

【初心者向けWebinar】 AWSで開発環境を構築しよう

アマゾンデータサービスジャパン株式会社
技術本部 エマージングソリューション部
ソリューションアーキテクト 舟崎健治

アジェンダ

Agenda

- Introduction
- AWS概要おさらい
- 開発環境をAWSで構築するメリット
- 開発環境をAWSで構築するには
- 事例紹介
- まとめ

アジェンダ

Agenda

- Introduction
- AWS概要おさらい
- 開発環境をAWSで構築するメリット
- 開発環境をAWSで構築するには
- 事例紹介
- まとめ

本資料の対象コンテンツについて

- 本資料に含まれるもの
 - 開発環境自体のインフラの構築・管理について
- 本資料に含まれないもの
 - アプリケーションの開発技術や開発手法の詳細について
- 開発者の方が簡単に開発用のサーバー環境を取得するためのソリューションを紹介

開発環境関連の悩み（1）

- 社内サーバーリソースの枯渇
 - アプリの十分なテストができない
 - 機能テストや負荷テストなど
 - 本番環境と同スペックのサーバーを用意できない
 - 開発環境では正常に稼働したのに、本番環境では不具合が起きる等
 - お客様の環境でトラブルが起きた時に、それを再現できる環境がない
 - 同じバージョンのOS, M/W, アプリなど
 - （多くの場合）社内サーバーは共有用途なので自由にさわれない。
 - サービスイン後も検証・開発環境を保持する必要あり

開発環境関連の悩み（2）

- 環境構築・運用に手間およびコストがかかる
 - ラックマウント、ネットワークのケーブルリング、OSやM/Wインストール、アプリのデプロイなど
 - ビルの停電やオフィス引越しに伴うサーバー停止
 - サーバルーム、空調、電源の維持費や運用に必要な人件費等のコスト
- インターネットアクセスのテスト
 - 社内のインターネットアクセス可能なリソースが限られるため、モバイルアクセスのテストができない

**AWSを活用すれば、
これらの悩みを解決できます。**

アジェンダ

Agenda

- Introduction
- AWS概要おさらい
- 開発環境をAWSで構築するメリット
- 開発環境をAWSで構築するには
- 事例紹介
- まとめ

AWSのグローバルインフラストラクチャー

GovCloud	US West		US East	South America	EU	Asia Pacific		
US ITAR Region	N. California	Oregon	N. Virginia	Sao Paulo	Ireland	Sydney	Tokyo	Singapore



AWSの様々なサービス

開発 & 運用管理

モニタリング: CloudWatch

デプロイと自動化: BeanStalk, OpsWork, Cloud Formation, CloudTrail

認証: IAM, Federation

アプリケーションサービス

アプリケーションサービス: SES, SQS, Elastic CloudSearch, Transcoder, SWF, AppStream

モバイル: Cognito, Mobile Analytics, SNS

アプリケーション: WorkSpaces, Zocalo

インフラストラクチャサービス

データベース: RDS, Dynamo, ElastiCache, RedShift

データ分析: EMR, DataPipeline, Kinesis

コンテンツ配信: CloudFront

コンピュータ処理: EC2

ストレージ: S3, EBS, Glacier, Storage Gateway

ネットワーク & ルーティング: VPC, Direct Connect, ELB, Route53

API

サポート: Support

管理: Web Console, Command Line

ライブラリ & SDK: node, Java, JS, python, .net, php, iOS, android

AWS リージョン

AWS アベイラビリティゾーン

Edge Locations

Amazon EC2(Elastic Compute Cloud)

リージョン



任意のゾーンに
分散配置可能



- 数分で起動可能な仮想サーバ
- 1時間ごとの従量課金で利用可能
- スケールアップ/ダウン、アウト/インが即座に可能
 - 約30種類のサーバサイズから選択
- Windows, Linuxなどx86-64アーキテクチャのOS利用可能
 - Windowsライセンスも従量課金
- OS以上はお客様の自由
 - お手持ちのソフトをそのまま利用

