bash起動時における設定ファイルの実行順序bashがログインシェルとして起動された場合、（/etc/profileがあれば）まず最初に/etc/profileを読み込んで実行します。

/etc/bash.bashrcファイル(Debian系のシステムで全ユーザーのbash起動時に共通の処理を実行)があれば、/etc/profile内で読み込み実行します。

その後、bashは~/.bash\_profile、~/.bash\_login、~/.profileの順にファイルを探し、最初に見つかったものを読み込んで実行します。

bashがログインシェルではなく対話型のシェルとして起動された場合は、~/.bashrcがあればこれを読み込んで実行します【注5】。

対話的ログインを禁止するにはログインシェルを「/bin/false」または「/sbin/nologin」に設定

usermod -s /bin/false hoge2

chsh -s /sbin/nologin hoge2

年末年始の休暇の間、ユーザ「user3」にアカウントを使用させないようにしたい

「/etc/shadow」の該当ユーザのパスワードフィールドの1文字目に「!」または「\*」を追加する

「/etc/passwd」の該当ユーザのパスワードフィールドの1文字目に「!」または「\*」を追加する

「passwd -l user3」コマンドを実行する

「usermod -L user3」コマンドを実行する

・ifconfigコマンド

ネットワークインターフェースの情報の表示や設定を行う事ができます。

コンピュータが接続されているネットワーク状況を確認するコマンド

ホストに設置された有線LANや無線LANなどのネットワークインタフェースに対し、IPアドレスやサブネットマスク、ブロードキャストアドレスなどの基本的な設定ができる。加えて、現在の設定を確認できる。

ifconfigコマンド使用時の影響範囲

インターフェースのアップ・ダウン状態が変更される

ifconfigコマンドを実行しても、何も影響がない場合もある

・useraddコマンドに関して

-D:デフォルト値を確認することが可能

-d: ホームディレクトリを指定

-s: ログインシェルを指定

デフォルトの値は、「/etc/default/useradd」ファイルに保存

ログインできないユーザーアカウントは、ログインシェルとして/bin/falseを指定する。新規にユーザーアカウントを作成するときは、useraddコマンドで/bin/falseをシェルとして指定する。

# useradd -s /bin/false ＜ユーザーアカウント＞

「useradd」コマンドでユーザを新規作成する際、ホームディレクトリが存在しない場合には、ホームディレクトリが作成され、

「/etc/skel」ディレクトリ以下のファイルが、雛形（スケルトン）としてホームディレクトリにコピーされます。

共通の設定ファイルなどをユーザ作成時に配布する場合などに便利です。

/etc/group

グループに関する情報が記載してある。

・パスワードの変更

passwd

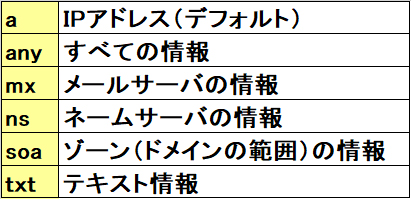
usermod -p

X Window Systemに代わる新しい仕組みとしてWaylandが開発されています。

DNSサーバに問い合わせて名前解決を行うコマンドには、nslookup、dig、hostがありますが、出力する情報が最も少ないのはhostコマンドです。

・digコマンドに関して

DNSサーバへ問い合わせ、DNSサーバへの登録情報を表示するコマンド



・NetworkManagerに関して

NetworkManagerは動的にネットワークを管理する仕組みです。様々なディストリビューションで採用されており、Red Hat Enterprise Linux 6では、従来のinitスクリプトによるネットワーク設定に代わり標準のネットワーク設定として導入されています。

ノートPCなど移動式のデバイスでは、ネットワーク環境を切り替えるのに便利な機能です。ただし、サーバのように設定を固定した方がよい環境では動的に設定が変更されて問題が発生するのを防ぐため、NetworkManagerを無効化することもあります。

NetworkManagerはGUIやCLIで操作できます。

・ssコマンドに関して

ネットワークのソケットの情報を表示

-n:名前解決をしない

-t:TCPソケットを表示

-u:UDPソケットを表示

・ウェルノウンポート(最小値0最大値1023)に関して

20:FTPファイル転送（データ用）

21:FTPファイル転送（制御用）

22:SSHリモートホストの遠隔操作暗号化なし

23:TELNETリモートホストの遠隔操作暗号化なし

25:SMTP電子メール送信

53:DNS名前解決

80:HTTPテキストや画像などのコンテンツの送受信(暗号化なし)

110:POP3電子メール受信

143:IMAP電子メール受信

443:HTTPSテキストや画像などのコンテンツの送受信(暗号化あり)

