

## eラーニングを普及させるための Moodle インストール研修の実践

梅崎 卓哉<sup>1</sup>、穂屋下 茂<sup>1,2</sup>

### The Practice of the Moodle Installation Workshop to Spread e-Learning

Takuya UMEZAKI<sup>1</sup>, Shigeru HOYASHITA<sup>1,2</sup>

#### 要 旨

eラーニングを実施する上で、LMSは重要な役割を果たす。最近、サーバはクラウドサーバを利用する傾向にある。そこで、クラウド基盤ソフトウェアを利用した仮想サーバ環境を使い、LAMP環境におけるWEBアプリケーションシステム（Moodleを利用）の構築を行う研修を実施した。Moodleは、Modular Object-Oriented Dynamic Learning Environmentの頭字語で、オープンソースでもあるため、世界的に最もよく利用されているLMSである。このMoodleの最新バージョンはスマートフォンにも対応し、英語学習などでは、ちょっとした隙間時間でも手軽に利用できるようになってきた。これからMoodleを利用する大学等は増えることが予想される。本稿では、大学教育改革にICT活用教育を導入してみたい大学、導入しているが内容をもっと知って効率よく使いたい教職員をターゲットにして行った研修実施内容について紹介する。

【キーワード】 eラーニング、LMS、Moodle、インストール、サーバ

#### 1. はじめに

最近、アクティブ・ラーニングが注目され、さらに小中高校の学習指導要領の改定案では「主体的・対話的で深い学び」と言い換えされている。これらを実現するためには、LMS等のICTツールを活用すれば、カリスマ教師でなくても誰でも比較的容易に実施できることを明らかにしてきた。

LMS（学習管理システム：Learning Management System）とは、eラーニングの実施に必要な、学習教材の配信や成績などを統合して管理するシステムであり、eラーニングのOS（オペレーティングシステム）とも言える重要な存在である。その種類は様々ある。その中でオープンソースのMoodleは、最もよく利用されている。このMoodleも最新バージョンではスマートフォンに対応し、英語学習などでは、ちょっとした隙間時間でも手軽に利用で

---

<sup>1</sup> 全学教育機構

<sup>2</sup> 責任著者

きるようになってきた。自宅や大学等でしっかり学習する時は PC から Moodle を利用する機会が多いと思われるが、スマートフォンから利用した進捗状況も同時に表示されるので使い勝手は非常に良い。今後、Moodle を使う大学は増えることが期待される。そうすると、大学で管理運営する必要があるが出てくる。サーバを構築し、運営継続することは、セキュリティへの対応も含めて深刻になっており、サーバを利用するならば、クラウドサーバが主流になることも考えられる。

ところで、佐賀大学は2016年7月29日に文部科学省の教育関係共同利用拠点（ICT活用教育）に認定された。目的の一つは、佐賀大学で培ってきた ICT 活用教育を活用して、教育改善を促すための FD/SD 研修プログラムを開発することである。

本拠点では、いろいろな効果的と思われる LMS やアクティブ・ラーニングの研修を試行しながら、今後少しでも効果的な FD/SD 研修プログラムの開発を進めようとしている。その中の一つとして、Moodle インストール研修を試みた。クラウド基盤ソフトウェアを利用した仮想サーバ環境を使い、LAMP（Linux/Apache/MySQL/PHP の各先頭文字4つを並べた表記）環境における WEB アプリケーションシステム（Moodle を利用）の構築を行う研修として開催した。その内容等を報告する。

## 2. 研修のスケジュール

本研修のスケジュールを表1に示す。1日目5時間、2日目5時間、合計10時間で実施した。研修内容を説明した後、実際に持込 PC で用意した無線 LAN に接続できることを確認し、OpenStack/Dashboard 画面を使い、2つの仮想サーバ（WEB、DB）を構築する<sup>1)</sup>。これら仮想サーバに SSH 接続し、Moodle をインストール、設定し、初期コースの作成、コースのバックアップやリストア等までを行う。サーバ構築では、表2に示すような Linux コマンドの知識が必要であり、さらにインストールした設定ファイルの一部を修正するために vi エディタを利用するので表3に示すような vi のコマンド（ファイルの作成、保存、中止、テキスト挿入、置換、コピー、検索など）の知識を必要とする<sup>2)</sup>。この vi (visual display editor) は Linux 環境では標準的なテキストエディタである。

表1 研修のスケジュール

- 1日目(5時間)
  - ・研修内容を説明
  - ・持込PCから無線LANに接続
  - ・OpenStack/Dashboard画面にて、
    - 2仮想サーバ(web/db)の生成
  - ・SSH接続にて2仮想サーバの構築
  - ・Moodleをインストール、初期セットアップ
- 2日目(5時間)
  - ・持込PCから無線LANに接続
  - ・Moodleサイトにログイン、管理画面でのサイト基本設定
  - ・ユーザ作成・登録およびコースの作成
  - ・Moodleプラグインの追加等
  - ・コースのバックアップやリストア等