AWSは最も汎用性の高いクラウドの一つ



その他多くのソフトウェアが持ち込み(BYOL)可能

ビジネスアプリケーション



アナリティクス



セキュリティ



データ連携



ネットワーク/ストレージ



開発/運用





Amazon RDS(Relational Database Service)

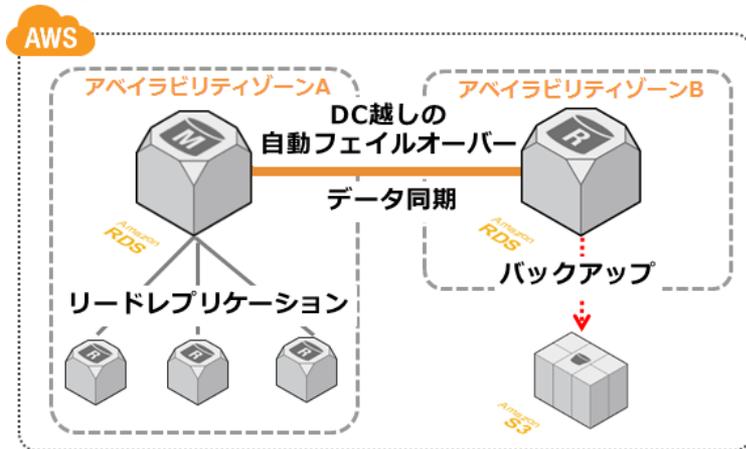
- マネージドRDBMSサービス
- バックアップやフェイルオーバーに対応したDBを数クリックで利用可能
- サポートRDBMS