ウェルノウンポートを使用するにはroot権限が必要になります。一般ユーザ権限では使用できません。

ポート番号は0～65535番まである

・「nmap」コマンド

指定したホストの開いているポートをネットワーク経由で確認（ポートスキャン）する事ができる。

・lsofコマンド（LiSt Open Files）

特定のポートを使用しているプロセスなど、ローカルホストの様々なプロセス情報を表示させることができます。

プロセスが開いているファイルを表示する。

・sshコマンド

ユーザ名＠接続先ホスト名

-l:ログインユーザ名の指定

-p:ポート番号の指定

-i:秘密鍵ファイルの指定

-o:ssh\_configで設定できるオプションの指定

公開鍵と秘密鍵のペアを作成するのは「ssh-keygen」

「ssh-agent bash」コマンドでssh-agentはbashを起動し、バックグラウンドで動作します。ssh-agentによって起動されたシェル内で子プロセスとしてssh-addやsshコマンドを実行すると、これらの認証エージェントとしてssh-agentが機能します。

rootユーザのSSHログインを不許可とする場合は「PermitRootLogin」の項目を「no」とします。（デフォルトはyes）

SSHサーバのデーモン「sshd」の設定ファイルは「/etc/ssh/sshd\_config」です。

・ホスト認証に関して

ホスト認証は、SSHクライアントが接続先のリモートホスト（SSHサーバ）が正しいホストかどうかを確認する（なりすましを防ぐ）ために行います。

クライアントからの接続時に、リモートホストは自身の公開鍵をSSHクライアントに送信します。SSHクライアントは受け取った公開鍵と、SSHクライアントの「~/.ssh/known\_hosts」ファイルに格納されているSSHサーバの公開鍵を比べることで正しいホストかどうかを確認

ユーザ認証

ユーザの公開鍵をサーバーに登録する



・gpgコマンド

-e:ファイルを暗号化する

・crontabコマンドに関して

定期的に自動でジョブを実行する「cron」

crontabファイルの書式設定

[分] [時] [日] [月] [曜日] [実行ユーザー名(システム用設定ファイルでのみ指定)]　[コマンド]

曜日：0,7日曜日,1月.....

・localeコマンド

現在のロケールの設定はlocaleコマンドを使用する。

言語の設定。

「/usr/share/zoneinfo/」ディレクトリ以下には、国や地域ごとにタイムゾーンの情報がバイナリファイルとして格納されています。システムのタイムゾーンは、それらのバイナリファイル群から、システムで使用したいタイムゾーンのものを「/etc/localtime」へコピーまたはシンボリックリンクを作成することで設定します。

export TZ=Asia/Tokyo

「/etc/localtime」と「/etc/timezone」ファイルをまとめて設定できるコマンドは「tzconfig」

・cat /etc/shadow

/etc/shadowには暗号化されたパスワードが記録される

・cat /etc/group

グループ情報を管理するファイルの中身を表示

・cat /etc/skel

新規ユーザー作成時にコピーする雛形（スケルトン）ファイルを配置するディレクトリ

・cat /etc/nologin

一般ユーザのログインを禁止する際に使用するファイルです。通常touchコマンドで、中身の無いファイルを作成します

・cat /etc/passwd

ユーザーのアカウント情報が格納されたファイルです。

・ログインシェルは，ユーザごとに設定ファイル /etc/passwd 内に定義されている

bashはログインシェルとして起動する

/etc/profile、~/.bash\_profile、~/.bash\_login、~/.profileの順番で各ファイルをログイン時に一度だけ読み込んで実行します。

bashは非ログインシェルとして起動した場合、~/.bashrcファイルがあれば起動のたびにこれを読み込んで実行します。

ログインシェル実行手順

①「/etc/profile」の実行

②「/etc/profile」により「/etc/profile.d/」の全スクリプトが実行される

（「/etc/profile.d/\*.sh」が実行される）

③「~/.bash\_profile」の実行

④「~/.bash\_profile」により「~/.bashrc」が実行される

⑤上記が存在しない場合は「~/.bash\_login」⇒「~/.profile」と順番に読み込んでいきます。

■非ログインシェルの動作

ノンログインシェルは環境をセットするために以下の処理を実行します。

①「~/.bashrc」の実行

②「~/.bashrc」により「/etc/bashrc」が

③「/etc/bashrc」により「/etc/profile.d」配下のスクリプトを実行

・bashをログインシェルとするユーザがログインすると/etc/profileですべてのユーザに共通する環境の設定を行うことができます。

・非ログインシェル起動時にどのユーザにも共通した設定を行うファイルとして/etc/bashrcが使われています。

コマンドの実行結果をシェル変数に代入するには、以下の2通りの方法があります。

①コマンドを$()で囲んで、その結果をシェル変数に代入する

②コマンドをバッククォート「`」で囲んで、その結果をシェル変数に代入する

・suコマンドに関して

suコマンドには以下の特徴があります。

→引数に「-」をつけると、切り替わる先のユーザの環境変数を使用する

→ユーザー名を指定しない場合、rootユーザへ切り替わる

→ユーザを切り替える時に求められるパスワードは、切り替わる先のユーザのパスワード

→但し、rootユーザから別のユーザに切り替わる時はパスワードは要求されない

・sudoコマンドに関して

sudo -l : 一般ユーザが持っている権限の確認

・ulimitコマンドに関して

「ulimit」はユーザやシェルが利用できるリソースを制限するコマンドです。

ユーザーの誤操作やプログラムのバグ、悪意のあるユーザーによるリソースの消費などを回避する為などに使用されます。

-a 制限の設定値をすべて表示

-c 生成されるコアファイルのサイズを指定

-f シェルが生成できるファイルの最大サイズをブロック単位で指定  
(出力ファイルのサイズを制限)

-n 同時に開くことのできるファイルの最大数

-u 1人のユーザーが利用できる最大プロセス数を指定

-v シェルとその子プロセスが利用できる最大仮想メモリサイズ

・changeコマンドに関して

「chage」はパスワードの有効期限の設定に特化したコマンド（CHange password AGEの略）で、詳細な設定やその表示が可能です。

なお、パスワードの有効期限に関する情報は「/etc/shadow」ファイルに保存されます。

-E:ユーザーアカウントの有効期限を指定

-m:パスワード変更の間隔の最短日数を指定（minimam）

-M:パスワードが有効な最長日数を指定(max)

