

AWS をはじめよう

mochikoAsTech 著

2018-10-08 版 **mochikoAsTech** 発行

はじめに

2018 年 10 月 mochikoAsTech

この本を手にとってくださったあなた、こんにちは、あるいははじめまして。「AWS をはじめよう」の筆者、mochikoAsTech です。

本著は、前作「DNS をはじめよう」の続きとなっています。「DNS をはじめよう」ではドメインを買って、AWS のアカウントを作り、Route53 をネームサーバとして使うところまでをやってみました。

「AWS をはじめよう」では、そのドメインを使って自分のサイトを作ってみます。AWS でウェブサーバを立てたり、DNS サーバを立てたり、WordPress をインストールしたりして、ブラウザで自分のサイトが見られるところまで一緒に頑張ってやっていきましょう！

え、この本って別の本の続きなの？「DNS をはじめよう」を先に読んだ方がいい？ と思ったあなた、ぜひ「DNS をはじめよう」を先に読んでください。

ケーキのデコレーションはスポンジを焼いてからでないと出来ないように、順を追ってやっていった方が絶対に分かりやすいです。ぜひ、はじめようシリーズ 1 作目の「DNS をはじめよう」を読んでから、「AWS をはじめよう」に帰ってきてください。

想定する読者層

本著は、こんな人に向けて書かれています。

- これからシステムやプログラミングを学ぼうと思っている新人
- ウェブ系で開発や運用をしているアプリケーションエンジニア
- 「インフラがよく分からないこと」にコンプレックスのある人
- 折角ドメインを買ったので自分のブログやポートフォリオサイトを作りたい人
- 自分のドメインでメールアドレスを作りたい人
- AWS や EC2 や ELB や RDS といった単語に興味のある人
- クラウドってなんだろう？ サーバってなんだろう？ という初心者

本著の特徴

本著では前作「DNS をはじめよう」で買ったドメインを使って、実際に WordPress で自分のサイトを作ってみます。手を動かして AWS でサーバを立てたりロードバランサーの設定をしたりしながら学べるので理解がしやすく、インフラ初心者でも安心して読み進められる内容です。

また実際にありがちなトラブルをとり上げて、

- 上手くいかないときは原因をどう調べたらいいのか？
- 問題をどう解決したらいいのか？
- どうしたら事前に避けられるのか？

を解説するとともに、実際にコマンドを叩いて反復学習するためのドリルもついています。

本著のゴール

本著を読み終わると、あなたはこのような状態になっています。

- WordPress でおしゃれなサイトができあがっている
- 使うも壊すも自由な勉強用の Linux サーバ環境が 1 台手に入る
- 「mochiko@startdns.fun」のような自分のメールアドレスでメールを受信できるようになっている
- 読む前より AWS やサーバや黒い画面が怖くなくなっている

免責事項

本著に記載されている内容は筆者の所属する組織の公式見解ではありません。

また本著はできるだけ正確を期すように努めました。が、筆者が内容を保証するものではありません。よって本著の記載内容に基づいて読者が行った行為、及び読者が被った損害について筆者は何ら責任を負うものではありません。

不正確あるいは誤認と思われる箇所がありましたら、電子版については必要に応じて適宜改訂を行いますので筆者までお知らせいただけますと幸いです。

目次

はじめに	3
想定する読者層	3
本書の特徴	4
本書のゴール	4
免責事項	4
第 1 章 インフラとサーバってなに？	7
1.1 AWS を理解するには先ずインフラを知ろう	8
1.2 インフラとは？	8
1.3 サーバとは？	9
1.3.1 オンプレミスとクラウド	11
1.4 AWS 以外のクラウド	13
1.4.1 【ドリル】サンプル	13
第 2 章 AWS とは？	15
2.1 AWS は最初の 1 年無料	15
2.1.1 無料枠に含まれるもの	15
2.1.2 無料の 1 年が終わる前にすべきこと	15
第 3 章 AWS でウェブサーバを立てよう	17
3.1 マネジメントコンソール	18
3.2 IAM	18
3.3 請求アラート	18
3.4 リージョン	18
3.5 CloudTrail	18
3.6 SecurityGroup	18
3.7 VPC	18

