

09 VirtualBoxでの学習準備

VirtualBoxはオラクル社が提供している仮想マシンを実行するためのソフトです。

仮想マシンとは、パソコンの中に、もう1台仮想的なパソコンを用意するようなものです。仮想マシンは、複数台作ることもできます。

本書は、Desktop版を使用したDockerの解説書であるため、VirtualBoxの操作方法について詳しく解説しませんが、以下に簡単にインストール方法を紹介しておくので、参考にしてください。VirtualBoxの使用方法については、別途書籍やWebサイトなどでご確認ください。

[手順] VirtualBoxを準備する① インストール

この手順ではまず、VirtualBoxを用意します。

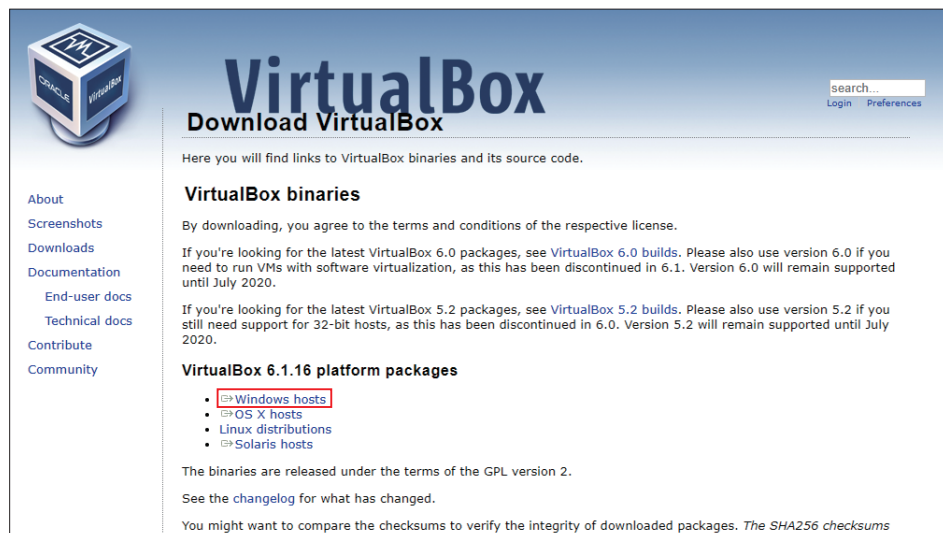
STEP 0 ダウンロードする

ブラウザでVirtualBoxのダウンロードページにアクセスします。Windowsの場合は、「VirtualBox binaries」の「Windows hosts」をクリックします。すると、ブラウザの下部に「VirtualBox-6.1.16-140961-Win(103MB)について行う操作を選んでください。」と表示されるので保存してください。

・VirtualBoxのダウンロードページ

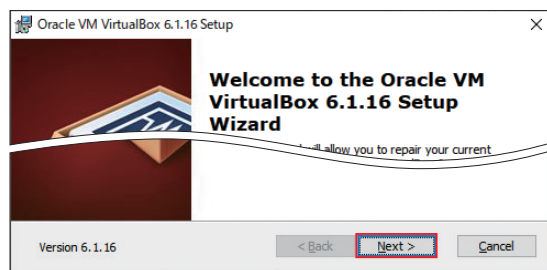
<https://www.virtualbox.org/wiki/Downloads>

VirtualBoxのファイルをダウンロードして保存したら、それをダブルクリックして起動し、次のようにしてインストールします。



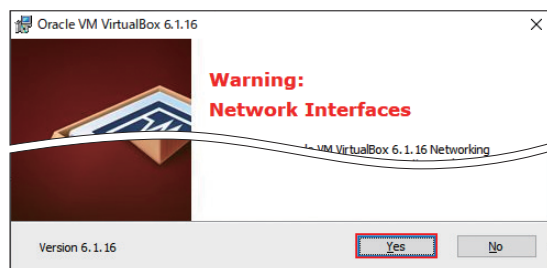
STEP ① ウィザードを進める

セットアップウィザード画面が表示されます。
[Next] をクリックします。そのまま画面に沿ってインストールを進めてください。特に要件がなければ、標準のままで進めて構いません。