短期のアルバイト社員「user1」のユーザーアカウントを、2010年4月20日までの期限付きにしたい。

chage -E 2010-04-20 user1

・システムロック、ハードウェアロック

ハードウェアロックの参照や設定を行うときはhwclockコマンドを使用する

オプション

-r:ハードウェアロックを表示

-s:ハードウェアロックの時刻をシステムロックに反映させる

-w:システムロックの時刻をハードウェアロックに反映させる

デーモンに関して

**メモリ上で「さぁ、いつでもかかってこい」と待機している常駐プログラムのUNIX系OSにおける呼び名**

・setコマンドに関して

シェルの設定を確認、変更するコマンド

-oで指定するとオプションが有効になり、+oで指定するとオプションが無効になる

オプションの説明

-a, allexport 作成・変更した変数を自動的にエクスポートする

emacs emacs風のキーバインドにする

ignoreeof Ctrl+Dによってログアウトしないようにする

-C, noclobber 出力リダイレクトによる上書きを禁止する

-f, noglob メタキャラクタを使ったファイル名展開を無効にする

vi vi風のキーバインドにする

-n, noexec シェルスクリプトを読み込むが、実行はせずに構文エラーのみ確認

・aliasコマンドに関して

alias [エイリアス名='コマンド']

コマンドが2語以上になる場合は、シングルクォート（'）もしくはダブルクォート（"）で括る

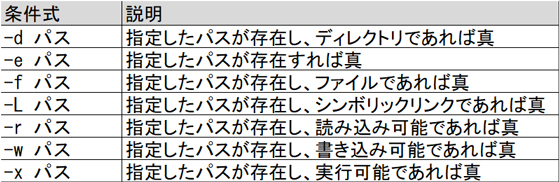
・unaliasコマンドに関して

エイリアスを解除する。

一時的にエイリアスを使用しない場合は、実行するコマンドの前に「\」を付けると、そのときだけエイリアス設定が無視されます。

・関数の定義（{}このニョロかっこ必要なので注意）

[function]関数名(){コマンド;}



functionlslink(){lsl|grep'^l';}

関数の確認

declare -f 関数名

・unsetコマンド

関数や変数の削除

・bashの設定ファイル

/etc/profile ログイン時に実行され、全ユーザーから参照される（基本的な変数が設定される。）

/etc/bash.bashrc bash起動時に実行され、全ユーザーから参照される

/etc/bashrc ~/.bashrcから参照される

~/.bash\_profile ログイン時に実行される（そのユーザ独自の設定を行う。）

~/.bash\_login ~/.bash\_profileがない場合、ログイン時に実行される

~/.profile ~/.bash\_profileも~/.bash\_loginもない場合、ログイン時に実行される

~/.bashrc bash起動時に実行される（対話型シェルが起動されるたびに実行される。）

~/.bash\_logout ログアウト時に実行される

bash実行ファイル設定順序

<https://qiita.com/tatesuke/items/88629e9550b813109964>

・シェルスクリプトの実行

bash シェルスクリプト

source シェルスクリプト

. シェルスクリプト

sourceコマンドや「.」コマンドを使った場合は、元のシェル上で実行されます

文字列が等しい→=

数値が等しい→-eq

スクリプトファイルに実行権を追加すると、ファイル名を指定するだけで、コマンドのように実行できるようになります。

$chmoda+x lsld

$ ./lsld

execコマンドを使うと、シェルスクリプトを実施しているシェルのプロセスが、指定したコマンドのプロセスに置き換わります。

普通にシェルスクリプトを実行したらbashのプロセスIDがそのまま残り（裏でプロセスが動いたまま）、新しいプロセスIDで実行されるがexecコマンドを使用するとそれが置き換わる。

・シェル変数の特殊変数

$0 シェルスクリプトファイル名(フルパス)

$1 1番目の引数

$2 2番目の引数。n番目も同様（10番目以降は${10}のように記載する）

$# 引数の数

$@ すべての引数(スペース区切り)

$\* すべての引数(区切りは環境変数IFSで指定されたもの)

$$ 現在のシェルのPID(プロセスID)

$? 実行結果確認

シェルスクリプトに最初に記載する文字

#!/bin/bash

Xサーバは、モニターやビデオカード、キーボードといったハードウェアの管理を行います。Xクライアントはユーザーアプリケーションで、Webブラウザやオフィスアプリケーションなどが相当します。

利用するXサーバは環境変数のDISPLAYで設定することができる

環境変数DISPLAYは以下のような書式になる。

[ホスト名]: ディスプレイ番号

X.Orgの設定

設定は/etc/X11/xorg.confファイルのほか、/etc/X11/xorg.conf.dディレクトリ以下に複数の.confファイルとして配置されることもあります

デフォルトの設定ファイルは/usr/share/X11/xorg.conf.dに配置されています。手動で設定が必要な場合は、設定を変更するファイルを/etc/X11/xorg.conf.dディレクトリ以下にコピーし、編集します。

セクションの書き方!