ORACLE



PostgreSQL



自動
バックアップ

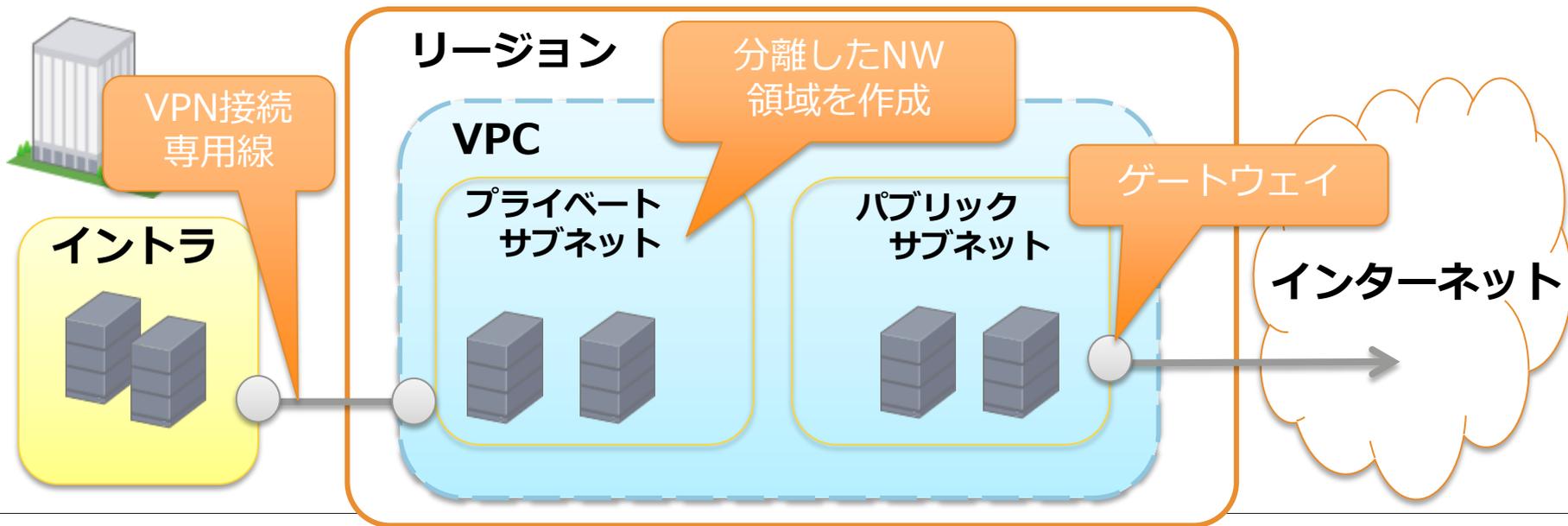
自動フェイル
オーバー

パッチ更新



Amazon VPC(Virtual Private Cloud)

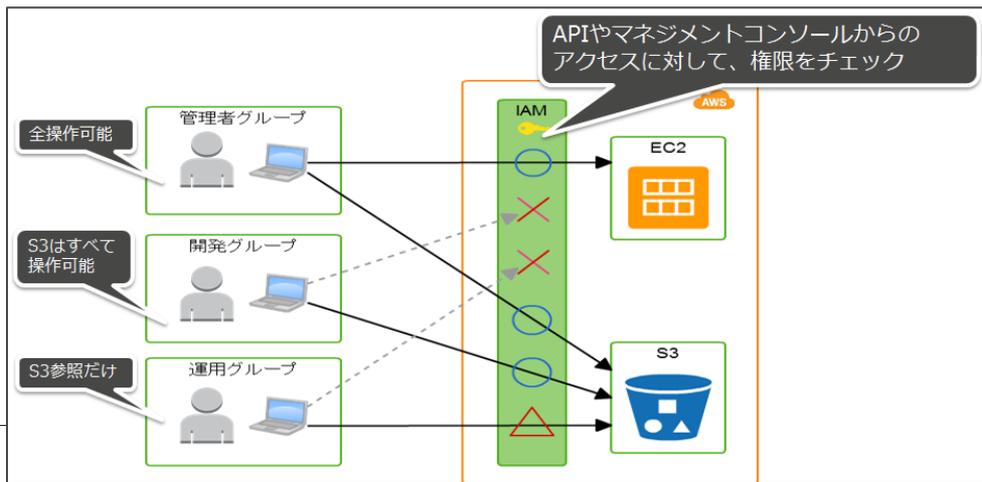
- クラウド内にプライベートネットワークを構築可能
- 企業イントラの延長/1拠点としてAWSを利用





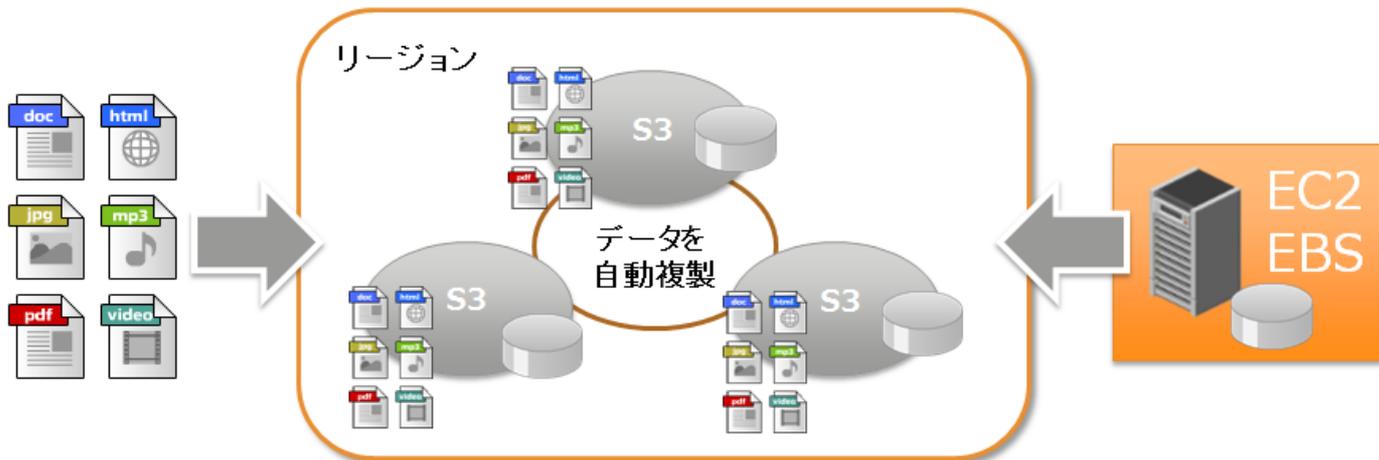
AWS Identity and Access Management (IAM)