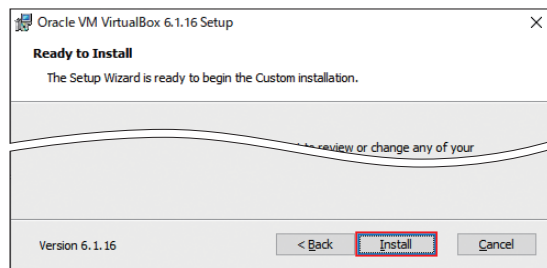
STEP ② ネットワークの一時切断の警告

インストール中ネットワークが一時的に切断されるという警告が表示されます。例えば、ファイルをダウンロードしているときなどは中断する恐れがあります。切断しても問題がなければ、[Yes] をクリックして、次の画面に進んでください。



STEP ③ インストールの開始

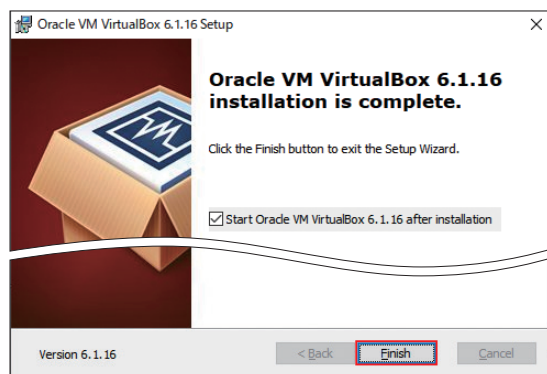
[Install] をクリックして、インストールを開始します。ユーザーアカウント制御のダイアログが表示されるので、[はい] をクリックします。途中、デバイスドライバのインストールが始まり、[Windows セキュリティ] ダイアログが表示されます。表示されたら [インストール] をクリックしてください。



STEP ④ インストールの完了と VirtualBox の起動

しばらくすると、インストールが完了した画面が表示されます。[Finish] をクリックすると、インストールが完了します。

このときに、[Start Oracle VM VirtualBox 6.1.16 after installation] にチェックを付けておくと、すぐに VirtualBox が起動します。もし、間違って [×] ボタンを押してしまった場合は、[スタート] メニューから [Oracle VM VirtualBox] — [Oracle VM VirtualBox] をクリックして、VirtualBox を起動してください。

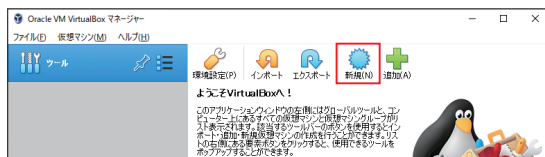


[手順] VirtualBoxを準備する② 新規仮想マシンの作成

用意したVirtualBoxに、仮想マシンを作ります。このマシンの中に、Linuxを入れることになります。

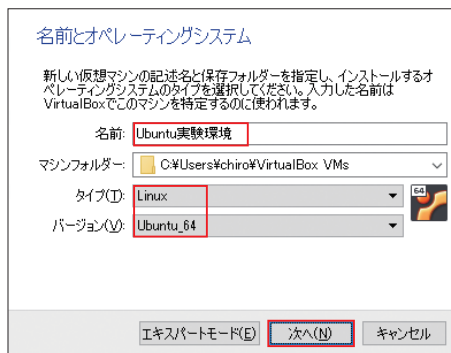
STEP ① 新規仮想マシンの起動

[新規] ボタンをクリックして、新しい仮想マシンを作成します。



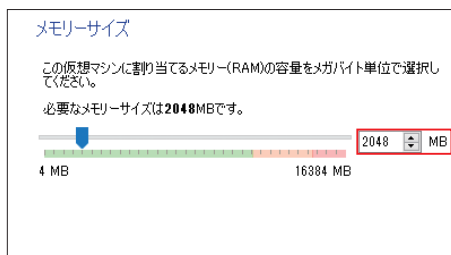
STEP ② 仮想マシン名と OS を決める

仮想マシン名とOSを決めます。仮想マシン名は、わかりやすければどんなものでも良いのですが、本書では、仮想マシン名は「Ubuntu実験環境」としました。タイプには「Linux」を選択し、バージョンには「Ubuntu_64」を選択してください。



