

【UNIX】基礎

基礎編

■UNIX の概要

- ▶ ※ UNIX はサーバー用によく使われる。
- ▶ ※ UNIX は無料の互換OS がたくさん登場し広まった。
- ▶ ※ 互換OS の1つ Linux はいくつもの企業・団体によって独自の設定や形式で配布されたものがあり、それらの総称を Distribution という。
- ▶ ※ UNIX 系のOS（macOS 含む）はすべて POSIX という規格で統一されているため、基本的なコマンドは共通している。

■シェルのショートカットキー

- ▶ コマンドを実行せずに新しいプロンプトに移る
- ▶ 今書いているコマンドを消す
- ▶ 上に書いてきたのを消す
- ▶ コマンドやファ、ディ名を途中まで書いて補完
- ▶ 以前のコマンドを使う

■ディレクトリの構成

- ▶ ☆ 各ディレクトリの説明
- ▶ ☆ 絶対PATH と 相対PATH

■用語

- ▶ ストリーム
- ▶ ディスクリプタ

■基本操作

- ▶ カレントディを表示
- ▶ 他のディに移る
- ▶ 出力
- ▶ 現在のユーザ名を表示
- ▶ 文字列をトリムし出力

■コマンド実行全般

- ▶ コマの使い方を調べる

【UNIX】基礎

基礎編

■UNIX の概要

- ▶ ※ UNIX はサーバー用によく使われる。
- ▶ ※ UNIX は無料の互換OS がたくさん登場し広まった。
- ▶ ※ 互換OS の1つ Linux はいくつもの企業・団体によって独自の設定や形式で配布されたものがあり、それらの総称を Distribution という。
- ▶ ※ UNIX 系のOS（macOS 含む）はすべて POSIX という規格で統一されているため、基本的なコマンドは共通している。

■シェルのショートカットキー

- ▶ コマンドを実行せずに新しいプロンプトに移る {Ctrl}+{C}
- ▶ 今書いているコマンドを消す {Ctrl}+{U}
- ▶ 上に書いてきたのを消す {Ctrl}+{L} ※ `$ clear` でもいい
- ▶ コマンドやファ、ディ名を途中まで書いて補完 {Tab}
- ▶ 以前のコマンドを使う {↑} や {↓}

■ディレクトリの構成

- ▶ ☆ 各ディレクトリの説明
- ▶ ☆ 絶対PATH と 相対PATH

■用語

- ▶ ストリーム 何らかの入出力において、データ (バイト列) が流れる通り道
- ▶ ディスクリプタ 各プロセス内でストリームを区別する識別子。実体は整数値。

■基本操作

- ▶ カレントディを表示 \$ pwd ※ print working directory の略
- ▶ 他のディに移る \$ cd *dirPath* ※ change directory の略
- ▶ 出力 \$ echo '*str*'
- ▶ 現在のユーザ名を表示 \$ whoami か \$ echo \$USER
- ▶ 文字列をトリムし出力 \$ sed -r 's/^[[:space:]]*[[[:space:]]*\$//g' <<< \$hoge

■コマンド実行全般

- ▶ コマの使い方を調べる \$ man *com* か \$ help *com* か \$ *com* --help

- ▶ 管理者権限で実行
- ▶ 出力せずファを上書き
- ▶ 出力せずファに追記
- ▶ コマの引数にファを
- ▶ 出力せず別コマ引数に
- ▶ ※ コマンドは実行すると、終了ステータス（またはリターンコード）と呼ばれる数値を返す。成功の場合 `0`、失敗の場合 `1` またはその他の数値（`0` を除く）。なお、直前に実行したコマンドの終了ステータスの値は特殊変数 `$?` に格納されている。
- ▶ コマを立て続けに実行
- ▶ `"`（失敗したら次のコマは実行しない）
- ▶ `"`（失敗した場合のみ次のコマを実行）
- ▶ ※ `{0,0,0}` `{0..0}` でfor文のようにできる（ブレース展開）。
- ▶ 長いコマで改行したい
- ▶ ログを表示させない
- ▶ ※ オプションは複数付けられる。
- ▶ あるコマが出力するか

■コマンドの履歴を活用

- ▶ コマの履歴を見る
- ▶ 履歴上の n 番目を実行
- ▶ 直前のコマを実行
- ▶ n 個前のコマを実行
- ▶ 直前コマの最終引数
- ▶ `str`で始まる直近のコマ
- ▶ コマを検索