表2 ターミナル等での Linux コマンド一覧

| コマンド名        | 説明   | 使用例                              |
|--------------|--|----------------------------------|
| pwd          | print name of current/working directory                            | pwd                              |
| ls           | list directory contents  | ls -la                           |
| cd           | change the working directory                                       | cd dir1                          |
| cp           | copy files and directories   | cp file1 /dir1/                  |
| mv           | move (rename) files  | mv old-file1 new-file1           |
| mkdir        | make directories   | mkdir dir1                       |
| rmdir        | remove empty directories   | rmdir dir1                       |
| rm           | remove files or directories  | rm dir1 -rf                      |
| chmod        | change file mode bits  | chmod 755 file1                  |
| chown        | change file owner and group  | chown apache:apache dir1 -R      |
| chcon        | change file SELinux security context                               | chcon reference=../ file1        |
| df           | report file system disk space usage                                | df                               |
| hostname     | show or set the system's host name                                 | hostname                         |
| uname        | print system information   | uname                            |
| ip           | show / manipulate routing, devices, policy routing and tunnels     | ip a                             |
| lsblk        | list block devices   | lsblk                            |
| sudo         | execute a command as another user                                  | sudo vi file1                    |
| fdisk        | manipulate disk partition table                                    | fdisk -l                         |
| getenforce   | get the current mode of SELinux                                    | getenforce                       |
| (setenforce) | modify the mode SELinux is running in                              | setenforce 1                     |
| yum          | Yellowdog Updater Modified   | yum -y update                    |
| reboot       | halt, poweroff, reboot - Halt, power-off or reboot the machine     | reboot                           |
| systemctl    | Control the system system and service manager                      | systemctl start httpd24-httpd    |
| scl          | Setup and run software from Software Collection environment        | scl enable rh-mysql57 bash       |
| rpm          | RPM Package Manager  | rpm -qa                          |
| scp          | secure copy (remote file copy program)                             | scp -i ./xxx_key.pem ... (履歴参照)  |
| tar          | manual page for tar 1.26   | tar zxvf moodle-latest-34.tgz    |
| man          | an interface to the on-line reference manuals                      | man command1                     |
| mkfs.xfs     | construct an XFS filesystem  | mkfs.xfs -L mydata /dev/vdb1     |
| tune2fs      | a just tunable filesystem parameters on ext2/ext3/ext4 filesystems | tune2fs -c 0 -i 0 -r 0 /dev/vdb1 |
| mkfs.ext4    | create an ext2/ext3/ext4 filesystem                                | mkfs.ext4 -L mydata /dev/vdb1    |
| mount        | mount a filesystem   | mount LABEL=mydata /mydata       |
| (umount)     | unmount file systems   | umount /mydata                   |
| getsebool    | get SELinux boolean value(s)                                       | getsebool -a                     |
| setsebool    | set SELinux boolean value  | setsebool httpd_unified 1 -P     |

表3 vi エディタのコマンド

| コマンド名          | 説明                     |
|----------------|------------------------|
| vi filename    | ファイルを開くまたは新規作成する       |
| :w Return キー   | バッファの内容を保存する           |
| / string       | 文字列を検索                 |
| i              | カーソルの左にテキストを追加         |
| x              | カーソルの文字を削除             |
| s              | カーソルの 1 文字をテキストに変換     |
| Esc            | コマンドモードに戻る             |
| :q Return キー   | 保存しないで終了               |
| :wq Return キー  | 変更を保存 (バッファをファイルに書き込む) |
| :wq! Return キー | 変更を保存して vi を終了         |
| :q! Return キー  | キー変更を保存しないで vi を終了     |

### 3. 研修環境の構築

研修者は、サーバ管理者としての権限で作業を行う必要があるため、サーバの構築環境のセキュリティ面に注意を払わなければならない。そこで、クラウド基盤を構築するオープンソース OpenStack を利用した仮想サーバ環境を学内 LAN 配下の無線 LAN 環境(192.168.xxx.0/24)に設置し、LAMP 環境にて、WEB サーバと DB サーバの 2 台構成を構築する。次に学習管理システムとして利用されるオープンソース Moodle をインストール、

セットアップを行う。本研修のサーバの利用環境を表 4 に示す。PC ホストと仮想サーバの構成概略を図 1 に、外観を図 2 に示す。OpenStack は2010年からリリースされたクラウドコンピューティング構築向けのオープンソースであり、今回2016年・秋リリースの Newton

表4 利用環境

|                                  |
|----------------------------------|
| (ハードウェア)                         |
| •PC: HP EliteDesk 800 G1 DM      |
| CPU: Intel Core i7-4770 4コア8スレッド |
| メモリー: 16GB/SSD: 480 GB           |
| NW: 2NIC                         |
| (ソフトウェア)                         |
| •CentOS 7.4版(1708)               |
| •OpenStack/Newton版(2016.2)       |