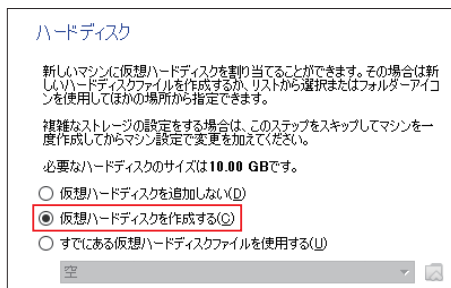
STEP ③ メモリ容量を決める

割り当てるメモリ容量を決めます。ここでは2048MBにしておきますが、パソコンに搭載されているメモリの容量に応じて調整してください。たくさんさんのDockerコンテナを起動する場合は、これでは足りない可能性もあります。パソコンのメモリに余裕があれば、もう少し大きな値を設定しておくといでしょう。



STEP ④ 仮想ディスクを作成する

仮想サーバのディスクを設定します。[仮想ハードディスクを作成する]を選択して[作成]をクリックします。



STEP 5 ハードディスクタイプの設定

するとハードディスクのファイルタイプが尋ねられます。デフォルトでは「VDI」が選択されているので、そのまま[次へ]ボタンをクリックしてください。

ハードディスクのファイルタイプ

新しい仮想ハードディスクで使いたいファイルのタイプを選択してください。もしほかの仮想ソフトウェアで使用する必要がなければ、設定はそのままにしてください。

- ☒ VDI (VirtualBox Disk Image)
- ☐ VHD (Virtual Hard Disk)
- ☐ VMDK (Virtual Machine Disk)

STEP 6 ストレージの設定

可変サイズにするか固定サイズにするかを決めます。ここでは「可変サイズ」を設定します。すると、実際に保存に必要な容量しか消費されないの、ディスク容量を節約できます（仮に「固定サイズ」を指定すると、仮想マシンを作成したときに、たとえ、何もまだデータを保存していなくても、次の手順で指定した容量のファイルが確保されます）。

物理ハードディスクにあるストレージ

新しい仮想ハードディスクファイルは使用したぶんだけ大きくなるか（可変サイズ）、または最大サイズで作成するか（固定サイズ）を選択してください。

可変サイズのハードディスクファイルは使用した分だけ（固定サイズを上限として）物理ハードディスクの領域を消費しますが、スペースを開放しても自動的に縮小はしません。

固定サイズのハードディスクファイルはシステムによっては作成に時間がかかるかもしれませんが、使用すると高速です

- ☒ 可変サイズ(D)
- ☐ 固定サイズ(E)

STEP 7 ファイルの場所とサイズ

ファイルの場所とサイズを決めます。場所はデフォルトのままでもかまいません。容量のデフォルトは10GBですが、Dockerでの利用を考えると、イメージのダウンロードなどで少し心細いため、20GBにしておきます。

ファイルの場所とサイズ

新しい仮想ハードディスクファイルの名前を下のボックスに入力するか、フォルダーアイコンをクリックしてファイルを作成する別のフォルダーを選択してください。

\\wa\VirtualBox VMs\Ubuntu実験環境\Ubuntu実験環境.vdi

仮想ハードディスクのサイズをメガバイト単位で指定してください。このサイズは仮想マシンがハードディスクに置くことができるファイルデータの上限です。

4.00 MB 2.00 TB 20.00 GB

STEP 8 仮想マシンの作成

以上で仮想マシンができました。「Ubuntu 実験環境」という名前の仮想サーバができたことがわかります。



[手順] VirtualBoxを準備する③ ISOイメージのマウントと微調整

用意したマシンに、Linux(Ubuntu)を入れます。そのためには、ISOイメージが必要です。あらかじめ書籍P.300を参考に、UbuntuのISOイメージをダウンロードしておいてください。VirtualBoxを使用するので、DVDに焼く必要はありません。