3.8	AMI	18
3.9	EC2	18
3.9.1	SSH の鍵認証	18
3.9.2	鍵の変換	18
3.9.3	ElasticIP	18
3.9.4	Bastion	18
3.10	ELB	18
第 4 章	DB サーバを立てよう	19
4.1	RDS	19
第 5 章	ネームサーバの設定をしよう	21
5.1	Route53	21
付録 A	本当の Git	23
A.1	Git - ぎゅつと言えないトゥインクル	24
あとがき		25
Special Thanks:	25
レビューアー	25
参考書	25
著者紹介		27

第 1 章

インフラとサーバってなに？

この章では AWS とはなにか？ そもそもクラウドとは何か？ サーバとは何か？ という基本を学びます。

1.1 AWSを理解するには先ずインフラを知ろう

AWS とはアマゾン ウェブ サービス (Amazon Web Services) の略で、欲しいものをぽちっとな！すると翌日には届くあのアマゾンがやっているクラウドです。

「AWS はアマゾンがやっているクラウドです」と言われても、「クラウド」が分からないと結局 AWS が何なのかよく分からないままですよね。

クラウドって何なのでしょう？

クラウドだけではありません。IT 系で働いていてもクラウドやサーバやインフラは「なんか・・・ふんわり・・・なんか雲の向こう側にある・・・ウェブサイト作るための何か・・・？」という程度の認識で、クラウドってなに？とか、サーバってなに？と聞かれたときに、ちゃんと説明できる人は意外と少ないのではと思います。

なので、先ずは「AWS はアマゾンがやっているクラウド」という文章の意味が分かるよう、インフラ周りから順に学んでいきましょう。

1.2 インフラとは？

インフラという言葉は知っていますか？

はじめて聞いたという人も、なんとなくは分かるけど説明してと言われたらうーん・・・な人も、いま自分が考える「インフラ」についての説明をここに書いてみましょう。(いきなり正解を聞かされるより、自分で答えを考えて書き出してみたらの方が、きっと正解を聞いたときに自分の中へ染み渡ってくるはずです)

インフラとは

では答え合わせをしてみましょう。

インフラとはサーバやネットワークのことです。

そもそも「インフラ」こと「Infrastructure」は、直訳すると基盤や下部構造といった意味です。ですので「生活インフラ」と言う一般的なには上下水道や道路、そしてインターネットなど、生活に欠かすことの出来ない社会基盤のことを指します。

そして技術用語としては、インフラはシステムやサービスの基盤となる「設備」のことをいいます。なので、分かりやすく言うと「インフラとはサーバやネットワークのこと」なのです。