- AWS操作をよりセキュアに行うための認証・認可の仕組み
- AWS利用者の認証と、アクセスポリシーを管理
 - AWS操作のためのグループ・ユーザー・ロールの作成が可能
 - グループ、ユーザーごとに、実行出来る操作を規定できる
 - ユーザーごとに認証情報の設定が可能





Amazon S3(Simple Storage Service)



- データを書き込むと、**3つ以上のデータセンター**にデータを自動複製
- 設計上のデータ耐久性は、**99.9999999999%**
- 従量課金 約3.3円/1GB/1ヶ月, 容量は無制限
- EC2/EBSの仮想ディスクバックアップはS3へ保管
- 現在世界中で2.0兆個以上のファイルを格納

アジェンダ

Agenda

- Introduction
- AWS概要おさらい
- 開発環境をAWSで構築するメリット
- 開発環境をAWSで構築するには
- 事例紹介
- まとめ

開発環境をAWSで構築するメリット（1）

- いつでも好きな台数の仮想サーバーを立てることが可能
- 不要な時はサーバーを停止することで、利用料金を最適化可能
- WindowsやLinux上に好きなソフトウェアをインストールして動作確認が可能
 - 開発者一人一人に個別のサーバー環境を配布可能
- OracleやWindows, SQL Serverなどの各種有償ソフトウェアを時間課金ですぐに利用可能