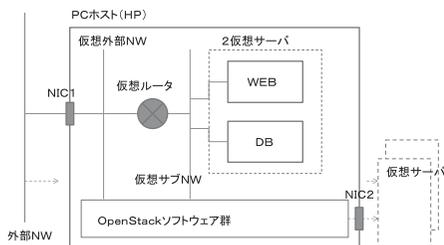


図1 PCホストと仮想サーバ



図2 PC、無線LAN等の写真

版<sup>3)</sup>を使用する。仮想サーバの Moodle3.4<sup>4)</sup>では、Linux (CentOS7.4)、Apache (httpd2.4)<sup>5)</sup>、MySQL (mysql5.7)<sup>6)</sup>、PHP (php7.1)<sup>7)</sup>を使用するため、rpm (RPM Package Manager、Red Hat Package Manager) リポジトリは、標準に CentOS-SCL リポジトリ<sup>8)</sup>を追加する。

### 3.1 仮想サーバの生成

研修では、WEBサーバはCentOS+Apache+PHPの環境、DBサーバはCentOS+MySQLの環境を利用し、各サーバのOSは既にインストール完了した状態(OSイメージ)として、次のようにOpenStackにログイン後、Dashboard画面を使ってそれぞれの仮想サーバを生成する。

- ①与えられたIDとパスワードでログイン
- ②仮想サーバの環境確認(図3)  
接続する外部ネットワーク、起動するOSイメージ<sup>9)</sup>、起動するOSサイズ等(表5)
- ③仮想サーバ用サブネットワークの作成
- ④仮想ルータの作成
- ⑤ネットワーク間の接続(GWの設定)
- ⑥Moodleデータ用の永続ボリュームの作成
- ⑦アクセスとセキュリティの設定  
セキュリティグループのルール作成、キーの生成・ダウンロード、FloatingIP(仮想外部NW)の確認
- ⑧仮想サーバ(インスタンス)の起動
- ⑨正常起動の確認、休止・再起動の確認
- ⑩起動インスタンスへのSSH接続の確認

コマンドプロンプトが表示され、起動した仮想サーバ(またはインスタンス)がネットワークに繋がっているかをpingコマンドにて応答確認する。



図3 仮想サーバの環境確認 (OpenStack/Dashboard画面より)

表5 仮想サーバの環境確認 (補足説明)

- 外部ネットワーク:192.168.xxx.0/24
- OS:CentOS7.4イメージ
- CPU:1仮想コア/メモリ:2GB/ディスク:10GB
- 永続ボリューム:5GB以下(WEBのみ割当)
- 追加開設ポート:  
WEB)http/https/ssh/icmp  
DB)mysql/ssh/icmp



## 2) フェーズ 2

- ① httpd2.4設定ファイルの保存・修正（補足 2-1；後述の付録参照のこと）
- ② php7.1設定ファイルの修正（補足 2-2）
- ③ httpd2.4サーバの再起動

上記の終了後、再度 Apache テストページを表示し、「Forbidden」状態となっているか確認する。

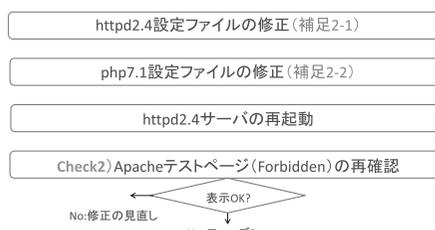


図 6-2 作業フロー 3) 仮想 WEB サーバの構築（フェーズ 2）

## 3) フェーズ 3

- ① SCL リポジトリより、mysql コマンドのインストール、DB 接続確認

ここで、本サーバから仮想 DB サーバに mysql コマンドを使用して接続できるか確認する。

- ② Moodle プログラム等のアップロード（補足 3-1）
- ③ Moodle プログラム等の配置・解凍、権限等の設定
- ④ Moodle データ用永続ボリュームの生成・初期化・マウント<sup>10)</sup>
- ⑤ Moodle データ用ディレクトリの作成・権限設定
- ⑥ 2 仮想サーバ(WEB/DB)接続の SELinux の設定
- ⑦ Moodle 設定ファイルの作成・修正（補足 3-2）
- ⑧ httpd2.4の再起動
- ⑨ Moodle サイトへのアクセス確認（補足 3-3）

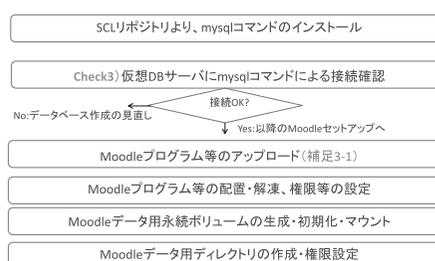


図 6-3 作業フロー 4) 仮想 WEB サーバの構築（フェーズ 3）

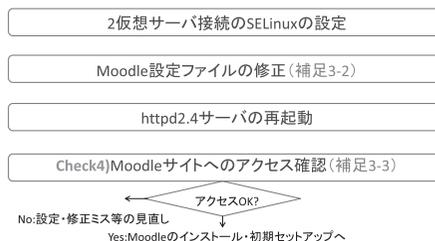


図 6-4 作業フロー 5) 仮想 WEB サーバの構築(フェーズ 3 : つづき)