これでもう会社で新卒社員に「インフラってなんですか？」と聞かれても、堂々と「サーバとかネットワークのことだよ」と答えられますね！

でも後輩に「インフラっていうのはサーバやネットワークのことだよ」と教えて、今度は「え、サーバってなんですか？」と聞かれたらどうでしょう？

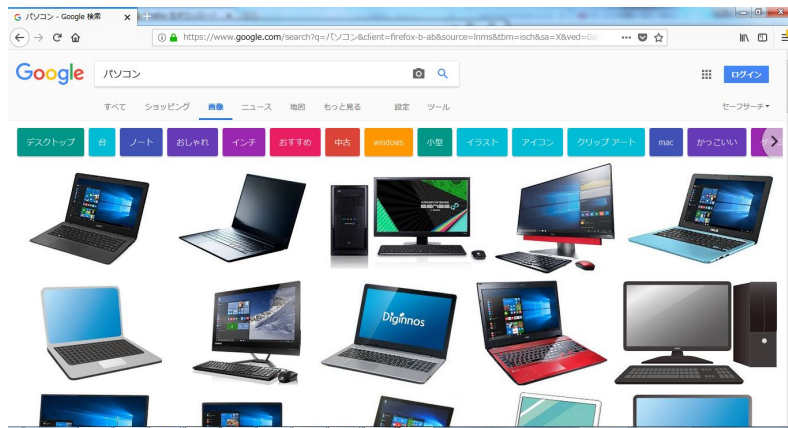
1.3 サーバとは？

後輩から「サーバってなんですか？」という直球の質問を投げつけられたら、しっかりホームランで打ち返せますか？

しっかり答えられるか不安な人も、自信のある人も、ちょっと頭の中でサーバの姿を思い浮かべてみてください。できればイラストじゃなくて実写をお願いします。

思い浮かびますか？ あんまり浮かばないですよね、サーバの姿って。

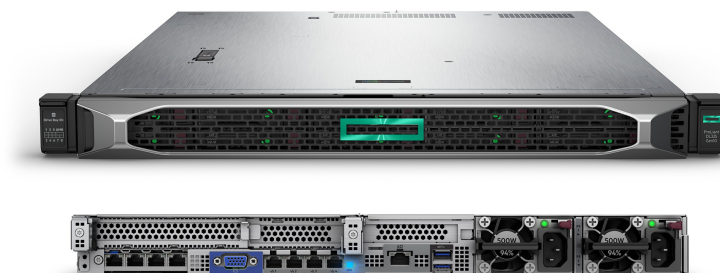
これが「パソコンの姿を思い浮かべてください」なら、きっとすぐに浮かんでくるはずです。(図 1.1)



▲ 図 1.1 パソコンの姿

でもサーバの姿は？ と考えるとなかなか思い浮かびません。ではサーバの姿をお見せ

したいと思います。



▲図 1.2 サーバの姿 (HPE ProLiant DL325 Gen10)

これは Hewlett Packard Enterprise (ヒューレット・パッカード エンタープライズ) が 2018 年 7 月に発売開始した HPE ProLiant DL325 Gen10^{*1} というラックマウント型のサーバで、1 台でおおよそ 50 万円以上します。

本は本棚に収めるように、サーバはサーバラックという専用の棚に収めることが多いです。この HPE ProLiant DL325 Gen10 のようなサーバは、ラック (=棚) にマウントする (=乗せる) ことが出来る形状のため「ラックマウント型サーバ」、略してラックサーバと呼ばれています。

ラックマウント型はサーバによって 1U (ワンユー)、2U、4U のように厚みが異なり、1U ならサーバラックの 1 段分、2U なら 2 段分を使うことになります。そのためラックサーバは 1U サーバという名前でも呼ばれることもあります。サーバラックは 42U サイズが多く、名前のとおり 1U サーバを 42 台収めることができます。^{*2}

「ラックマウント型サーバ」だけでなく、デスクトップパソコンのような「タワー型サーバ」や、シャーシと呼ばれる箱の中に刀のように薄いサーバ (ブレード) を何本も差し込んだ「ブレード型サーバ」など、サーバには色々な形があります。

^{*1} https://h50146.www5.hpe.com/directplus_ent/server/dl325gen10/

^{*2} 但しラックに供給される電源の量や放熱の問題もあるため、実際は 42U サイズのラックにサーバ 42 台をぎちぎちに詰めることは少ないのでは、と思います。

1.3.1 オンプレミスとクラウド

ラックサーバ、タワーサーバ、ブレードサーバのように、手で触れる実体があるサーバのことを「物理サーバ」といいます。物理的な実態があるから物理サーバです。

そもそもですが、人がウェブサイトを作る時には土台となるサーバが必ず必要となります。たとえばてなブログで無料のブログを作ったときでも、Ameba Ownd でホームページを作ったときでも、どこかしらに必ずそのブログやサイトが乗っかっているサーバは存在しています。

昔は企業が「自社のウェブサイト作りたいなあ」と思ったら、DELL がいいかなー、IBM がいいかなー、どのメーカーにしようかなーと見積もりとって、数十万から数百万するので稟議通して、やっと購入。購入後は数週間待って、やっと届いたら段ボールから出して自社のサーバールのラックにがちゃこんとラッキングして同じく自前のネットワーク機器から LAN ケーブル繋いで、電源も繋いで OS のインストールディスクを用意して、サーバに OS インストールして（以下略）・・・と要は自分でサーバを買わないといけませんでした。

なのでお金もいるし、置くための場所もいるし、使い始めるまでに時間もすっごくかかりました。これがいわゆる「オンプレミス」です。

これに対して「クラウド」は、オンプレミスと違ってサーバを買うのではなく、サービスとして「使う」だけなので、

「自社のウェブサイト作りたいなあ」と思ったら、ブラウザ上で使いたいサーバのスペックを選んで、ぼちぼちとするだけで、すぐにサーバを立てることができます。課金は AWS だったら 1 秒単位の従量課金なので、例えば 5 分使ったら 5 分ぶんのお金しかかかりません。