開発スピードの向上が可能

開発環境をAWSで構築するメリット（2）

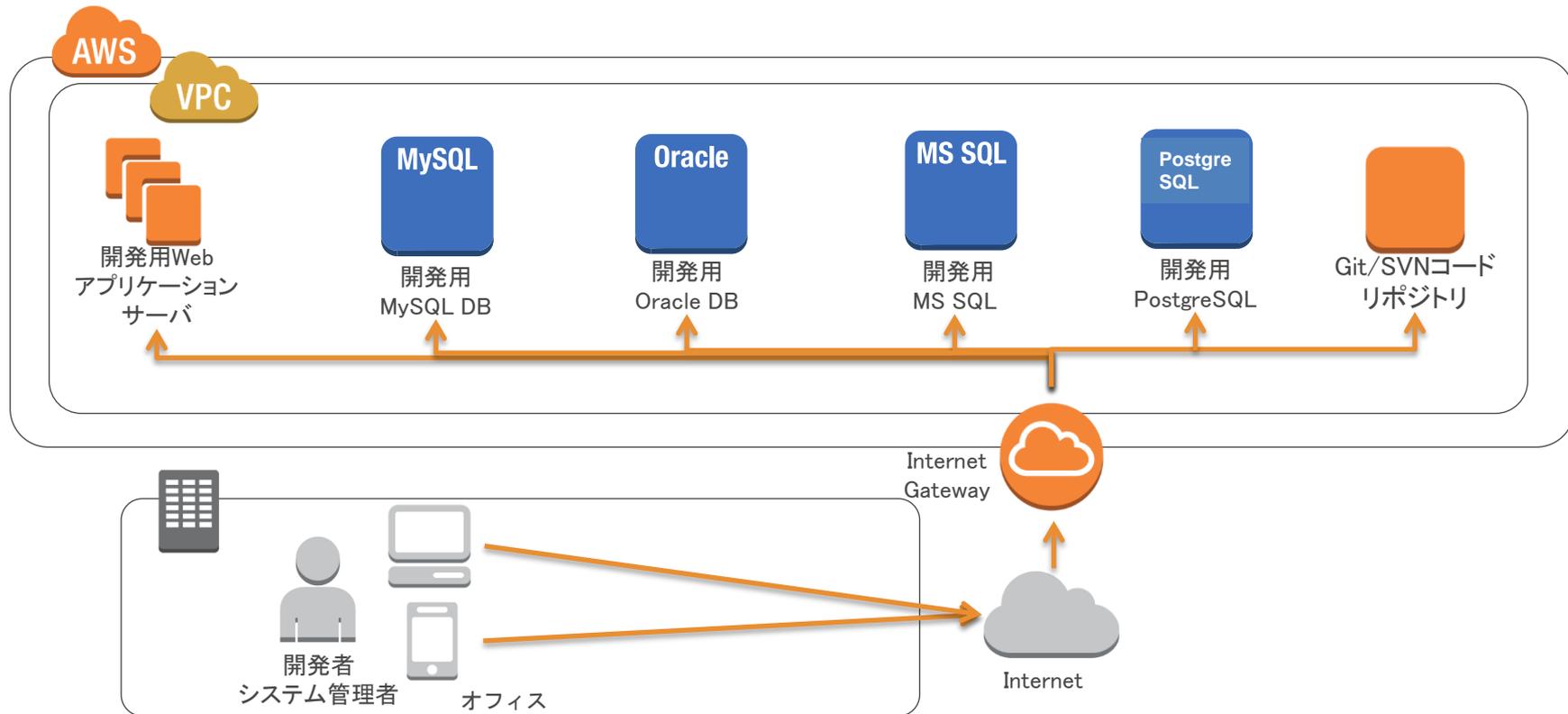
- 一度インストール作業を行った後は、そのサーバーをイメージに保存しておけば、再利用可能。本番への利用も可能
- 社内とVPNで接続することで、よりセキュアに安心して社内のユーザーがAWS上のサーバーと接続が可能
- 保守期限切れによるサーバーの買い替えが不要