ISOイメージをマウントして、インストールするのですが、ここではまずマウントをします。

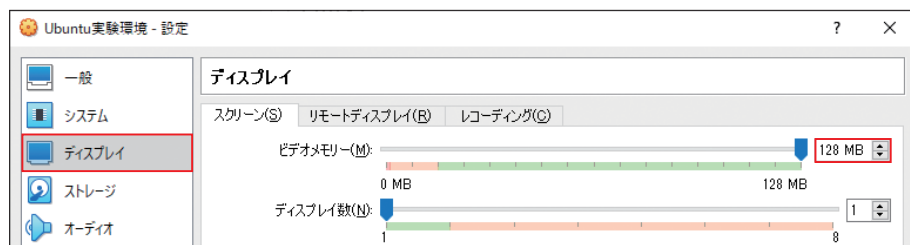
STEP ① 設定画面を開く

VirtualBoxの画面で「Ubuntu実験環境」をクリックして選択状態にし、[設定] ボタンをクリックして設定画面を開きます。



STEP ② ビデオ RAM を調整する

VirtualBoxとUbuntuの相性の問題で、ビデオRAMを128MBに変更しないと起動しない問題があります。設定の[ディスプレイ]の設定から、[ビデオメモリー]を「128MB」に変更してください。

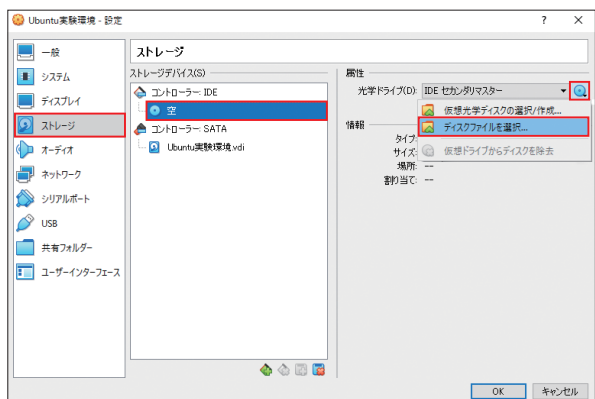


STEP ③ ISOイメージを挿入する

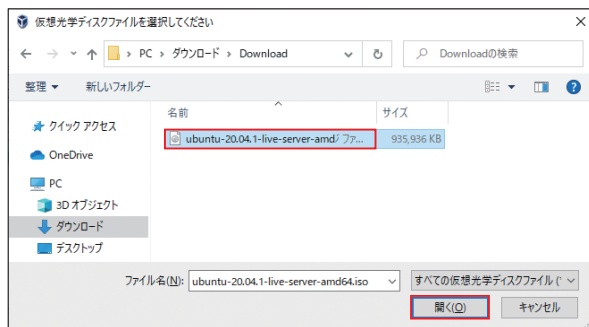
UbuntuのISOイメージをダウンロードしておきます（書籍P.300を参照）。そのISOイメージを、この仮想マシンに挿入した状態にします。

そのためには、[ストレージ] をクリックしてストレージの設定画面をクリックします。

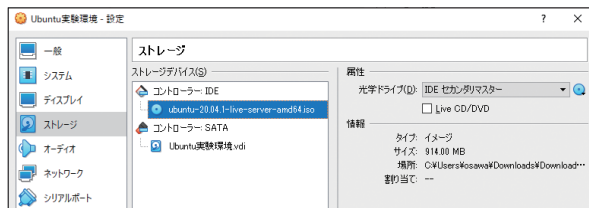
そして[コントローラIDE]の下にある[空]の部分をクリックします。右側の「属性」に表示されているCDのアイコンをクリックすると「ディスクファイルを選択」というメニューが表示されるのでそれを選択します。