これがクラウド、ないしクラウドサービスと呼ばれるものです。

オンプレミスはサーバを買って使う、クラウドはサービスとして使う、ということですが。でもまだちょっとわかりにくいと思うので、お店を例にしてみましょう。

例えば私が突然お菓子作りに目覚めて、もうコードなんか書いてる場合じゃない！「ケーキ屋さんをはじめたい！」と思ったときに、

土地を買って、そこにケーキ屋さんの建物を建てて、電気とガスと水道を通して床板とか壁紙とか貼って内装も作って・・・からやると、お金も場所も時間もすっごく必要ですよ。

頑張って 1 年かけてやっとオープンしたと思ったら、たった 1 か月で資金が足りなくなってお店がつぶれることになっても、今度は建物の取り壊しとか、土地の処分とか、やるべきことがいっぱいあります。このように全部自分で買って、自分でやるオンプレミス

だと、ちょっと「気軽にケーキ屋さんやってみよう」はなかなか厳しいんですよね。

一方クラウドは、ショッピングモールへの出店に似ています。

ケーキ屋さんをはじめたい！ だからショッピングモールの一区画を借りてやってみよう！

これだと建物はもうあって、電気ガス水道ももう整ってるし私は契約してその中の一区画を使わせてもらうだけなので、オンプレと違ってすぐに始められます。

しかも数か月やってみて、やめたいと思ったら区画を借りるのを止めるだけでいい。さっきも言ったけど AWS は 1 秒単位の従量課金なので、前月の 25 日までにやめるって言わないとダメ、とか契約は 1 年単位、とかがないんです。いつでもやめられます。

簡単に出店できるので、私は本来やりたかったケーキ焼いたりとかそれを売ったりとかの本業だけに注力できます。

さらに、もしお店が大繁盛したら、ショッピングモール内で自分の店を隣の区画も借りてつなげて、広くすることも簡単にできるので、最初から広いところを借りておく必要ありません。

よくクラウドはスモールスタートに向いてる、と言われる理由はまさにこういうところにあるんですね。

但し、長い目で見ればショッピングモールにテナント料を払い続ける方が土地と建物を買うより最終的には高くなるかも知れません。

なので初期投資は少なく済むのですが、クラウドのいいところは、決して「コストが安くなる」ということではなく、例えば万が一ショッピングモールの東館が地震で壊れても、すぐ西棟に移って営業を再開できる、といった冗長性だったりします。

これ自分でやろうとしたら大変ですよ。いつ来るか分からない地震に備えて、最初から 2 か所にお店を構えておくとなったら、相当なコストがかかります。サーバでいったら、ただ自社サイト作りたいだけなのに、東京と大阪の 2 か所にデータセンターを用意して両方に 1 台ずつサーバ置いとかないといけない、みたいな感じですよ。クラウドすごい。

ただし、ショッピングモールにもデメリットはあります。

何かトラブルがあって、モール全体がお休みになるときは、問答無用でお店もお休みになってしまいます。

つまり使っているクラウドで大規模障害が起きたら、利用者である我々に出来ることはなくて復旧までひたすら待つしかない、ということです。

AWS でも大規模障害は定期的に起きています。去年も豪雨による電源障害でサーバがダウンしたりしてました。こうした障害の際も、AWS が発表してくれることがすべてなので、原因が分かるまで自分で徹底的に調べる、ということができません。

それから通路やトイレ、駐車場といった共有スペースは他店舗と共有しているので、

ショッピングモールで他のお店が混んでくると、駐車場が満杯でうちのお店に来たかったお客さんが入れなかったり、人波が自分の店の方まで押し寄せてきたりとマイナスな影響も受けます。

つまり同じクラウドを使っている他のサイトにアクセスが集中すると、回線がひっ迫して、自分のところまでつながりにくくなる、ということですね。

これがクラウドです。

いっぱい喋ったので、もう 1 回おさらいすると、「自社サイト作りたいなあ」と思ったら自分でサーバを買わないといけないのがオンプレミスで、「自社サイト作りたいなあ」と思ったらサーバを従量課金ですぐ使えるのがクラウドです。

