

LinuC レベル1 技術解説セミナー
2023/2/19 開催

テキストエディタ

主題	1.03 : GNUとUnixのコマンド
副題	1.03.5 : エディタを使った基本的なファイル編集の実行

三澤 康巨

三澤 康巨

- KDDI株式会社で、サービス設備のエンジニアリングをはじめ様々な業務を担当。
- 2017年10月からKDDIグループ内のサーバ研修講師。200名超の LinuC レベル1 合格者。
- 2020年3月、KDDIを定年退職。
- LinuCレベル1技術解説セミナーの講師を担当
 - 2020年7月18日 [「ブートプロセスとsystemd」](#)
 - 2021年1月23日 [「ハードディスクのレイアウトとパーティション」](#)
 - 2021年3月 6日 [「ファイルシステムの作成と管理、マウント」](#)
 - 2021年6月26日 [「テキストデータ処理」](#)
 - 2022年1月15日 [「セキュリティ管理」](#)
 - 2022年3月12日 [「暗号の利用」](#)
 - 2022年7月30日 [「リンク、ファイル配置」](#)
- その他
 - 2020年11月28日、[オープンソースカンファレンス2020 オンライン/福岡「Linuxマシンを作ってみよう ～LinuC レベル1 /レベル2 学習環境構築ガイド～」](#)の講師を担当

■LinuCとは

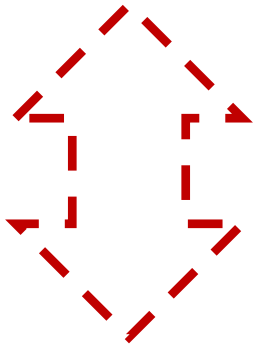
クラウド時代の即戦力エンジニアであることを証明するLinux技術者認定

- ✓現場で「今」求められている新しい技術要素に対応
 - オンプレミス／仮想化・コンテナを問わず様々な環境下でのサーバー構築
 - 他社とのコラボレーションの前提となるオープンソースへの理解
 - システムの多様化に対応できるアーキテクチャへの知見
- ✓全面的に見直した「今」身につけておくべき技術範囲を網羅

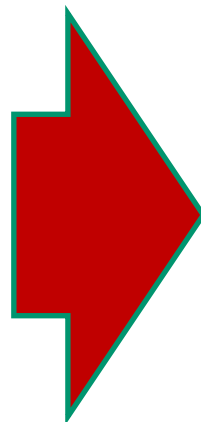
今となっては使わない技術やコマンドの削除、アップデート、新領域の取り込み
- ✓Linuxの範疇だけにとどまらない領域までカバー

セキュリティや監視など、ITエンジニアであれば必須の領域もカバー

AWSなどの
パブリッククラウドを
活用するための技術



間が
欠けて
いる状態



AWSなどの
パブリッククラウドを
活用するための技術

仮想マシン/コンテナ技術、
クラウドセキュリティ、
アーキテクチャ、ほか

オンプレミスの
サーバーサイドLinux技術

オンプレミスの
サーバーサイドLinux技術

【今まで/その他】



テキストエディタ

- 1.はじめに
- 2.viエディタの起動とモード
- 3.コマンドモード、入力モード
- 4.ラストラインモード、viの終了
- 5.エディタの選択

1.はじめに

2.viエディタの起動とモード

3.コマンドモード、入力モード

4.ラストラインモード、viの終了

5.エディタの選択

- 本セミナーでは、副題1.03.5のテキストエディタを学習します。viエディタの基本的な使い方と、デフォルトのエディタを設定する方法について解説します。
- 学習効果を高めるため、実行例の出てくる部分では、ご自分でも実行してみることをお勧めします。
- Linuxには多数のディストリビューションが存在しますが、本セミナーの実行例では、CentOS 7 を使用します。
 - ビジネス用サーバーの多くで稼働している Red Hat Enterprise Linux 7 (RHEL7) と互換性があります。
 - RHEL7は有料ですが、CentOS 7 は無料で利用できます。
 - CentOSのバージョンは「Stream 9」までありますが、本セミナーでは、安定版であるバージョン「7」を使います。

- CentOS 7 に基づく学習環境の構築方法を、LPI-Japanのサイトでご紹介しています。

LinuC レベル1 / レベル2 Version 10.0 学習環境構築ガイド

https://linuc.org/docs/v10/guide_text.pdf

- 学習環境構築ガイドでは、2種類の環境の構築方法を紹介しています。

【環境A】

- 用意したコンピュータの内蔵ストレージを上書きして、Linux専用コンピュータを構築します。
- WindowsやMacOS等の既存OSは使えなくなります。
- 不要になった古いPC等がある場合に、それを使ってください。