以上で、2つの仮想サーバのセットアップは終わり、上記⑨の Moodle サイトの URL ([http://192.168.xxx.yyy/moodle\\_34/](http://192.168.xxx.yyy/moodle_34/)) にアクセスして表示される Installation 画面に合わせて、最低限の必要な項目 (Admin 設定 : トップページ設定、New password、Email address、Email display(hide)) を入力する。この Moodle の初期設定が終了したら、サイトの標準トップページ (Dashboard) が表示され、Moodle のインストールは完了となる。いったんログアウトして、Admin ユーザでログインできるか再確認する。

## 4. Moodle サイトの基本的な設定と運用

Moodle を運用する際のサイト設定、利用者登録、コース作成・設定、およびコースバックアップ等の基本作業を実際に行う。その他として、システム管理等のツールのインストール・動作確認を行う。

### 4.1 Moodle のサイト設定 (Admin)

Admin/Site Administration 画面にて、

- ① 言語の変更：基本は英語表示であるので、画面表示を日本語にする場合はタイトルメニューの言語選択を日本語 (JA) に変更 (図 7)
- ② セキュリティ/HTTPセキュリティ：「ログインに HTTPS を使用する」を有効\*、  
セキュリティ/通知：「ログイン失敗を表示する」を有効
- ③ モバイルアプリ/モバイル設定：「モバイルデバイスの Web サービスを有効にする」を有効
- ④ サーバ/セッションハンドリング：「クッキー接頭辞」を入力
- ⑤ プラグイン/認証/認証管理：「E メールによる自己登録」を無効、  
：「ゲストログインボタン」を非表示

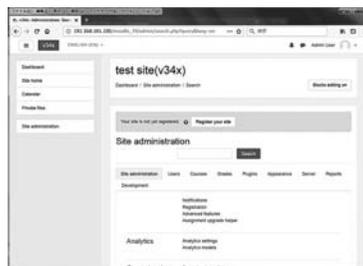


図 7 Moodle の設定例  
Admin/Site Administration 画面にて設定する

等を設定する。

### 4.2 ユーザ作成・コース作成 (Admin)

- ① ユーザ作成：新規ユーザの登録  
教師 (Teacher) は手動登録、学生 (複数) は csv でアップロード<sup>11)</sup>
- ② コースカテゴリの作成 (親カテゴリ、カテゴリ名の入力)
- ③ 新規コースの手動作成  
長いコース名/コース省略名の入力、コースフォーマットの指定、アップロードサイズの制限、登録済みユーザの教師登録
- ④ コースの csv アップロード<sup>12)</sup>による作成

### 4.3 教師権限でのコース編集 (Teacher)<sup>13)</sup>

- ① コース受講者 (学生) の登録 (手動)
- ② コース内セクション (週単位) の編集

※v3.4以降、この loginhttps 設定は削除され、代わりに cookiesecure 設定が標準で有効になった。

資料（リソース）の設置、活動（課題や小テスト等）の設置

- ③ コースの表示確認（学生権限での表示確認）

#### 4.4 Moodle プラグイン<sup>14)</sup>の設定・利用（Admin）

Moodle プラグインには、さまざまなタイプがあり、仕様にて独自開発・利用可能

- ① コースでの利用

活動モジュール、ブロックモジュール、リソースモジュールの3種類あり

- ② html エディタ（atto<sup>15)</sup>での利用

数式エディタ（標準）、cloze エディタ（追加）等、利用者で作成しアイコン化可能

- ③ テーマ変更<sup>16)</sup>

標準テーマは、Boost（v3.2以降に変更）、旧標準テーマは Clean  
デバイス特徴に合わせたテーマ指定が可能

#### 4.5 コースのバックアップ<sup>17)</sup>（Admin/Teacher）

- ① コース画面での手動バックアップ（Teacher）

ユーザ情報の保存有無指定

- ② 管理画面でのバックアップ（Admin）

- ・ 一般バックアップのデフォルト変更
- ・ 自動バックアップの設定（有効／無効／手動）

#### 4.6 コースのリストア<sup>17)</sup>（Admin/Teacher）

- ① 教師でのリストア

- ② 管理者でのリストア

教師：担当コースへの追加または、上書き

管理者：上記に加えて新規コース作成

#### 4.7 その他の設定

- ① Cron 処理の起動<sup>18)</sup>

- ② メール利用

・ Web サーバから PHP 送信（/etc/postfix/main.cf の変更）

・ 外部メールサーバ指定での送信

## 5. 考察

本拠点では、2016年11月に ICT 活用教育（eラーニング）実施状況について、日本にあるすべての大学（短期大学を含む）1,109校にアンケートを依頼した結果、全体で304校（27%）