アジェンダ

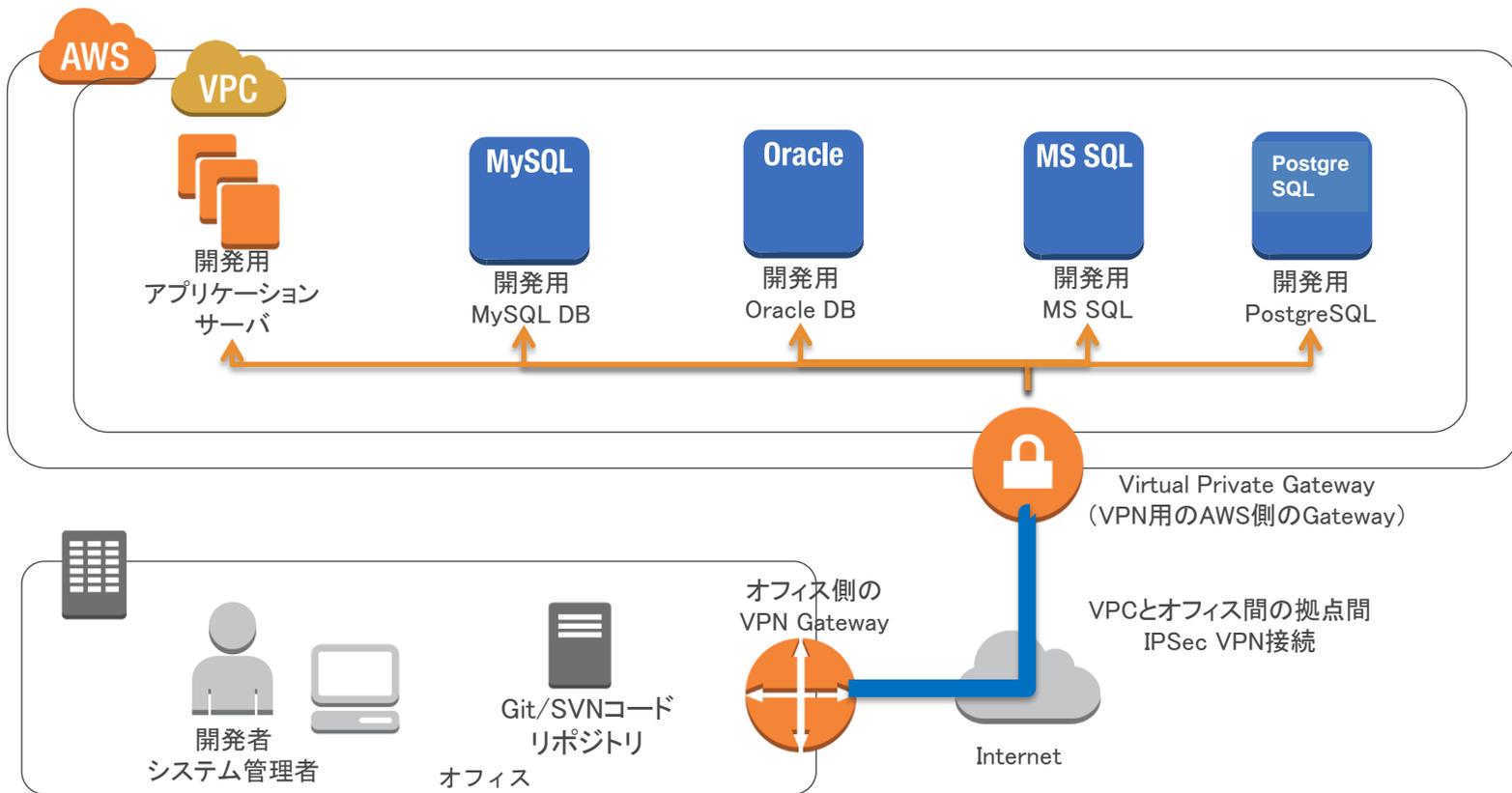
Agenda

- Introduction
- AWS概要おさらい
- 開発環境をAWSで構築するメリット
- 開発環境をAWSで構築するには
- 事例紹介
- まとめ

開発環境用の構成イメージ図：インターネット接続パターン



開発環境用の構成イメージ図：VPN接続パターン



AWSアカウント登録について

- Webフォーム上で数分程度の登録作業をするだけで、すぐさまAWSを利用可能
- AWSアカウント作成の流れ
<http://aws.amazon.com/jp/register-flow/>
- AWSアカウント作成方法についての動画（日本語字幕付き）
<http://aws.amazon.com/jp/getting-started/>

The screenshot shows the 'AWS クラウドのご利用開始方法' (Getting Started with AWS Cloud) page. At the top, there are two buttons: '今すぐ無料でお試し' (Get started for free now) and '導入・資料請求のお問い合わせ' (Contact us for introduction and request for materials). Below this is a paragraph explaining that the account creation, EC2 instance launch, and S3 bucket creation can be completed in 15 minutes. The page is divided into three sections, each with a video player icon and a title:

- AWS アカウント作成方法** (AWS Account Creation Method): This section is highlighted with a red border. It features an illustration of a laptop and a play button icon. Below the icon, it states: 'AWS アカウントにサインアップする方法をご紹介します。この4分間の動画（日本語字幕）をご覧ください。日本のお客様はこちらの「AWS アカウント作成の流れ」もご覧ください。' (We introduce the method to sign up for an AWS account. Please watch this 4-minute video (with Japanese subtitles). Japanese customers, please also watch the 'AWS Account Creation Flow' on the right.)
- 仮想サーバーの起動** (Starting a Virtual Server): This section features an illustration of EC2 instances and a play button icon. Below the icon, it states: 'Amazon EC2 で仮想マシンを起動する方法をご紹介します。この4分間の動画（日本語字幕）をご覧ください。' (We introduce the method to start a virtual machine on Amazon EC2. Please watch this 4-minute video (with Japanese subtitles).)
- メディアファイルの保存** (Saving Media Files): This section features an illustration of a server and a play button icon. Below the icon, it states: 'Amazon S3 でメディアとファイルの保存を始める方法をご紹介します。この3分間の動画（日本語字幕）をご覧ください。' (We introduce the method to start saving media and files on Amazon S3. Please watch this 3-minute video (with Japanese subtitles).)