Section "InputDevice"

Identifier "Keyboard0"

Driver "kbd"

EndSection

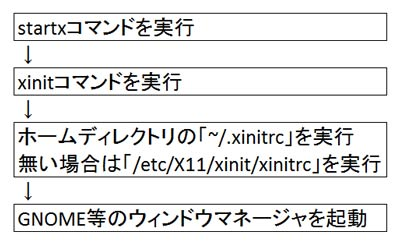
GUI環境に問題がある場合は、Xのログファイル（/var/log/Xorg.0.log）やデスクトップアプリケーションのログファイル（~/.xsession-errors）を確認し、原因を調べます。

​​​​​​​​​​

​​

・Xサーバの起動

startxコマンドを実行すると、XWindowSystemが起動します



・xhostコマンド

XクライアントがXサーバを利用できるよう許可を与える

xhostは、Xサーバへのアクセス制御を調整するコマンドです。+オプションで許可するホストを、－オプションで許可を取り消すホストを指定する。

・リモートデスクトップに関して

VNC クロスプラットフォーム対応（、LinuxやWindows、macOSなど異なるOS間で接続できるリモートデスクトップソフトウェア）

RDP Windows標準

SPICE RDPと同様の画面転送プロトコル。オープンソース。4画面までのマルチモニタやネットワーク経由でのUSB転送など、VNCがサポートしていない多くの機能をサポートしています。

XDMCP ディスプレイマネージャをネットワーク越しに利用できるプロトコル（リモートホストにグラフィカルログインする場合、使用されるプロトコル）

GUI環境をインストールしても、デフォルトのログインがCUIである場合、システム起動時にディスプレイマネージャが自動的に有効になるよう設定する。(LightDMの場合は以下)

# systemctl enable lightdm.service

ディスプレイマネージャに関する説明(三つ)

・ユーザにGUI環境のログインを提供する

・initやsystemdによりディスプレイマネージャーサービスが起動される

・デスクトップ環境を準備する

・/etc/passwdファイル

→ユーザアカウント情報

現在ではシャドウパスワードを使い、パスワード情報は/etc/shadowに格納するようになっています。

・useradd

ユーザアカウント作成コマンド

-m:ホームディレクトリを作成

・usermod

既存のユーザアカウントの変更

-d:ホームディレクトリを指定

-g:プライマリグループを指定

-G:プライマリグループ以外に参加するグループを指定

-s:ログインシェルを指定

-L:パスワードをロックする

-U:パスワードをアンロックする

-p:暗号化済みのパスワードを設定する。

アカウントを一時不使用にするときは

usermod -L アカウント名

・userdel

ユーザアカウントの削除

・id

ユーザー情報を表示

UIDやGIDを表示

ルートユーザはIDが０

・getent (名前解決を行う！)

LDAPサーバーの情報を表示

getentコマンドでnsswitch.confに記載してあるようなことが見れる。

Linuxでは、定期的に実行するジョブについてはcronを、1回限りのジョブの予約についてはatコマンドを使って、スケジューリングできます。

定期的にジョブを実行するcronは、スケジュールを管理するデーモンであるcrondと、スケジューリングを編集するcrontabコマンドから構成

crontabファイルを編集するには、crontabコマンドを使用する

ユーザーのcrontabファイルとは別にシステム用のcrontabファイル（/etc/crontab）もあります。/etc/crontabファイルでは一般的に、そこから/etc/cron.\*ディレクトリに置かれたファイルを呼び出すようになっています。/etc/crontabファイルには、実行ユーザー名を指定するフィールドが加わります。

cronを利用するユーザーを制限するには、/etc/cron.allow、/etc/cron.denyを使います。

systemdのtimerユニットは、定期的に自動でジョブを実行するジョブスケジューラとしてcronの代わりに使用できます。

studentユーザのcrontabファイルは

/var/spool/cron/student

または

/var/spool/cron/crontabs/studen

・atコマンドに関して

1回限りの実行スケジュール

-d/-r 予約中のジョブをジョブ番号指定で削除(=atrm)

-l 予約中のジョブを表示(=atq)

-f コマンドを記述したファイルを指定

/etc/at.allow、/etc/at.denyでアクセス制御

ジョブスケジュールの一覧表示

at -l

atq

・ロケールに関して

LC\_CTYPE 文字の分類、文字の種類の規定

LC\_NUMERIC 数値の書式

LC\_TIME 日付・時刻の書式

LC\_MESSAGES 出力メッセージの言語

LC\_MONETARY 通貨の書式

LC\_NAME 名前の書式

LC\_ADDRESS アドレス、ロケーションの書式

LC\_TELEPHONE 電話番号の書式

LC\_ALL 指定した値が全カテゴリを上書き、個別設定はできない

LANG デフォルト値

LC\_ALLは全カテゴリがそのロケールになり、

LANGはデフォルトとして使用され、カテゴリごとに個別に設定することができると

・iconv(International Codeset Conversion Library)

文字コードの変換

-f:変換まえの文字コードを指定（from）

-t:変換後の文字コードを指定(to)

ASCII 7ビットの文字コード。128種類の文字(英数字、記号、制御コード)を表現

ISO-8859 ASCIIを8ビットに拡張したもの。256種類の文字を表現

Unicode 他言語を扱うために作成された文字コード

UTF-8 Unicodeで定義された文字を表示する符号化方式(ASCIIと互換)