から回答があった。eラーニングを活用した教育を実施している大学は214校で、そのうち「大学全体で実施している」と回答した大学は61校（29%）であった。運用している学習管理システム（LMS）は、オープンソースのMoodleが94校で、一時期eラーニングを風靡したBlackboard（WebCT）は9校であった。Call-system（言語系）は50校であった。Callシステムは、かなりの大学に導入されたことを考えると、あまり現在使われていないのだろう。その他が、115校で、WebClass（23校）、ALC NetAcademy 2（10校）、manaba（11校）、CoursePower（7校）、Glexa（5校）、Google Classroom（5校）、Sakai（4校）、KnowledgeDeliver（2校）、C-Learning（2校）、独自開発のもの（13校）など様々あった。

eラーニングの導入を困難にしている原因を図8に示す。eラーニングを導入していない大学（86校）のうち、「実施したいが予算がない」が65校、「実施したいが担当できる教職員がいない」が49校であった。

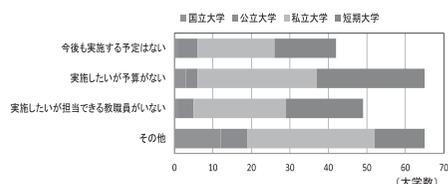


図8 eラーニングの実施について  
（複数回答可）

eラーニングを実施するためには、eラーニングの核をなすLMSをある程度理解し、できれば構築できる中心となる人物が必要となる。そのような理由で、最も使われており、これからもっと使われるであろうMoodleの構築についての研修を、少しでも役立てられることを期待して実施した。利用したPCホストの制約やセキュリティも考慮し、小人数できちんと伝達できるようにした。本研修は定員5名で2回実施した。1回目の参加者は5名、2回目は3名であった。1回目の研修の様子を図9に示す。定員は多くした方が研修の効率は良くなるが、初心者がいる場合はプログラムエラー等が頻繁に起こり対処に時間がかかるので、定員は3名程度がよさそうであった。また、本研修を受けることにより、本学と親密になることにより、安心して分からないところが聞ける繋がりが非常に大事である。研修の効果や満足度は非常に良かった。



図9 研修の様子

## 6. まとめ

クラウド基盤ソフトウェアを利用した仮想サーバ環境を前提にしたMoodleのインストール研修を実施して、次のような知見を得た。

- ・ Moodleのインストール研修は、サーバの設定を行ったことのある経験者にとっては10時間程度でも十分である。
- ・ これまでにサーバの設定を経験したことのない初心者には、10時間では厳しい。
- ・ 研修に参加するためにはLinuxコマンドやviエディタのコマンドに対してある程度の知

識が必要である。

- ・初心者もいる場合、入力ミス・設定漏れ等でのプログラムエラーが起こった場合に対処するには定員は5名では多く、3名以内が良いようである。
- ・クラウド環境が一般的になってきた今、Moodle インストールを体験できる研修はeラーニングの利用普及に貢献できる。

## 謝辞

本研究を進めるに当たり、多大のご協力をいただいた福崎優子氏、およびクリエイティブ・ラーニングセンターのスタッフの皆様がこの場を借りて、あらためて感謝の意を表す。なお、本実践研究の一部は、2014年度科学研究費補助金（基盤研究（B）一般、研究課題名：eラーニングと協同学習を効果的に利用して反転授業を促す教育改革の研究）の補助、および教育関係共同実施分「全国の大学間連携による ICT 活用教育の拡充と教育改革の推進」により行った。

## 引用・参考文献資料

- 1) OpenStack/Software について：<https://www.openstack.org/software> (2018/1/15アクセス)
- 2) Solaris ユーザーズガイド（上級編）>第6章 vi エディタの使い方：  
<https://docs.oracle.com/cd/E19683-01/816-3946/6ma6m5bnf/index.html> (2018/1/15アクセス)
- 3) OpenStack/Newton のデモ：<https://www.youtube.com/watch?v=z6ftW7fUdp4> (2018/1/15アクセス)
- 4) Moodle 3.4：[https://docs.moodle.org/34/en/Main\\_page](https://docs.moodle.org/34/en/Main_page) (2018/1/15アクセス)
- 5) SCL/httpd 24：<https://www.softwarecollections.org/en/scls/rhscl/httpd24/> (2018/1/15アクセス)
- 6) SCL/rh-mysql 57：<https://www.softwarecollections.org/en/scls/rhscl/rh-mysql57/> (2018/1/15アクセス)
- 7) SCL/rh-php 71：<https://www.softwarecollections.org/en/scls/rhscl/rh-php71/> (2018/1/15アクセス)
- 8) SCL リポジトリ：<https://wiki.centos.org/AdditionalResources/Repositories/SCL> (2018/1/15アクセス)
- 9) OpenStack イメージ：<https://docs.openstack.org/ja/image-guide/obtain-images.html> (2018/1/15アクセス)
- 10) 日本 OpenStack ユーザ会：OpenStack クラウドインテグレーション ISBN978-4-7981-3978-4 (2015)
- 11) Moodle/ユーザアップロード：[https://docs.moodle.org/34/en/Upload\\_users](https://docs.moodle.org/34/en/Upload_users) (2018/1/15アクセス)
- 12) Moodle/コースアップロード：[https://docs.moodle.org/34/en/Upload\\_courses](https://docs.moodle.org/34/en/Upload_courses) (2018/1/15アクセス)