■ディレクトリの情報を確認

- ▶ ディの情報を確認

■ディレクトリの中身を確認

- ▶ ディの中身を確認
- ▶ 隠しファ含めて `"`
- ▶ タイプ識別子付きで `"`

- ▶ 管理者権限で実行 `$ sudo commands※` ※ `!` も使える。
- ▶ 出力せずファを上書き `$ commands > filePath` : リダイレクション
- ▶ 出力せずファに追記 `$ commands >> filePath` : `"`
- ▶ コマの引数にファを `$ commands < filePath` : `"`
- ▶ 出力せず別コマ引数に `$ commands | anotherCommands` : パイプ
- ▶ ※ コマンドは実行すると、終了ステータス（またはリターンコード）と呼ばれる数値を返す。成功の場合 `0`、失敗の場合 `1` またはその他の数値（`0` を除く）。なお、直前に実行したコマンドの終了ステータスの値は特殊変数 `$?` に格納されている。
- ▶ コマを立て続けに実行 `$ commands1 ; commands2 ; ...`
- ▶ `"`（失敗したら次のコマは実行しない） `$ commands1 && commands2 && ...`
- ▶ `"`（失敗した場合のみ次のコマを実行） `$ commands1 || commands2 || ...`
- ▶ ※ `{0,0,0}` `{0..0}` でfor文のようにできる（ブレース展開）。
- ▶ 長いコマで改行したい `\`を入力後 Enter キーで改行可能
- ▶ ログを表示させない `$ commands > /dev/null 2>&1` ※インストール時などに
- ▶ ※ オプションは複数付けられる。
- ▶ あるコマが出力するか `$ [-z $(commands)]; echo $?` ※ `1` は出力アリ `0` はナシ

■コマンドの履歴を活用

- ▶ コマの履歴を見る `$ history`
- ▶ 履歴上の n 番目を実行 `$!n`
- ▶ 直前のコマを実行 `$!!`
- ▶ n 個前のコマを実行 `$!-n`
- ▶ 直前コマの最終引数 `!$` で使える
- ▶ `str`で始まる直近のコマ `$!str` ※ `!str:p` とすれば実行はせず、その後 `!!` で実行
- ▶ コマを検索 `{Ctrl}+{R} → str入力 → {Ctrl}+{R}`で次候補へ `→ {Enter}`か`{Ctrl}+{C}`

■ディレクトリの情報を確認

- ▶ ディの情報を確認 `$ ls -ld dirPath`

■ディレクトリの中身を確認

- ▶ ディの中身を確認 `$ ls dirPath` : list ※`dirPath`省略ならカレディ
- ▶ 隠しファ含めて `"` `$ ls -a dirPath`
- ▶ タイプ識別子付きで `"` `$ ls -F dirPath` ※ `*` `/` `=` `>` `@` `|` のどれかが末尾につく

▶ 1件1行で "	
▶ 権限の情報付きで "	
▶ 再帰的に "	

■ファイルやディレクトリの操作

▶ ファの更新日時を更新	
▶ ファを新規作成	
▶ ディを新規作成	
▶ リンクを作成	
▶ ファを複製	
▶ ディを複製	
▶ 移動	
▶ 名前の変更	
▶ ファを削除	
▶ ディの中身を空に	
▶ 中身が空のディを削除	
▶ 中身ごとディを削除	

■ファイルの情報を確認

▶ 権限を確認	
▶ リンクのリンク先	

■ファイルの中身を確認

▶ 一気にすべて確認	
▶ ちょっとずつ確認 (ページャ)	
▶ 行数や単語数を確認	
▶ 先頭・末尾の幾行か	
▶ 随時追記されるたびに	
▶ <i>n</i> 行目のみ	
▶ 検索して該当行を抽出	
▶ 文字列を置換して出力	

■ファイルを編集 (viエディター)

▶ 1件1行で "	\$ ls -l <i>dirPath</i>
▶ 権限の情報付きで "	\$ ls -l <i>dirPath</i>
▶ 再帰的に "	\$ ls -R <i>dirPath</i>

■ファイルやディレクトリの操作

▶ ファの更新日時を更新	\$ touch <i>filePath</i>	※ファイルがない場合自動作成
▶ ファを新規作成	\$ touch <i>filePath</i>	※深いファならその祖先ディは既存の必要
▶ ディを新規作成	\$ mkdir <i>childDirName</i>	\$ mkdir -p <i>dirPath</i>
▶ リンクを作成	\$ ln -s <i>dirPath name</i>	※シンボリックリンクという
▶ ファを複製	\$ cp <i>filePath newFilePath</i>	
▶ ディを複製	\$ cp -r <i>dirPath newDirPath</i>	
▶ 移動	\$ mv <i>path destinedDirPath</i>	
▶ 名前の変更	\$ mv <i>path newPath</i>	
▶ ファを削除	\$ rm <i>filePath</i>	
▶ ディの中身を空に	\$ rm -r <i>dirPath/*</i>	※とても危険なので要注意
▶ 中身が空のディを削除	\$ rmdir <i>dirPath</i>	
▶ 中身ごとディを削除	\$ rm -r <i>dirPath</i>	※とても危険なので要注意