EUC-JP UNIX環境で標準的に利用される日本語文字コード

Shift\_JIS Windowsで標準的に利用される日本語文字コード

現在のシステム時刻や状態を表示するsystemdのコマンド

timedatectl

timedatectl status

タイムゾーンの情報は、/usr/share/zoneinfoディレクトリ以下のバイナリファイルに格納

/etc/localtimeにコピーすることでタイムゾーンを設定することが可能

・dateコマンド

システムロックを確認する。

・システムクロックを表示する

・システムクロックを設定する

・任意の形式で日付時刻を表示する

・hwclockコマンドに関して

ハードウェアクロックを参照・設定

-r ハードウェアクロックを表示

-w(--systohc) システムクロックの時刻をハードウェアクロックに設定

-s(--hctosys) ハードウェアクロックの時刻をシステムクロックに設定

NTPを使用しインターネット上にあるNTPサーバから正確な時刻を取得する。

ntpdateコマンド

NTPサーバから現在時刻を取得

公開ntpサーバから現在の時刻を持ってくる場合

ntpdate pool.ntp.org(ntpdateで持ってくる！！！)

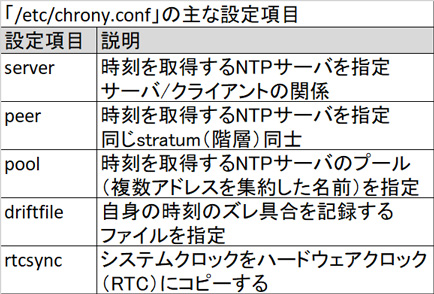
ntpdateの説明

NTPサーバから正確な時刻を取得し、直ちにシステムクロックに反映させる

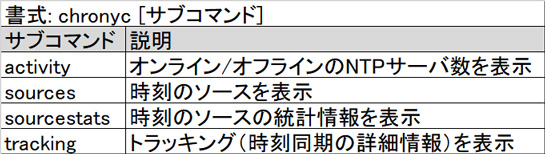
NTPクライアントである

ntpdの各種設定は「/etc/ntp.conf」で行います。

Chronyはntpd/ntpdateの代替となるNTPサーバ/クライアントソフトウェアです。デーモンプロセスchronydと、クライアントコマンドchronycから構成される



chronyコマンドに関して



ファシリティに関して

ファシリティはログメッセージの生成元をさす。

auth（security） 認証サービスのメッセージ（現在はauthprivが推奨されている）

authpriv 認証サービス（カテゴリはauthと同じ。authとは出力結果が異なる）

cron cronのメッセージ

daemon デーモンのメッセージ

kern カーネルのメッセージ

lpr プリンタサービスのメッセージ

mail メールサービスのメッセージ

news ニュースサービスのメッセージ

syslog syslogのメッセージ

user ユーザープロセスのメッセージ

uucp uucp転送を行うプログラムのメッセージ

local0～7 アプリケーションに依存する

プライオリティ

メッセージの重要度をさす。

emerg emergency:パニックの状態でシステムは使用不可

alert alert:緊急に対処が必要

crit critical:緊急に対処が必要。alertより緊急度は低い

err error:エラー発生

warning warning:警告。対処しないとエラーが発生する可能性がある

notice notice:通常では無いが、エラーでも無い

info information:通常の稼働の情報

debug debug:デバッグ情報

none none:ログメッセージを記録しない

・ファシリティープライオリティーのルール

ファシリティ.プライオリティ アクション

・ファシリティとプライオリティは「.（ドット）」でつなぎます

・ファシリティとプライオリティをあわせて「セレクタ」と呼び、セレクタによって扱うログメッセージを特定します

・全てのファシリティや全てのプライオリティを指定するには「\*（アスタリスク）」を使います

・プライオリティは、指定したレベル以上のログを出力します（errを指定すると、emerg, alert, crit, errが出力される）。特定のレベルのログだけを指定するには、「=（イコール）プライオリティ」とします。

・アクションには、出力先を指定します。出力先は、ファイルであれば絶対パス、外部のsyslogサーバであれば「@サーバアドレス」とします。出力先に「/dev/console」を指定すると、ログメッセージはコンソールに表示されます。また、緊急メッセージなど全ユーザーのコンソールに送信したい場合は「\*（アスタリスク）」を指定します。

・「;（セミコロン）」でつなぐことで、1つのアクションに対して複数のセレクタが指定できます

・lastコマンド

最近ログインしたユーザの一覧表示

/var/log/wtmpファイルを参照し、最近ログインしたユーザの一覧を表示するコマンド

設定ファイル/etc/systemd/journald.confで「Storage=persistent」となっていれば、ジャーナルは/var/log/journalディレクトリ以下に永続的に保存されます。「Storage=auto」となっていて、かつ/var/log/journalディレクトリが存在していなければ、ジャーナルは/run/log/journalディレクトリ以下に格納されます。

journalctlコマンドでのログ出力において、前回の起動からシャットダウンまでのログを出力

/var/log/journal ディレクトリが存在する

journalctl -b -1

systemdの動作するシステムではsystemd-journaldデーモンを動作させ、ログの一元管理を行います。systemd-journaldはsystemdから起動したプロセスの標準出力やsyslogへのログメッセージをバイナリ形式で記録します。