【環境B】

- WindowsやMacOS等の既存OSを壊すことなく、外付けSSDにLinuxをインストールします。
- これによって、既存OSとLinuxとの間を切り替えて利用することができます。
- 但し、既存OSとLinuxとを同時に利用することはできません。

※他に、VirtualBox等も紹介しています。

1.はじめに

2.viエディタの起動とモード

3.コマンドモード、入力モード

4.ラストラインモード、viの終了

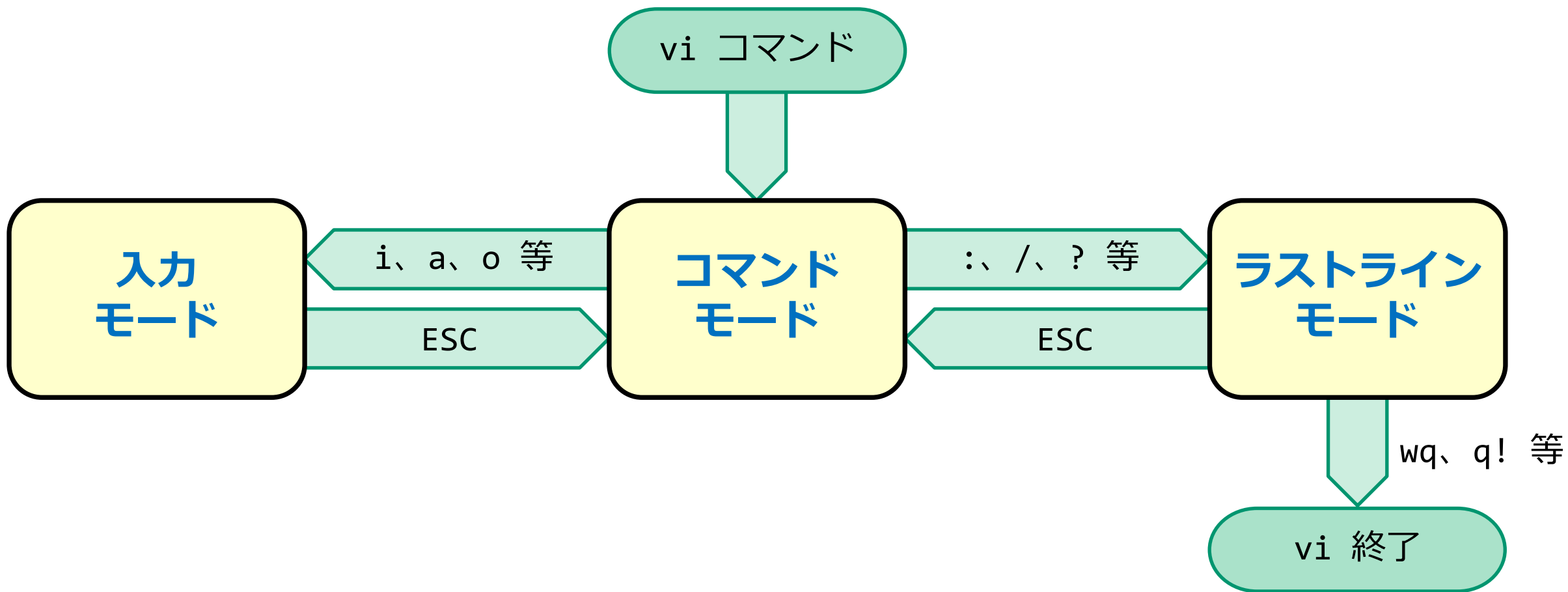
5.エディタの選択

- 設定ファイルの変更やシェルスクリプトの作成など、テキストファイルを編集することがあります。graphical.targetであればGUI環境でマウスを使うエディタも利用可能ですが、リモートアクセス環境などでは利用できないことが少なくありません。
- UNIXには古くからviエディタが標準で組み込まれていました。Linuxの多くのディストリビューションには、viの改良版であるVimが標準搭載されています。
- viエディタ (Vim) を起動するには、viコマンドを使用します。

vi コマンド

vi [オプション] ファイル名

- viエディタには3つのモードがあり、切り換えながら編集します。
- vi起動時はコマンドモードになり、入力・編集後、ラストラインモードから終了します。



1.はじめに

2.viエディタの起動とモード

3.コマンドモード、入力モード

4.ラストラインモード、viの終了

5.エディタの選択

- 矢印キーでカーソルを移動することができますが、矢印キーのない場合、コマンドモードでは h, j, k, l コマンドで移動します。
- コマンドの前に数字を添えると、その回数だけコマンドを繰り返します。(5l → 右に5文字移動)