■ファイルの情報を確認

▶ 権限を確認	\$ ls -l <i>filePath</i>
▶ リンクのリンク先	\$ readlink <i>filePath</i>

■ファイルの中身を確認

▶ 一気にすべて確認	\$ cat <i>filePath</i>	
▶ ちょっとずつ確認 (ページャ)	・ \$ more <i>filePath</i> ・ \$ less <i>filePath</i>	※{Space} で次頁へ、{Q} で終了。 ※{1}{1}で行スクロール、{Q} で終了。
▶ 行数や単語数を確認	\$ wc <i>filePath</i>	
▶ 先頭・末尾の幾行か	\$ head -n <i>lineCount filePath</i>	・ \$ tail -n <i>lineCount filePath</i>
▶ 随時追記されるたびに	\$ tail -f <i>filePath</i>	※追記されるたびに追加で表示する
▶ <i>n</i> 行目のみ	\$ sed -n <i>np filePath</i>	か \$ cat <i>filePath</i> head - <i>n</i> tail -1
▶ 検索して該当行を抽出	\$ grep ' <i>str</i> ' <i>filePath</i>	
▶ 文字列を置換して出力	\$ sed s/ <i>what/replacement/g filePath</i>	※見つければ全て置換

■ファイルを編集 (viエディター)

▶ ※ ファイルを上書きや追記するくらいなら、viエディターを使わずリダイレクションを使えばいい。

▶ 起動

▶ ※ 起動直後はコマンドモードで、左下に `-` と出ている。

▶ 編集モードに

▶ コマンドモードに戻る

▶ 終了 (`''`)

▶ 保存して終了 (`''`)

▶ 保存せずに終了 (`''`)

■ファイル実行

▶ ファイル実行

▶ ファ名だけで実行したいなら

▶ コマのファの場所

■ファイルやディレクトリの検索

▶ ファヤディの検索

▶ ファだけ検索

▶ ディだけ検索

■ユーザやグループの管理

▶ 全ユーザを表示

▶ 全グループを表示

▶ ヌの属するグを確認

▶ ユーザを新規作成

▶ ユーザを削除

▶ ヌのパスワード変更

▶ グループを新規作成

▶ グループを削除

▶ ヌをグに追加

▶ ヌをグから排除

▶ ☆ ユーザのデフォルトシェル（ログインシェル）を変更

▶ ※ ファイルを上書きや追記するくらいなら、viエディターを使わずリダイレクションを使えばいい。

▶ 起動 `$ vi filePath※` ※未存でも可

▶ ※ 起動直後はコマンドモードで、左下に `-` と出ている。

▶ 編集モードに `{|}` ※左下に `I` と出る

▶ コマンドモードに戻る `{Esc}`

▶ 終了 (`''`) `:q`

▶ 保存して終了 (`''`) `:wq`

▶ 保存せずに終了 (`''`) `:q!`

■ファイル実行

▶ ファイル実行 `・ $ filePath※` ※カレディ中のファは `./` に続ける必要あり
`・ $ $SHELL filePath※` ※ `./` に続ける必要なし

▶ ファ名だけで実行したいなら `$ export PATH=dirAbsPath:$PATH` でパスを通しておく

▶ コマのファの場所 `$ which commandName`

■ファイルやディレクトリの検索

▶ ファヤディの検索 `$ find dirPath -name 'str※'` ※ワイルドカード可

▶ ファだけ検索 `$ find dirPath -name 'str※' type -f` ※ `''`

▶ ディだけ検索 `$ find dirPath -name 'str※' type -d` ※ `''`

■ユーザやグループの管理

▶ 全ユーザを表示 `$ cat /etc/passwd`

▶ 全グループを表示 `$ cat /etc/group`

▶ ヌの属するグを確認 `$ groups user1 ...`

▶ ユーザを新規作成 `$ sudo useradd userName` ※ `-m` でホームディも自動作成

▶ ユーザを削除 `$ sudo userdel user` ※ `-r` でホームディも削除

▶ ヌのパスワード変更 現ユーザなら `$ passwd` 他人なら `$ sudo passwd user`

▶ グループを新規作成 `$ sudo groupadd groupName`

▶ グループを削除 `$ sudo groupdel group`

▶ ヌをグに追加 `$ gpasswd -a user group` か `$ usermod -aG group user`

▶ ヌをグから排除 `$ gpasswd -d user group`

▶ ☆ ユーザのデフォルトシェル（ログインシェル）を変更

■アクセス権限（パーミッション）

▶ ファヤディのパーミ、所有者などを確認	
▶ ファヤディのパーミを変更	
▶ ファヤディの所有者を変更	

■システム

▶ 環境変数の一覧	
▶ 環境変数の編集	
▶ 環境変数の値を出力	
▶ 再起動	
▶ 実行中プロセス一覧	

■ネットワーク

▶ HTTPアクセスをしてコンテンツを取得	
▶ リダイレクトも処理して "	
▶ 自己署名証明書を受け入れて "	
▶ オンラインファイルをダウンロード	
▶ グローバルIPアドレス	
▶ プライベートIPアドレス	
▶ 別ホストまでのネットワーク経路	
▶ 別ホストにパケットを飛ばす	
▶ ホスト名	