systemd-journaldの設定は「/etc/systemd/journald.conf」で行います。

systemd-journaldに関して（三つ）

・syslogのファシリティとプライオリティに対応している

・「/run/log/journal」配下に保存されたログは、システム再起動後はクリアされる

・バイナリ形式のファイル名は「system.journal」である

指定した実行結果をsystemd-journald のジャーナルファイルに書き込むコマンドはsystemd-cat

systemdの動作するシステムで、sshd.service ユニットからのログのみを出力させたい。

journalctl -u sshd.service

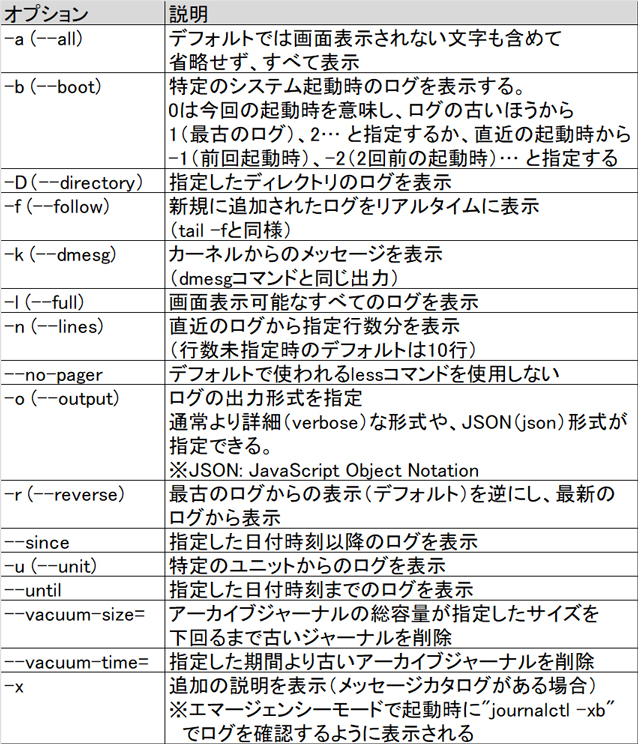
journalctl \_SYSTEMD\_UNIT=sshd.service

・journalctlコマンドに関して

systemd-journaldが収集したログを表示するためのコマンドです。時刻、ユニット名等を指定して、特定のログの絞り込みを簡単に行うことができます。

-o:ログの出力

-n(--lines):直近のログから指定行表示



CUPS

→印刷サブシステム

etc/cups/cupsd.confファイル

UNIXの印刷サービスCUPSのデーモン「cupsd」の設定ファイルです。接続要求を受けるポート番号（デフォルトは631)や、接続するクライアントのアクセス制御などの設定を行います。

/etc/cups/printers.conf

プリンタに関する設定を記述します。

/var/spool/cups/

プリントスプールの場所

・lprコマンド

ファイルを印刷するコマンド（印刷ジョブを生成し、プリントキューに登録）

-#数　指定した部数を印刷する。

-P 指定したプリンタに出力する。

・lprmコマンド

プリントキューにある印刷ジョブを削除

・lpqコマンド

プリントキューの内容を表示する。(プリントキューにある印刷ジョブを表示)