コマンド	説明
h	左に移動
j	下に移動
k	上に移動
l	右に移動



コマンド	説明
0	行頭へ移動
\$	行末へ移動

- カーソルを行間で移動するには、Gコマンド等を使えます。

コマンド	説明
gg	ファイルの先頭行に移動
G	ファイルの最終行に移動
nG	n行目に移動

- 入力モードへ切り替えるには、i コマンド等を使います。

コマンド	説明
i	カーソル位置に挿入
a	カーソル位置の後ろに挿入
o	カーソル位置の下に行を挿入

コマンド	説明
I	行頭に挿入
A	行末に挿入
O	カーソル位置の上に行を挿入

- 入力モードでは、キーボードで普通に文字を入力します。
- コマンドモードへ戻るには、ESCキーを押します。

vi起動、モード切り換え、カーソル移動、終了

```

$ vi /etc/services
:set nu      ※行番号を表示
G           ※最終行へ移動
gg          ※先頭行へ移動
9G         ※9行目へ移動
i           ※入力モードへ切り換え
iiii       ※文字を入力
ESC        ※コマンドモードへ復帰
$          ※行末へ移動
5h         ※5文字左に移動
0          ※行頭へ移動
7l         ※7文字右に移動
3j         ※3行下に移動
9k         ※9行上に移動
:         ※ラストラインモードへ切り替え
q!        ※vi終了（編集結果を破棄）

$

```

文字の入力

```

$ ps ax > viex.txt
$ vi viex.txt
10G
i                ※入力モードへ切り換え
iiii
ESC              ※コマンドモードへ復帰
$
a                ※入力モードへ切り換え
aaaaa
ESC              ※コマンドモードへ復帰
o                ※入力モードへ切り換え
0000000000
ESC              ※コマンドモードへ復帰

```

文字の入力

```

I                ※入力モードへ切り換え
IIIII
ESC              ※コマンドモードへ復帰
A                ※入力モードへ切り換え
AAAAA
ESC              ※コマンドモードへ復帰
O                ※入力モードへ切り換え
0000000000
ESC              ※コマンドモードへ復帰

```


- テキストの削除には、xコマンド等を使います。

コマンド	説明
x	1文字削除
r	1文字置換
cc	1行置換（削除して入力モードへ）
cw	1語置換（削除して入力モードへ）
dd	1行削除
dw	1語削除（カーソルから右側を削除）
u	直前のコマンドを取消す

テキストの削除・置換、コマンドの取消し

(続き)

x	※ 1 文字削除
r	※ 1 文字置換
r	
gg	
cc	※ 1 行置換
cc	
ESC	
j	
\$	
9h	
cw	※ 1 語置換
CW	
ESC	
5dd	※ 5 行削除
\$	
6h	
dw	※ 1 語削除
u	※ コマンドの取消し

- コマンドモードでコピー、ペーストを行えます。
- dd, dwで削除した文字列もペーストできます。

コマンド	説明
yy	1行コピー
yw	1語コピー（カーソルから右側をコピー）

コマンド	説明
p	後ろにペースト
P	前にペースト

コピー、ペースト

(続き)

10j

5yy

※5行コピー

3G

p

※後ろにペースト

gg

P

※前にペースト

25l

yw

※1語コピー

\$

p

※後ろにペースト

3dd

※3行削除

3j

p

※後ろにペースト

- 1.はじめに
- 2.viエディタの起動とモード
- 3.コマンドモード、入力モード
- 4.ラストラインモード、viの終了**
- 5.エディタの選択

- 文字列を検索するには、/ 等でラストラインモードへ切り替えてから、検索する文字列を入力します。
- n等で次の検索対象へ移動します。

コマンド	説明
/文字列	下方向に検索
?文字列	上方向に検索

コマンド	説明
n	順方向へ検索
N	逆方向へ検索
:noh	ハイライトを削除

文字列の検索

(続き)

```

/set          ※「set」を検索 (下方向)
n (数回押下) ※順方向へ検索
N (数回押下) ※逆方向へ検索
G
?usr         ※「usr」を検索 (上方向)
n (数回押下) ※順方向へ検索
N (数回押下) ※逆方向へ検索
:wq          ※vi終了 (編集結果を保存)

$ vi viex.txt
:noh        ※ハイライトを削除