AWSアカウント登録にあたってのTips

- 無料利用枠をご利用可能
- クレジットカードは個人用・法人用いずれも利用可能
- 登録したメールアドレスやクレジットカードは、後からでも変更が可能
 - 利用開始当初は個人のクレジットカードで登録、毎月经費精算を行い、利用増加に伴い会社のクレジットカードに切り替える、等

AWSを早速使ってみる

- アカウント登録後にAWSのTopページへアクセス
<http://aws.amazon.com/jp/>
- マネージメントコンソールにログインする



ログインが成功すると以下のサービスの一覧画面が表示される

EC2のサービスを選択して、仮想サーバを起動する

The screenshot shows the Amazon Web Services console interface. At the top, there is a navigation bar with the AWS logo, 'Services' dropdown, 'Edit' dropdown, a location selector set to 'Tokyo', and a 'Help' dropdown. Below the navigation bar, the main content area is titled 'Amazon Web Services' and is organized into several columns of service categories. The 'Compute & Networking' category is expanded, and the 'EC2' service is highlighted with a red rectangular box. An orange callout box on the left side of the image contains the text 'EC2のサービスを選択して、仮想サーバを起動する' and has an arrow pointing to the EC2 service icon. Other visible services include Direct Connect, Route 53, VPC, CloudFormation, CloudTrail, CloudWatch, Elastic Beanstalk, IAM, OpsWorks, Trusted Advisor, Data Pipeline, Elastic MapReduce, Kinesis, Cognito, Mobile Analytics, and SNS. On the right side of the console, there are sections for 'Additional Resources' (Getting Started, AWS Console Mobile App, AWS Marketplace), 'Service Health' (All services operating normally), and 'Set Start Page' (Console Home).