IPv6のIPアドレスは、ユニキャストアドレス、エニーキャストアドレス、マルチキャストアドレスに分類

IPv6ではブロードキャストが廃止され、一斉送信は全てマルチキャスト（特定グループ宛通信）で行われる

前半64ビットを「サブネットプレフィックス」、後半64ビットを「インタフェース識別子（Interface-ID）」とよびます。

IPv6の特徴

・128ビット表記による事実上無限のIPアドレス

・IPアドレスの省略表記が可能

・IPアドレスの自動割り当て機能を標準で実装

・IPsec（暗号化通信）を標準で実装

・IPv4との互換性は無い（但し、機器にIPv4とIPv6の両方のアドレスを割り当てるデュアルスタックなどの仕組みで相互運用は可

IPv6ヘッダの説明として正しいもの

ホップリミットはIPv4のTTLと同じ役割を持つ

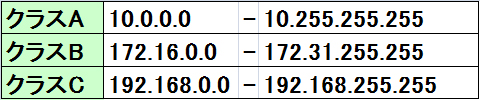
送信元アドレス、宛先アドレスは128ビットである

IPv4ヘッダ

「ヘッダ長」や「オプション」フィールドなど、いくつかのフィールドはIPv6ヘッダには存在しない

TTLはルータを1つ通過するごとに1ずつ減算される

IPアドレスのうちローカルネットワーク飲み使用が許可されているアドレスを「プライベートアドレス」という。クラスごとに範囲が決められている。



プライベートアドレスの範囲

10

172

192

192.168.10.128/28で使用可能なIPアドレスの数を書きなさい。

4ビットで表せる数は、2の4乗=16（0～15）となります。

このうち、ネットワークアドレスと、ブロードキャストアドレスを引いた16-2=14（個）が正解となります。

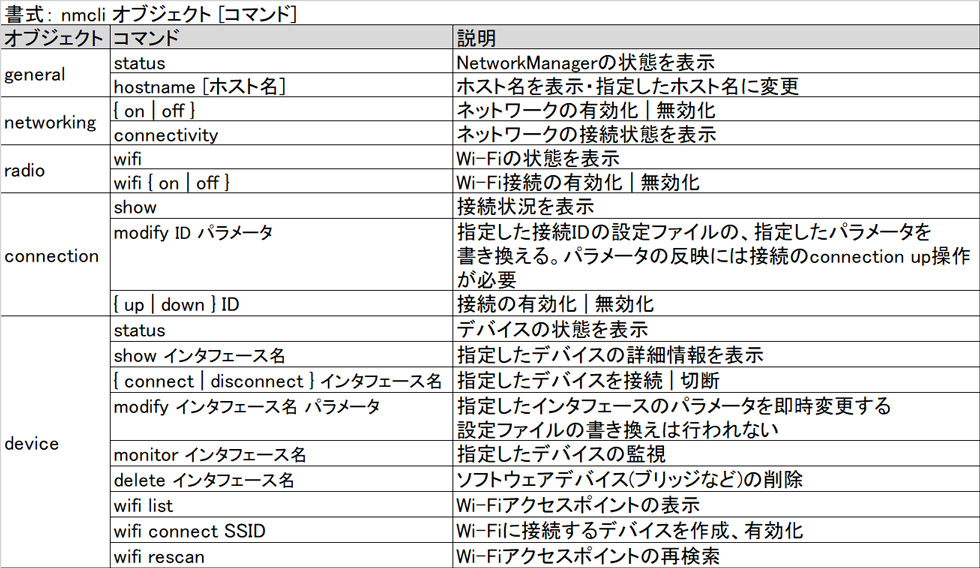
IPアドレスのクラス



・nmcliコマンド(Net WorkManagerの設定)

ネットワークの設定、接続の管理、状態の確認など

connection upもしくはdown



・netstatコマンド

ホストのネットワーク接続状態やソケット・インターフェイスごとのネットワーク統計などを表示するコマンド

「netstat」を使うと、実行するコンピュータが、どのコンピュータとどんなプロトコルを使って何番ポートで通信しているのかがわかる

-r ルーティングテーブルを表示

-i ネットワークインターフェースの状態を表示

-n アドレスやポートを数値で表示

・tracerouteコマンド

指定されたホスト（ホスト名もしくはIPアドレス）までパケットが伝わる経路を表示します。

・ifconfigコマンド

IPアドレスを確認するときによく使われるのはifconfigコマンドです。ifconfigコマンドは、ネットワークインターフェースの状態を表示したり、設定を行ったりします。

・ifup/ifdown

指定したインターフェースの有効と無効

・/etc/resolv.confファイルに関して

参照先DNSサーバは、/etc/resolv.confファイルに設定します。

問い合わせ先のDNSサーバのIPアドレスの指定！

「nameserver」の後にDNSサーバのIPアドレスを、「domain」の後に自ドメイン名を指定します。

# cat /etc/resolv.conf

nameserver 192.168.1.1

domain ping-t.com

・/etc/nsswitch.confファイル

名前解決をする手段はさまざまです。たとえばホスト名とIPアドレスの名前解決では、①/etc/hostsファイルを使う、②DNSサーバに問い合わせる、③LDAPサーバに問い合わせる、といった手段が考えられます。それらを、どういう順序で利用するか設定するファイルが/etc/nsswitch.confです。

getentコマンドでnsswitch.confに記載してあるようなことが見れる。

スーパーサーバには、inetdとxinetdがあります

「xinetd」は「inetd」より新しいスーパーサーバで、サービス毎に詳細な設定が可能

スーパーサーバinetdやxinetdはサービスのアクセス制御を行う際、TCPラッパーを利用します。TCPラッパーは「tcpd」というデーモン名で起動しています。他のデーモンさんがお休みできるように「お仕事してね」の要求が来ないか代表して見張っているデーモンさんのこと

特定のサービス宛のネットワーク接続を監視する「socketユニット」がスーパーサーバの代わりに使えます

xinetdのアクセス制御

・only\_fromもno\_accessも設定されていない場合、サービスへのアクセスは全て許可

・only\_fromだけが設定されている場合、指定されたアクセス元しかサービスへアクセス出来ない

・no\_accessだけが設定されている場合、指定されたアクセス元はサービスへアクセス出来ない

・only\_fromとno\_accessが同時に指定された場合、より厳密にマッチするほうに従う

xinetdの設定項目

bind/interface サービスを提供するインターフェースのIPアドレス

disable サービスを無効化するかの設定(yes/no)

instances サーバプログラムの最大起動プロセス数

log\_type ログの記録先指定(syslogや記録するログファイルの絶対パス)

no\_access サービスへのアクセスを拒否する接続元

only\_from サービスへのアクセスを許可する接続元

server サーバプログラムの絶対パス

server\_args サーバプログラム起動時に指定する引数

socket\_type ストリーム型、データグラム型といったサービスの接続タイプ

user サーバプログラムを起動するユーザー

wait サーバプログラムへの接続処理を待ち合わせるかの設定

waitに関して

・マルチスレッドのサーバプログラムの場合はnoを設定する

・シングルスレッドのサーバプログラムの場合はyesを設定する

・ホスト名を記述するファイル名

/etc/hostname

/etc/sysconfig/network

・fuserコマンド

システムの特定のリソースを使用中のプロセスを検索し、強制終了することのできるコマンド

-k:プロセスにSIGKILLを送信

-m:マウントされたファイルシステムを指定

-n:検索する領域を指定

-u:検索結果にユーザー名を追加

-v:詳細な情報を表示

アクセシビリティ

スティッキーキー 修飾キーを押されたままの状態にする

スローキー キー押下を認識する時間を調整、ビープ音を鳴らす等

バウンスキー 素早く何度も押したキーを認識しない

トグルキー NumLock、CapsLockなどの切り替え時にビープ音を鳴らす

・pingコマンドに関して

-c:送信回数

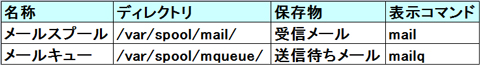
-i:送信間隔

-n:結果表示の形式をアドレスで表示(numeric)