すると、ISO ファイルを選択できるので、すでにダウンロードしたUbuntuのイメージファイルを選択し、[開く] ボタンをクリックします。



そして最後に [OK] ボタンをクリックします。



[手順] VirtualBoxを準備する④ Linux (Ubuntu) をインストール

ようやくUbuntuをインストールします。仮想マシンを起動し、インストール操作を行います。

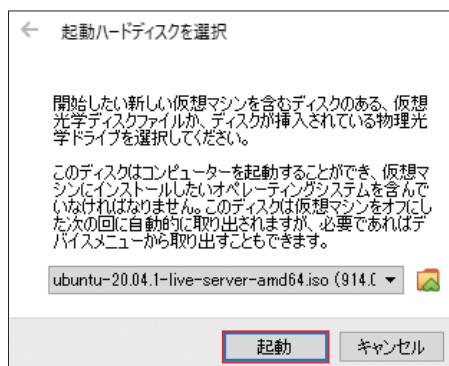
STEP ① 仮想マシンを起動する

「Ubuntu実験環境」をクリックして選択した状態にして、[起動] ボタンをクリックします。

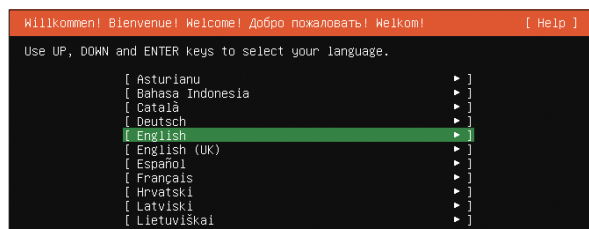


STEP ② 仮想マシンが起動しインストーラが起動する

別ウィンドウで仮想マシンが動きます。はじめて起動したときは、ブートするメディアの選択画面が表示されるので、UbuntuのISOイメージを選択してください。



しばらくするとUbuntuのインストーラが起動します。インストールの方法については、書籍P.301を参照してください。またその後のDockerのインストールについては、書籍P.307を参照してください。

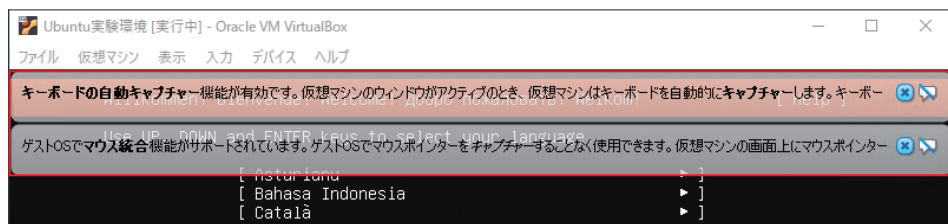


COLUMN

VirtualBoxの操作

最初にVirtualBoxを起動したときは、下図のようにいくつかのメッセージが表示されることがあります。これらのメッセージは、VirtualBoxでの操作

を説明するものなので、[×] をクリックして閉じてください。



VirtualBoxでは、ウィンドウをクリックするとそれ以降、VirtualBox内で操作できます。VirtualBox外の操作をするには、マウス操作してマウスポインタをVirtualBoxの外に出すか、キーボード右側の[Ctrl] キーを押します。

なおVirtualBoxの画面に対して、コピー ([Ctrl] + [C]、[Ctrl] + [V]) をすることはできません (WindowsやUbuntu DesktopなどGUI環境ではドライバをインストールすることでコピーできるようになりますが、その詳細は省略します)。



COLUMN

仮想マシンを停止するには

VirtualBoxを終了させたいときは、コマンドから「sudo shutdown -h now」と入力してシャットダウンするのが基本ですが、VirtualBox上でのマウス操作でも終了させることができます。サーバ名を右クリックして「閉じる」→「ACPI シャットダウン」をクリックします。しばらくすると、この学習環境が終了します。もう一度起動したいときは、先と同じ手順で起動し直してください。

サーバは電源をいきなり落とすとデータが壊れる恐れがあります。VirtualBoxによる仮想サーバの場合も同じです。必ず、「ACPIシャットダウン」で終了させてください。[電源オフ] は、物理的に電源を落とすことです。学習環境が壊れる恐れがあるので選択してはいけません。

10 AWS での学習準備

AWS の仮想マシンである「EC2」に Ubuntu をインストールして、Docker を利用することもできます。その手順を簡単に下記に示します。

なお下記では、AWS アカウントをすでに持っていて、AWS マネジメントコンソールから操作できる人を対象としています。アカウントの作成方法や操作方法などは、別途専門書や Web サイトなどで確認してください。