■別サーバにSSH接続

鍵作成

▶ 公開鍵／秘密鍵のペアを作る	
-----------------	--

普通のSSH接続

▶ サーバに接続	
▶ ログアウト (サーバと切断)	

サーバとのファイルの授受 (SFTP)

■アクセス権限（パーミッション）

▶ ファヤディのパーミ、所有者などを確認	ファ : \$ ls -l <i>filePath</i> ディ : \$ ls -ld <i>dirPath</i> ※ <i>dirPath</i> 省略ならカレディ
▶ ファヤディのパーミを変更	ファ : \$ chmod <i>mode filePath</i> ディ : \$ chmod -R <i>mode dirPath</i>
▶ ファヤディの所有者を変更	ファ : \$ sudo chown <i>user:group filePath</i> ※ . じゃなく : ディ : \$ sudo chown -R <i>user:group dirPath</i> ※ "

■システム

▶ 環境変数の一覧	\$ env
▶ 環境変数の編集	\$ export 環境変数名=値
▶ 環境変数の値を出力	\$ echo \$環境変数名
▶ 再起動	\$ sudo reboot
▶ 実行中プロセス一覧	\$ ps -x

■ネットワーク

▶ HTTPアクセスをしてコンテンツを取得	\$ curl <i>URL</i>
▶ リダイレクトも処理して "	\$ curl -L <i>URL</i> ※ \$ wget <i>URL</i> と同じっぽい
▶ 自己署名証明書を受け入れて "	\$ curl -k <i>URL</i> ※ --insecure でも可。
▶ オンラインファイルをダウンロード	\$ curl -L -o <i>fileName URL</i>
▶ グローバルIPアドレス	\$ curl inet-ip.info など
▶ プライベートIPアドレス	\$ ifconfig
▶ 別ホストまでのネットワーク経路	\$ sudo traceroute <i>targetIP</i> か \$ tracepath <i>targetIP</i>
▶ 別ホストにパケットを飛ばす	\$ ping <i>targetIP</i>
▶ ホスト名	\$ hostname

■別サーバにSSH接続

鍵作成

▶ 公開鍵／秘密鍵のペアを作る	\$ ssh-keygen -t 暗号化方式 -b ビット数 -C "コメント"
-----------------	--

普通のSSH接続

▶ サーバに接続	\$ ssh <i>serverUser@serverIP</i>
▶ ログアウト (サーバと切断)	exit (か {Ctrl} + {D} ?)

サーバとのファイルの授受 (SFTP)

▶ SFTPを起動	
▶ ファをD L・	
ディをD L	
▶ ファをU P・	
ディをU P	
▶ SFTPを終了	

サーバとのファイルの授受 (SCP)

▶ ファをD L・	
ディをD L	
▶ ファをU P・	
ディをU P	

■日時

▶ 日時	
▶ 今月のカレンダー	

▶ SFTPを起動	\$ sftp <i>remoteUser@remoteIP</i>
▶ ファをD L・	\$ get <i>filePath</i> ※ ¹ ・
ディをD L	\$ get -r <i>dirPath</i> ※ ¹ ※ ¹ 相手サーバ上のパス
▶ ファをU P・	\$ put <i>filePath</i> ※ ¹ ・
ディをU P	\$ put -r <i>dirPath</i> ※ ¹ ※ ¹ 自身のWindows上のパス
▶ SFTPを終了	\$ exit quit bye {Ctrl} + {C} のいずれか

サーバとのファイルの授受 (SCP)

▶ ファをD L・	\$ scp <i>remoteUser@remoteIP:remoteSrcFilePath localDstPath</i> ・
ディをD L	\$ scp -r <i>remoteUser@remoteIP:remoteSrcDirPath localDstPath</i>
▶ ファをU P・	\$ scp <i>localSrcFilePath remoteUser@remoteIP:remoteDstPath</i> ・
ディをU P	\$ scp -r <i>localSrcDirPath remoteUser@remoteIP:remoteDstPath</i>

■日時

▶ 日時	\$ date
▶ 今月のカレンダー	\$ cal