- クセス)
- 13) Moodle/コースの管理 : [https://docs.moodle.org/34/en/Managing\\_a\\_Moodle\\_course](https://docs.moodle.org/34/en/Managing_a_Moodle_course) (2018/1/15アクセス)
  - 14) Moodle/Moodle プラグイン検索 : <https://moodle.org/plugins/> (2018/1/15アクセス)
  - 15) Moodle/Atto エディタ : <https://moodle.org/plugins/browse.php?list=category&id=53> (2018/1/15アクセス)
  - 16) Moodle/テーマ一覧 : <https://moodle.org/plugins/browse.php?list=category&id=3> (2018/1/15アクセス)
  - 17) Moodle/バックアップ・リストア : <https://docs.moodle.org/34/en/Backup> (2018/1/15アクセス)
  - 18) Moodle/Cron 処理 : <https://docs.moodle.org/34/en/Cron> (2018/1/15アクセス)

## 付録 仮想サーバの構築での操作コマンド

この付録は、本文 3.2.1 節、3.2.2 節でのコマンド操作履歴より抜粋した。

### ●操作履歴(1)／仮想 DB サーバのセットアップ (3.2.1)

- ① ssh 接続後の仮想サーバの状態確認
- ```
$ df
$ hostname
$ uname -a
$ ip a
$ lsblk
$ sudo fdisk -l
$ pwd
$ getenforce
```
- ←状態確認
- ② CentOS を更新後、仮想サーバの再起動
- ```
$ sudo yum clean all
$ sudo yum update -y
$ sudo reboot
```
- ←OS 更新
- ③ 再度ログインし、SCL リポジトリの追加・更新
- ```
$ sudo yum install centos-release-scl
$ sudo yum update -y
```
- ←SCL リポジトリ追加
- ④ SCL リポジトリより、MySQL5.7環境のインストール・起動
- ```
$ sudo yum install rh-mysql57-mysql
$ sudo yum install rh-mysql57-mysql-server
$ sudo systemctl enable rh-mysql57-mysqld
$ sudo systemctl start rh-mysql57-mysqld
$ sudo systemctl status rh-mysql57-mysqld
```
- ←mysql57をインストール  
←mysqld の起動設定および確認
- ⑤ Moodle 用データベースの生成・権限設定 (補足 1-1)
- ```
$ sudo scl enable rh-mysql57 bash
# mysql -u root
```
- ←mysql コマンドの有効  
←mysql コマンドの実行 (root/password なし)

補足 1-1. 利用する mysql コマンド (root)

- ```
> set password for root@localhost=password('????');
> quit
```
- ←セミコロン要
- ```
# mysql -u root -p ←mysql コマンドの実行 (root/pass 入力での moodle 用データベース、権限作成)
```
- ```
> show databases;
> create database aaaaaa default character set utf8mb4;
> grant all privileges on aaaaaa.* to xxxxxx@192.168.xxx.yyy identified by 'zzzzzz';
```
- ←aaaaaa : Moodle データベース名  
←xxxxxx : Moodle データベースのユーザ名  
←zzzzzz : 上記ユーザのパスワード  
←192.168.xxx.yyy : WEB サーバ IP アドレス
- ```
> show databases;
> quit
```

●操作履歴(2)/仮想 Web サーバのセットアップ

1) フェーズ 1

① 3.2.1項の①から③と同様

操作履歴(1)の①、②、③を参照

② 標準 httpd 環境のインストール

```
$ sudo yum install httpd mod_ssl
```

←標準 httpd のインストール

③ SCL リポジトリより、php7.1/httpd2.4環境をインストール

```
$ sudo yum install rh-php71-php rh-php71-php-mbstring rh-php71-php-mysqlnd
```

```
$ sudo yum install rh-php71-php-xml rh-php71-php-intl rh-php71-php-soap rh-php71-php-xmlrpc
```

```
$ sudo yum install rh-php71-php-gd
```

←追加 php モジュール (漏れ分)

```
$ sudo yum install rh-php71-php-opcache
```

```
$ sudo yum install httpd24-mod_ssl
```

←https 用の追加

④ httpd2.4サーバの起動

```
$ sudo systemctl enable httpd24-httpd
```

←httpd 24の起動設定および起動確認

```
$ sudo systemctl start httpd24-httpd
```

```
$ sudo systemctl status httpd24-httpd
```