[手順] AWS に Ubuntu をインストールする

STEP ① EC2 コンソールを起動する

AWS マネジメントコンソールから「EC2」を検索し、EC2 コンソールを起動します。



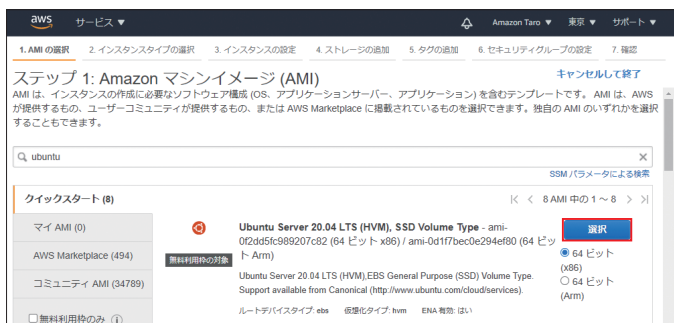
STEP ② インスタンスを作る

左メニューの「インスタンス」をクリックして開きます。[インスタンスを起動] をクリックして、インスタンスを作ります。



STEP ③ Ubuntu の AMI を選択する

Ubuntu を利用します。[Ubuntu Server 20.04 LTS (HVM), SSD Volume Type] を選択してください。



STEP 4 インスタンスタイプを選ぶ

インスタンスタイプを選びます。

ここでは、無償枠範囲である [t2.micro] を選択し、[確認と作成] を選択します。

※t2.microではKubernetesが利用できません。Kubernetesを利用するにはCPUが2つ以上ある、たとえば「t3.micro」などにする必要があります。ただしt3.microは無償利用枠の範囲ではありません。

ステップ 2: インスタンスタイプの選択

Amazon EC2 では、異なるユースケースに合わせて最適化されたさまざまなインスタンスタイプが用意されています。インスタンスを実行できる仮想サーバーです。インスタンスタイプはさまざまな CPU、メモリ、ストレージ、ネットワークキャパシティが用意されているため、使用するアプリケーションに合わせて適切なリソースの組み合わせを柔軟に選択できます。インスタンスタイプのニーズに適用する方法に関する [詳細はこちら](#)。

フィルター条件: All instance families 現行世代 列の表示/非表示

現在選択中: t2.micro (- ECU, 1 vCPU, 2.5 GHz, -, 1 GiB メモリ, EBS のみ)

	ファミリー	タイプ	vCPU	メモリ (GiB)	インスタンスストレージ (GB)	EBS 最適化利用
<input type="checkbox"/>	t2	t2.nano	1	0.5	EBS のみ	-
<input checked="" type="checkbox"/>	t2	t2.micro 無料利用枠の対応	1	1	EBS のみ	-

STEP 5 セキュリティグループの編集をはじめ

デフォルトの構成でEC2インスタンスが構成されます。ファイアウォールの設定を変更したいので、[セキュリティグループの編集] をクリックしてください。

インスタンスタイプの編集

インスタンスタイプ	ECU	vCPU	メモリ (GiB)	インスタンスストレージ (GB)	EBS 最適化利用	ネットワークパフォーマンス
t2.micro	-	1	1	EBS のみ	-	Low to Moderate

セキュリティグループ

セキュリティグループ名: launch-wizard-12
説明: launch-wizard-12 created 2020-11-21T17:05:09.768+09:00

セキュリティグループの編集

キャンセル 戻る 記録

STEP 6 セキュリティグループを設定する

デフォルトではSSHが通過するように構成されています。[ルールの追加] ボタンをクリックして、追加でいくつかのポートを通すように設定します。

続いて本書の Docker コンテナのサンプルのためにポート 8080 ~ ポート 8100 までを追加してください。

さらに、Kubernetesを使うのであれば、[ルールの追加] をクリックして、ポート 30080 を通す設定を加えてください。

最後に右下の [確認と作成] をクリックして完了です。

セキュリティグループの割り当て: ☒ 新しいセキュリティグループを作成する
☐ 既存のセキュリティグループを選択する

セキュリティグループ名: launch-wizard-12
説明: launch-wizard-12 created 2020-11-21T17:05:09.804+09:00

タイプ	プロトコル	ポート範囲	ソース
SSH	TCP	22	カスタム 0.0.0.0/0

ルールの追加

タイプ	プロトコル	ポート範囲	ソース
SSH	TCP	22	カスタム 0.0.0.0/0
カスタム TCP	TCP	8080-8100	カスタム 0.0.0.0/0