そしてようやく最初の話に戻ると、AWS は、Amazon がやっているクラウドのことです。

AWS がなんなのか、お分かりいただけましたでしょうか？

1.4 AWS 以外のクラウド

ところでクラウドの便利さは分かりましたが、クラウドは Google の Google Cloud Platform、Microsoft の Azure (アジュール)、さくらインターネットのクラウドや、GMO クラウド、REX 案件で使っている IDCf クラウド、などなど AWS 以外にもたくさんあります。

その中でもなぜ「AWS」がいい！ のでしょうか？

2017 年時点、クラウド市場では AWS がシェア 34% でトップを独走中です。そのため他のクラウドと比べると、シェア No.1 なので対応できる人も多いし、何か困って調べたときに出てくる情報も多い、というのが AWS を選ぶいちばんの理由です。

あとはサポートもしっかりしてるし、儲かった分だけ投資されて、どんどん改良されていくので、細かな使い勝手もいい・・・そんなところですかね。

1.4.1 【ドリル】 サンプル

問題

問題問題

- A. Route53
- B. お名前.com

答え _____

解答

正解は B です。

第 2 章

AWS とは？

2.1 AWS は最初の 1 年無料

2.1.1 無料枠に含まれるもの

2.1.2 無料の 1 年が終わる前にすべきこと

第 3 章

AWS でウェブサーバを立てよう

3.1 マネジメントコンソール

3.2 IAM

3.3 請求アラート

3.4 リージョン

3.5 CloudTrail

3.6 SecurityGroup

3.7 VPC

3.8 AMI

3.9 EC2

3.9.1 SSH の鍵認証

3.9.2 鍵の変換

3.9.3 ElasticIP

3.9.4 Bastion

3.10 ELB

第 4 章

DB サーバを立てよう

4.1 RDS

第 5 章

ネームサーバの設定をしよう

5.1 Route53

付録 A

本当の Git

またしても何を言っているのかわからないと思いますが、「Git 用語だけでアイドルソングを作って架空のアイドルに歌わせたい」そんな気持ちで作詞をしてしまいました。✂切が厳しい時ほど才能が開花してしまう傾向にある、と言いつをしたいのですが、本著を書くにあたって最初に着手したのがこの付録だったということは Git のコミットログを見れば一目瞭然です。それでは聞いてください。

A.1 Git - ぎゅっと言えないトウインクル

たくさんのライバルから わたしをチェリーピックして
分かれてしまった ふたりのブランチ切ないチェックアウトコミットログを読めば あの
日の気持ちも分かるはず
変わっていく私を ちゃんとプルして追いかけて
きっとわたしは あなたのクローンフォークした日から ずっとあなたを見つめてる
下駄箱に入れたプルリクエスト私の想いをマージして
いま何してるの？ リモートのあなた気付かれないように そっとフェッチ
ステージが上がったら もうコミット逃げられないためらわないで オリジンにプッシュ
して
イニットリモートヘッドリベーススタッシュリバート

あとがき

2018 年 10 月
mochikoAsTech

Special Thanks:

- ブロッコリー好きな茶色い猫に捧ぐ

レビュアー

-

参考書

-

著者紹介

mochiko / @mochikoAsTech

Web 制作会社のシステムエンジニア。モバイルサイトの Web アプリケーションエンジニア、SIer とソーシャルゲームの広報を経て、2013 年よりサーバホスティングサービスの構築と運用を担当。現在は再び Web アプリケーションエンジニアとしてシステム開発に従事。「分からない気持ち」に寄り添える技術者になれるように日々奮闘中。

Hikaru Wakamatsu

表紙デザインを担当。本著の名付け親。

Shinya Nagashio

挿絵デザインを担当。

AWSをはじめよう

2018 年 10 月 8 日 技術書典 5 版 v1.0.0

著 者 mochikoAsTech

デザイン Hikaru Wakamatsu / Shinya Nagashio

発行所 mochikoAsTech

印刷所 日光企画

(C) 2018 mochikoAsTech