★Check 1) ブラウザにて web サーバにアクセスし、apache テストページの表示確認

2) フェーズ 2

① httpd2.4設定ファイルの保存・修正 (補足 2 - 1)

```
$ cd /opt/rh/httpd24/root/etc/httpd/conf
```

```
$ sudo cp httpd.conf httpd.conf.new
```

```
$ sudo chcon --reference=httpd.conf httpd.conf.new
```

```
$ ls -Za
```

```
$ sudo mv httpd.conf httpd.conf.org
```

←旧 httpd.conf の保存

```
$ sudo mv httpd.conf.new httpd.conf
```

```
$ sudo vi httpd.conf
```

補足 2 - 1. httpd.conf ファイルの修正 (/opt/rh/httpd24/root/etc/httpd/conf/httpd.conf)

a) 公開パスの変更

/opt/rh/httpd24/root/var/www/html 等の文字列から、/opt/rh/httpd24/root 部分を除く：5 か所、Options 指定の変更

...

```
#DocumentRoot "/opt/rh/httpd24/root/var/www/html" <=1/5
```

```
DocumentRoot "/var/www/html"
```

...

```
#<Directory "/opt/rh/httpd24/root/var/www"> <=2/5
```

```
<Directory "/var/www">
```

...

```
#<Directory "/opt/rh/httpd24/root/var/www/html"> <=3/5
```

```
<Directory "/var/www/html">
```

...

```
#Options Indexes FollowSymLinks <=コメントアウト
```

```
Options FollowSymLinks <=Indexes 削除 (ディレクトリ一覧の非表示)
```

...

```
#ScriptAlias /cgi-bin/ "/opt/rh/httpd24/root/var/www/cgi-bin" <=4/5
```

```
ScriptAlias /cgi-bin/ "/var/www/cgi-bin"
```

...

```
#<Directory "/opt/rh/httpd24/root/var/www/cgi-bin"> <=5/5
<Directory "/var/www/cgi-bin">
※上記ファイルの修正後、入力ミス等がないか確認すること
```

```
b) welcome ページの非表示 (/opt/rh/httpd24/root/etc/httpd/conf.d/welcome.conf)
#Options -Indexes <=コメントアウト
#ErrorDocument... <=コメントアウト
```

- 
- ② php7.1設定ファイルの修正 (補足2-2)
- ```
$ cd /etc/rh/rh-php71
$ sudo vi php.ini
```

補足2-2. php.iniの修正 (/etc/opt/rh/rh-php71/php.ini)

```
max_execution_time=300 ←30
post_max_size=200 M ←8 M
upload_max_filesize=200 M ←2 M
```

- 
- ③ httpd2.4サーバの再起動
- ```
$ sudo systemctl restart httpd24-httpd ←設定ファイル修正後のhttpdの再起動
$ sudo systemctl status httpd24-httpd
```

★Check 2) webサーバにて、apache テストページ => 「Forbidden」 ページの表示確認

### 3) フェーズ3

- ① SCLリポジトリより、mysqlコマンドのインストール・DB接続確認
- ```
$ sudo yum install rh-mysql57-mysql ←mysqlコマンド関連のインストール
```

```
$ sudo scl enable rh-mysql57bash
# mysql -u xxx -p -h192.168.xxx.zzz ←接続確認 (OK)
```

★Check 3) 上記コマンドにて、webサーバからdbサーバへの接続確認

- ② Moodleプログラム等のアップロード (補足3-1)
- ```
$ cd ←ホームディレクトリ (centos) への移動
$ mkdir upload ←uploadディレクトリ作成
... scp 等でのファイルアップロード (moodle 等)
```

#### 補足3-1. scpコマンド例 (root)

・PCから仮想サーバ (WEB) へアップロード)

```
# scp -i ./xxx_key.pem ./moodle-latest-34.tgz centos@192.168.xxx.yyy:/home/centos/upload/
```

・仮想サーバ (WEB) からPCへダウンロード)

```
# scp -i ./xxx_key.pem centos@192.168.xxx.yyy:/home/centos/upload/xxxx.zip /xxx x/
```

- 
- ③ Moodleプログラム等の配置・解凍、権限等の設定
- ```
$ cd upload
$ sudo cp moodle-latest-34.tgz /var/www/html/ ←ファイルのコピー (moodle/v3.4)
$ sudo cp theme_photo_moodle32_2017022000.zip /var/www/html/ ←同上 (Boost子テーマ/photo)

$ cd /var/www/html
$ sudo tar zxvf moodle-latest-34.tgz ←moodle 解凍
$ ls -Za
$ sudo chown root:root moodle -R ←ディレクトリ、ファイルの所有者が異なる場合
$ sudo mv moodle moodle_34 ←解凍先リネーム
```

- ```

$ sudo yum install unzip
$ sudo unzip theme_photo_moodle32_2017022000.zip
$ ls -la
$ sudo chmod 755 photo -R
$ ls -la
$ sudo cp photo moodle_34/theme/ -R
$ ls theme