ルールの追加

タイプ	プロトコル	ポート範囲	ソース
SSH	TCP	22	カスタム 0.0.0.0/0
カスタム TCP	TCP	8080-8100	カスタム 0.0.0.0/0
カスタム TCP	TCP	30080	カスタム 0.0.0.0/0

ルールの追加

STEP 7 起動する

先の画面に戻るので「起動」ボタンをクリックして起動します。

▼ インスタンスタイプ

インスタンスタイプの編集

インスタンスタイプ	ECU	vCPU	メモリ (GiB)	インスタンスストレージ (GB)	EBS 最適化利用	ネットワークパフォーマンス
t2.micro	-	1	1	EBS のみ	-	Low to Moderate

▼ セキュリティグループ

セキュリティグループの編集

キャンセル 戻る 起動

STEP 8 キーペアの作成

この仮想マシンに SSH 接続するときに必要なキーペアを用意します。キーペアとは、パスワードのような認証ファイルです。

「新しいキーペアの作成」を選択し、適当なキーペア名を入力してください。ここでは、「mykey」という名前にします。「キーペアのダウンロード」をクリックしてダウンロードしてください。「キーペア名.pem」という名前でダウンロードできます。ダウンロードしたファイルは、なくさないようにしてください。なくしてしまうと、この EC2 インスタンスにアクセスすることはできなくなってしまう（つまり作り直しになります）。

なお、ダウンロードは、この画面限りです。ほかの画面に移動すると、もうダウンロードできないので注意してください。ダウンロードが終わったら「インスタンスの作成」をクリックしてください。

既存のキーペアを選択するか、新しいキーペアを作成します。 ×

キーペアは、AWS が保存するパブリックキーとユーザーが保存するプライベートキーファイルで構成されます。組み合わせることで、インスタンスに安全に接続できます。Windows AMI の場合、プライベートキーファイルは、インスタンスへのログインに使用されるパスワードを取得するために必要です。Linux AMI の場合、プライベートキーファイルを使用してインスタンスに SSH で安全に接続できます。

注 選択したキーペアは、このインスタンスに対して権限がある一連のキーに追加されます。「パブリック AMI から既存のキーペアを削除する」の詳細情報をご覧ください。

新しいキーペアの作成

キーペア名

mykey

キーペアのダウンロード

STEP 9 インスタンスの作成完了

インスタンスの作成が始まります。しばらく経つと使えるようになります。

aws サービス ▼

作成ステータス

✓

インスタンスは現在作成中です

次のインスタンスの作成が開始されました: i-08d4af54fcb8ce18b 作成ログの表示

!

予想請求額の通知を受け取る

請求アラートの作成 AWS 請求書の予想請求額が設定した金額を超えた場合（つまり、無料利用枠を超えた場合）、メ

インスタンスへの接続方法

インスタンスは作成中です。実行中状態になり、使用する準備ができるまでに数分かかることがあります。新しいインスタンスのたは終了するまで継続します。

[インスタンスの表示] をクリックして、インスタンスのステータスを監視します。インスタンスが一度実行中状態になれば、[インスタンスへの接続方法を詳細はこちら]。

[手順] SSH で EC2 インスタンスへの接続

起動したら、そのEC2インスタンスに接続してUbuntuを操作できるようにします。通常のLinuxマシンへの接続と違って、キーペアが必要です。

STEP ① SSH ソフトの準備

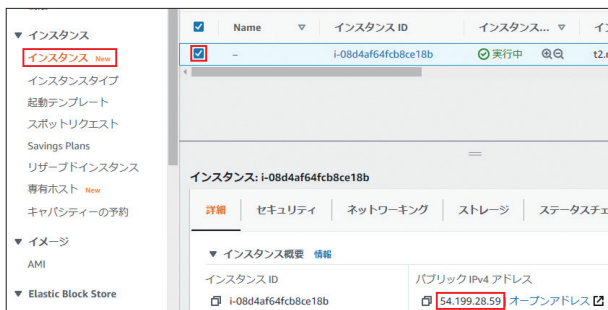
接続には、SSHソフトが必要です。Tera Term や PuTTY などのソフトがあります。ここではTera Termを使います。あらかじめダウンロードしてインストールしておいてください。

