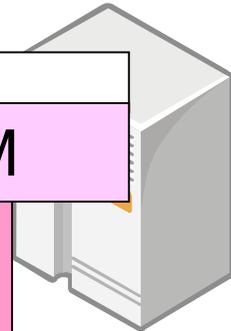
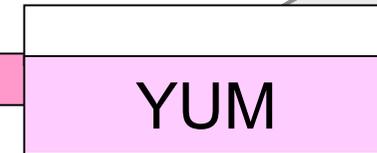
[RPM系] YUM



RedHat



ネットワーク経由でサーバから
パッケージを自動ダウンロードし
インストール又はアップデートを行う
〈パッケージ間の依存関係を解決〉



```
YUM
# yum install apache2
```

```
YUM
# yum update
```

```
YUM
# yum check-update
```

```
YUM
# yum search apache2
```



[RPM] rpm

rpmは RPM形式のパッケージを管理するためのコマンド
あらかじめサーバに用意されたパッケージを使用する
<パッケージ間の依存関係に問題があった場合はエラーとなる>

rpmコマンド インストール

```
# rpm Δ -ivh Δ apache2.1.i386.rpm
```

パッケージファイル名

rpmコマンド パッケージのアップデート(なければインストール)

```
# rpm Δ -Uvh Δ apache2.1.i386.rpm
```

パッケージファイル名

rpmコマンド インストール済みパッケージのみアップデート

```
# rpm Δ -Fvh Δ apache2.1.i386.rpm
```

パッケージファイル名

rpmコマンド パッケージをアンインストールする

```
# rpm Δ -e Δ apache2
```

パッケージ名

rpmコマンド [照会] インストール済みの全てのパッケージ表示

```
# rpm Δ -qa
```



パッケージ管理コマンド

rpm

RedHat系
Linuxサーバ

用意されたパッケージの
インストール・アップデートを行う

パッケージ情報システム

bind ←インストール済み
sendmail パッケージ名

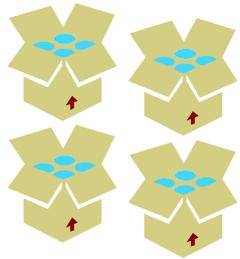
openssh

vsftpd

/usr/bin/vsftpd ←インストールされた
ファイル一覧
/etc/vsftpd.conf
/usr/share/man/vsftpd.gz



[Debian系] APT



Debian



ネットワーク経由でサーバから
パッケージを自動ダウンロードし
インストール又はアップデートを行う
〈パッケージ間の依存関係を解決〉

パッケージ管理ツール

APT

APT パッケージインストール

```
# apt-get Δ install Δ apache2
```

サブコマンド パッケージ名

APT 全てのパッケージをアップグレード

```
# apt-get Δ upgrade
```

APT リポジトリ上最新パッケージ情報を取得

```
# apt-get Δ update
```

APT

```
# apt-cache    search    apache2
```

パッケージ情報システム

```
bind                      ←インストール済み  
sendmail                      パッケージ名  
openssh  
vsftpd  
    /usr/bin/vsftpd    ←インストールされた  
    /etc/vsftpd.conf    ファイル一覧  
    /usr/share/man/vsftpd.gz
```



[Debian系] dpkg

dpkgは Debian形式のパッケージを管理するためのコマンド
あらかじめサーバに用意されたパッケージを使用する
<パッケージ間の依存関係に問題があった場合はエラーとなる>

dpkgコマンド パッケージのインストール

```
# dpkg Δ -i Δ apache2.1.i386.deb  
                パッケージファイル名
```

dpkgコマンド 設定ファイルを残してアンインストール

```
# dpkg Δ -r Δ apache2  
                パッケージ名
```

dpkgコマンド 設定ファイルも含め全てをアンインストール

```
# dpkg Δ -P Δ apache2  
                パッケージ名
```

dpkgコマンド インストール済みパッケージの詳細情報を表示

```
# dpkg Δ -s Δ apache2  
                パッケージ名
```

dpkgコマンド インストール済みパッケージを一覧表示

```
# dpkg Δ -l
```



パッケージ管理コマンド

dpkg

Debian系
Linuxサーバ

用意されたパッケージの
インストール・アップデートを行う

パッケージ情報システム

```
bind                ←インストール済み  
sendmail           パッケージ名  
openssh  
vsftpd  
/usr/bin/vsftpd ←インストールされた  
/etc/vsftpd.conf   ファイル一覧  
/usr/share/man/vsftpd.gz
```

【参考資料】

出題範囲詳細（Ver4.0）

101.2 システムのブート

101.3 ランレベル/ブートターゲットの変更と
システムのシャットダウンまたはリブート



systemd

systemd

Linux

LPI101(Ver4.0)
Upstart

SysVinit ,

CentOS

CentOS5	SysVinit
CentOS6	Upstart
CentOS7	systemd

systemd

Linuxディストリビューション	systemdが採用されたバージョン
Red Hat Enterprise Linux	2014年6月 (ver7.0)
CentOS	2014年7月 (ver7.0)
Fedora	2011年5月 (ver15)
Debian	2015年4月 (ver8)
Ubuntu	2015年4月 (ver15.04)



LPIC 1[101
102]Version 3.5
+
3,456 (3,200)
2013 2 18
ISBN 978-4-04-891147-4



Linux LPIC 1 5
() / ()
2012/09/19
3,990
ISBN- 978-4798127927
※Ver3.5



Linux Ver1.1.1
<http://www.lpi.or.jp/linuxtext/text.shtml>

ご清聴いただきありがとうございました。

ご質問がございましたら、お声掛け下さい。



LPI-JAPAN

<http://www.lpi.or.jp/>



NEC ネットエスアイ株式会社

<http://www.nesic.co.jp/>