```
- ←unzip インストール  
←子テーマ解凍、設置
- ④ Moodle データ用永続ボリュームの生成・初期化・マウント
- ```

$ cd
$ lsblk /dev/vdb

```
- ←web サーバへの永続ボリューム接続確認  
(未接続の場合、openstack 側にて接続実施)  
←仮想ボリュームのパーティション生成
- ```

$ sudo fdisk /dev/vdb
... 入力順 (n->p->1->リターン 2 回->w)

```
- ```

$ sudo fdisk -l
$ lsblk
$ sudo mkfs.xfs -L mydata /dev/vdb1
$ sudo tune2fs -c 0 -i 0 -r 0 /dev/vdb1
$ sudo mkfs.ext4 -L mydata /dev/vdb1
$ sudo tune2fs -c 0 -i 0 -r 0 /dev/vdb1

```
- ←実行後の確認  
←永続ボリュームの初期化等  
←上記でのエラー時の別タイプでの再初期化
- ```

$ sudo mkdir /mydata
$ sudo mount LABEL=mydata /mydata
$ df

```
- ←マウント先の作成、マウント確認
- ⑤ Moodle データ用ディレクトリの作成・権限設定
- ```

$ cd /mydata
$ sudo mkdir moodledata
$ ls -la
$ cd moodledata
$ sudo mkdir moodle_34
$ cd ..
$ ls
$ sudo chmod 777 moodledata -R
$ sudo chown apache:apache moodledata -R
$ sudo chcon --reference=/var/www/html moodledata -R
$ ls -Za

```
- ←Moodle データ用ディレクトリ作成  
←アクセス権限等の変更、設定確認
- ⑥ 2 仮想サーバ (WEB/DB) 接続の SELinux の設定
- ```

$ getsebool -a |grep httpd
$ sudo setsebool httpd_unified 1 -P
$ sudo setsebool httpd_can_network_connect_db 1 -P

```
- ←SELinux/httpd 関連ブール値の確認  
←2 ブール値の変更
- ⑦ Moodle 設定ファイルの作成・修正 (補足 3 - 2)
- ```

$ cd
$ cd /var/www/html
$ ls
$ cd moodle_34
$ sudo cp config-dist.php config.php
$ sudo vi config.php

```
- ←moodle 設定ファイルの作成  
←上記ファイルの修正

補足 3 - 2. 基本的な config.php の修正 (/var/www/html/moodle\_34/config.php)

```
...
$CFG->dbtype    = 'mysqli';      // 'pgsql', 'mariadb', 'mysqli', 'mssql', 'sqlsrv' or 'oci'
...
$CFG->dbhost     = '192.168.xxx.zzz'; // eg 'localhost' or 'db.isp.com' or IP
$CFG->dbname     = 'aaaaaa';      // database name, eg moodle
$CFG->dbuser    = 'xxxxxx';      // your database username
$CFG->dbpass     = 'zzzzzz';      // your database password
...
$CFG->wwwroot   = 'https://192.168.xxx.yyy/moodle_34'; ←web サーバの ip アドレス (https 強制)
...
$CFG->dataroot  = '/mydata/moodledata/moodle_34'; ←永続ボリューム
※上記ファイルの修正後、入力ミス等がないか確認すること (★補足 1 - 1 との整合性)
```

---

⑧ httpd2.4の再起動

```
$ sudo systemctl restart httpd24-httpd          ←httpd の再起動、確認
$ sudo systemctl status httpd24-httpd
```

⑨ Moodle サイトへのアクセス確認 (補足 3 - 3)

補足 3 - 3. 下記 URL でサイトにアクセスして、Moodle インストール開始・初期ページの表示確認

URL) [https://192.168.xxx.yyy/moodle\\_34](https://192.168.xxx.yyy/moodle_34)

★Check 4) エラー表示が出た場合、以下の再確認

```
-2sebool 値の設定の確認
-moodledata 配下のアクセス権限の確認
-config.php の設定内容 (利用の db タイプや ip アドレス等) の確認
- ...
```

---

●その他

※仮想サーバを再起動した場合、永続ボリューム (moodledata 等) のマウント切れへの対処例

1) ssh 接続後、下記コマンドによる手動マウント

```
$ sudo mount LABEL=mydata /mydata
```

2) rc.local(/etc/rc.d/rc.local)ファイルによる自動マウント

-上記ファイルに下記1行を追加・保存する

```
mount LABEL=mydata /mydata
```

-次に、ファイル権限を変更して本ファイルの実行を有効にする

```
$sudo chmod 755 /etc/rc.d/rc.local
```

※仮想サーバのタイムゾーンの変更 (UTC から JST へ)

```
$ sudo timedatectl set-timezone Asia/Tokyo
```

---