- Tera Term

<https://ja.osdn.net/projects/ttssh2/releases/>

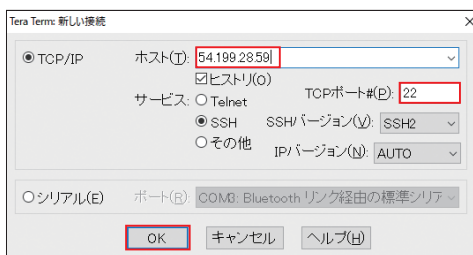
STEP ② IP アドレスを確認する

EC2コンソールの「インスタンス」メニューを開いて、インスタンス一覧を表示します。このなかにも、いま作成したインスタンスがあるので、クリックしてチェックを付けます。すると下に、その情報が表示されます。「パブリックIPv4アドレス」が、IPアドレスなので、この値を控えてください。



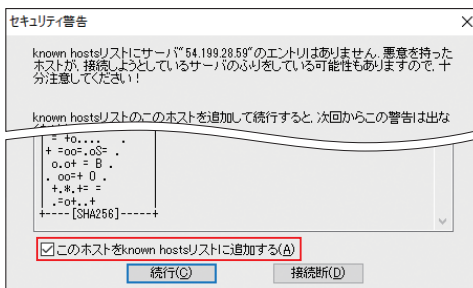
STEP ③ SSH で接続する

Tera Termを起動します。接続先が尋ねられるので、「ホスト」の部分に、いま調べたIPアドレスを入力します。「サービス」は「SSH」、TCPポートは「22」とし、「[OK]」をクリックします。



STEP ④ セキュリティの警告に回答する

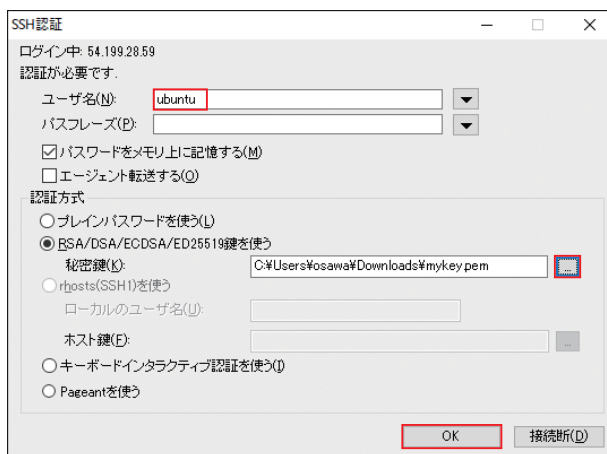
初回に限り、意図する接続先かを確認するための、セキュリティの警告画面が表示されます。[このホストをknown hostsリストに追加する]にチェックを付け、[続行]をクリックします。



STEP 5 ユーザー名を入力し、キーファイルを選ぶ

ユーザー名が尋ねられたら「ubuntu」と入力します。

認証方式で [RSA/DSA/ED25519 鍵を使う] を選択し、[秘密鍵] の横の [...] ボタンをクリックします。するとファイルを尋ねられるので、先ほどダウンロードしておいたキーペアファイル (mykey.pem) を選択します。そして [OK] をクリックします。



これで接続が完了し、リモートでUbuntuが利用できるようになっています。

このあと Docker をインストールしていきますが、その操作については、書籍 P.307 を参照してください。

COLUMN

EC2 インスタンスの停止

Ubuntu の利用が終わったら、Tera Term を閉じるもしくは「exit」と入力しますが、回線が切断されるだけで、仮想マシンは動きっぱなしです。つまり課金の対象です。

停止するには、インスタンス一覧で該当インスタンスを右クリックして [インスタンスを停止] します。すると停止します。停止したインスタンスは、再開することもできます（ただし再開すると IP アドレスが変わります）。

停止中は EC2 インスタンスの料金はかかりませんが、ディスクの料金は依然としてかかります。完全に削除するには、[インスタンスを終了] を選択してください。そうすると、この仮想マシンは完全に削除されます（復活することはできません）。

また、学習が終わったら、速やかにインスタンスなどを削除しましょう。放置しておくと「クラウド破産」という言葉があるくらい、費用がかかることがあります。