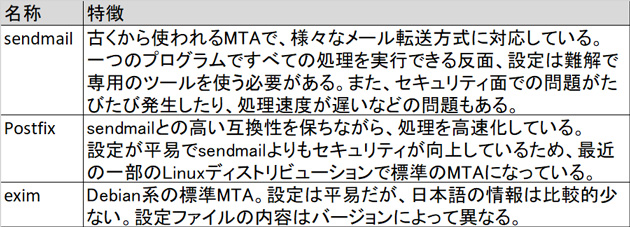
MTA(Mail Transfer Agent)は、通常SMTP（電子メール送信用のプロトコル）を利用して電子メールを送信します。

MTA間でメールの転送を行うことをリレー（中継）といい、リレー対象を制限せず、誰からのメール送信、転送依頼でも受け付ける設定になっているMTA（SMTPサーバ）の事を「オープンリレー」といいます。

オープンリレーは誰でも利用できるため、悪質な迷惑メール（スパム）送信に使用されてしまいます。そのためインターネットに公開されたSMTPサーバにはオープンリレー設定をするべきではありません。



代表的なメールサーバの特徴



/etc/aliasesファイルを利用すると、メールアドレスの別名（エイリアス）を設定できます。たとえば、root宛に届いたメールを、ユーザーadminとlpicでも受け取れるようにするには、/etc/aliasesで次のように設定します。

root:admin,lpic

メールを転送するもう一つの方法は.forwardファイルを使用する方法。

この場合は、各ユーザのホームディレクトリに.forwardファイルを用意し、その中に転送先のメールアドレスを記述します。一時的にメールを転送したい時に便利。ユーザが各自で設定することができ、管理者が設定する必要がない。

ネットワーク関連設定ファイル

etc/hostname ホスト名を設定(debian)(ホスト名のみ記述)

etc/sysconfig/network ホスト名を設定(Red hat)、ネットワーク機能の有効無効、デフォルトゲートウェイ等の設定

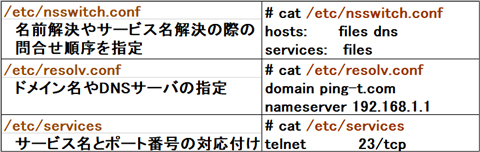
etc/hosts：IPアドレスとホスト名の対応付

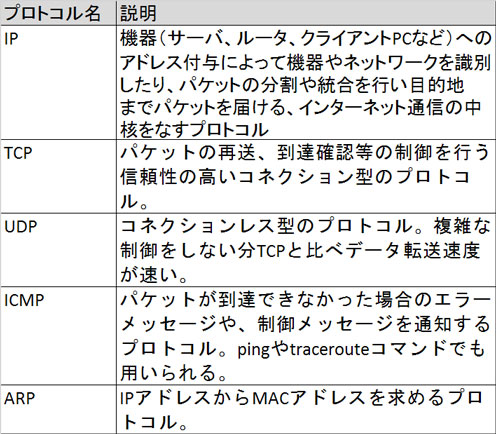
etc/nsswitch.conf：名前解決やサービス名解決の問い合わせ順序の設定

etc/resolv.conf：ドメイン名やDNSサーバの指定

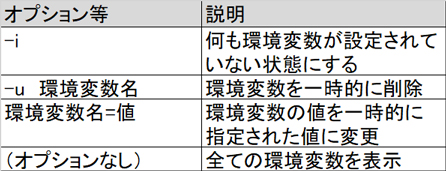
etc/services：サービス名とポート番号の対応つけ







・envコマンドに関して



コマンド実行時に何も環境変数が設定されていない状態にする

env -i

・whoコマンド

「/var/run/utmp」ファイルを参照し、現在システムにログインしているユーザーの情報を表示

管理者（＝ソケット）は、自分の島にも他の島にも存在するため、島の番号と港の番号（＝IPアドレスとポート番号）＋αの情報を組み合わせて特定されます。

ソケットを特定する情報の中にポート番号が含まれるため、ソケットの状況（IPアドレス＋ポート番号＋α）を確認すれば、ポートの状況が確認できる

nmapは外部のポートを調べるときに使用する

Lsofサーバー内のポートを確認

find [検索場所] -perm -[モード]

　[モード]で指定されている許可属性ビットのすべてが、ファイルでも立っていたら真

find [検索場所] -perm +[モード]

　[モード]で指定されている許可属性ビットのどれか一つが、ファイルでも立っていたら真

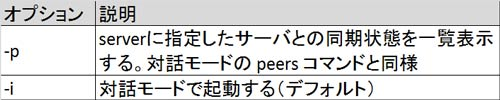
ルーティングテーブルに追加する方法

route add -net ネットワークアドレス gw ゲートウェイのIPアドレス

newaliases

/etc/aliasesの設定を有効にする

ntpdのコマンドに関して



デフォルトゲートウェイの設定の確認方法

デフォルトゲートウェイはルーティングテーブルに記載されています。

ルーティングテーブルは「netstat -r」や「route( -F)」、「ip route show」コマンドで表示できます。