```

- 文字列をまとめて置換するには、:でラストラインモードへ切り替えてから、sコマンドを使用します。
- 検索文字列には正規表現を使用できます。

コマンド	説明
<code>:%s/検索文字列/置換文字列/</code>	検索文字列を置換文字列に置換する（行内の最初の検索文字列のみ）
<code>:%s/検索文字列/置換文字列/g</code>	検索文字列を置換文字列に置換する（行内の全ての検索文字列）

文字列の置換

(続き)

G

?usr

:%s/usr/xyz/

※ 「usr」を「xyz」に置換する (行内最初の「usr」のみ)

u

G

:%s/usr/xyz/g

※ 「usr」を「xyz」に置換する (行内全ての「usr」)

- viを終了するには、: でラストラインモードへ切り替えてから、変更の保存または破棄コマンドを入力します。コマンドモードから保存終了することも可能です。

コマンド	説明
:wq	編集結果を保存して終了
ZZ	編集結果を保存して終了（コマンドモードから直接終了）
:q!	編集結果を破棄して終了

- viを終了する以外のラストラインモード操作を以下に示します。

コマンド	説明
:w [ファイル名]	viを終了せずに編集結果を保存
:e!	編集結果を破棄して最後に保存した内容に戻る
:!コマンド	シェルコマンドを実行

viの終了、変更の保存・破棄

(続き)

```
ZZ          ※vi終了 (編集結果を保存)

$ vi viex.txt
:%s/xyz/usr/g
:!!ls      ※lsコマンドを実行
ENTER     ※viエディタを続ける
:w viex2.txt ※編集結果を別ファイルに保存
:e!       ※編集結果を破棄
:wq       ※vi終了 (編集結果を保存)

$ ls
$ cat viex2.txt
```

- vi編集画面の設定をラストラインモードで変更できます。

コマンド	説明
<code>:set nu</code>	行番号を表示
<code>:set nonu</code>	行番号を非表示
<code>:set ts=文字数</code>	タブの表示文字数を設定

- ラストラインモードで変更した画面設定はviを終了すると失われます。
- Vimの設定ファイル `~/.vimrc` に設定を記述しておくと、Vim起動時に読み込まれて、自動的に設定されます。

viの設定

```
$ vi viex2.txt
:set nu          ※行番号を表示
i
TAB
ESC
:set ts=4       ※タブ表示を4文字に設定
:wq

$ vi viex2.txt
:q!
```

viの設定

```
$ vi ~/.vimrc
i
set nu
set ts=4
ESC
:wq

$ vi viex2.txt
:set nonu       ※行番号を非表示
:q!
```

1.はじめに

2.viエディタの起動とモード

3.コマンドモード、入力モード

4.ラストラインモード、viの終了

5.エディタの選択

- Linuxで利用できるテキストエディタには、vi (Vim) 以外にも、古くからある emacs や比較的に新しい nano があります。

emacs と nano

```
$ su -
パスワード:
# yum install emacs
# exit

$ emacs
CTRL+x
CTRL+c

$ nano
CTRL+x
```

- Red Hat系ディストリビューションではデフォルトのエディタが Vim、Debian系ディストリビューションでは nano になっています。
- デフォルトを変更するには、環境変数 EDITOR に設定します。

デフォルトのエディタの変更

```

$ crontab -e
    ※viが起動
:q!
$ echo $EDITOR

$ export EDITOR=nano           ※デフォルトをnanoに変更
$ echo $EDITOR
nano
$ crontab -e
    ※nanoが起動
CTRL+x
$ unset EDITOR
$ echo $EDITOR

$

```


1. CentOS 7 に基づく学習環境を自分で構築してみましょう。
2. viエディタでは、3つのモードを切り換えながら編集します。
3. コマンドモードでは h, j, k, l 等でカーソルを移動し、i, a, o 等で入力モードへ切り替えて入力後、ESCでコマンドモードへ復帰します。dd, cc 等で文字の削除や置換を行います。yy, p 等でコピー、ペーストを行います。
4. ラストラインモードでは、/ 等で検索、sコマンドで一括置換を行えます。:wq, :q! 等でviを終了します。
5. vi (Vim) 以外に emacs や nano を利用できます。デフォルトのエディタを変更するには、環境変数 EDITOR に設定します。

ご清聴ありがとうございました