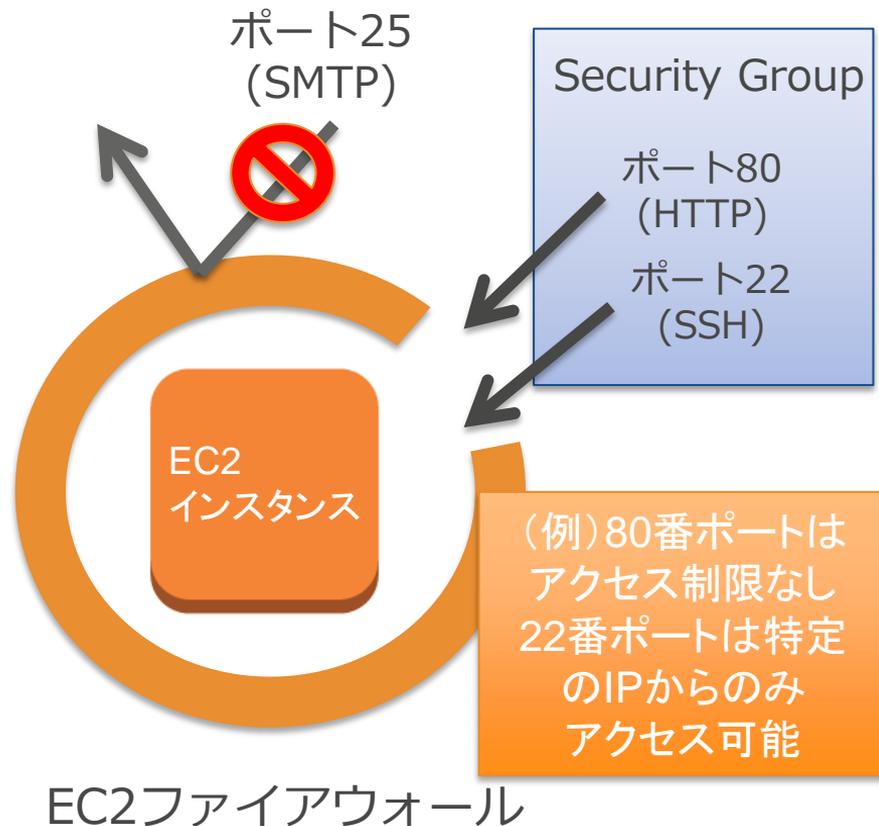
EC2インスタンス（=仮想サーバ）の起動手順



- EC2インスタンス起動の動画
<http://aws.amazon.com/jp/getting-started/>

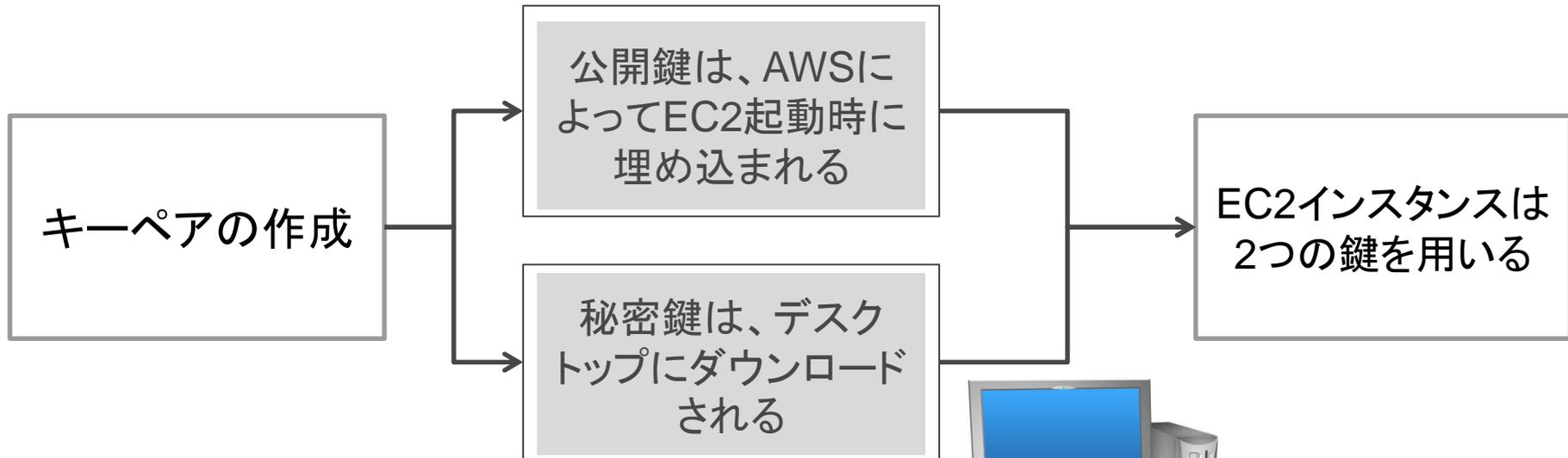
セキュリティグループとは

- インスタンスへのトラフィックを制限するSemi-statefulなファイアウォール機能
- デフォルトでは全トラフィックが閉じており、外からアクセスできない
- 必要な受信アクセスに対してアクセスルールを定義する
 - プロトコル(TCP/UDP)
 - 宛先ポート
 - アクセス元IP / Security Group
- 設定したルール以外の受信トラフィックは破棄される
- ルールをひとまとめにしたテンプレートを Security Groupと呼ぶ



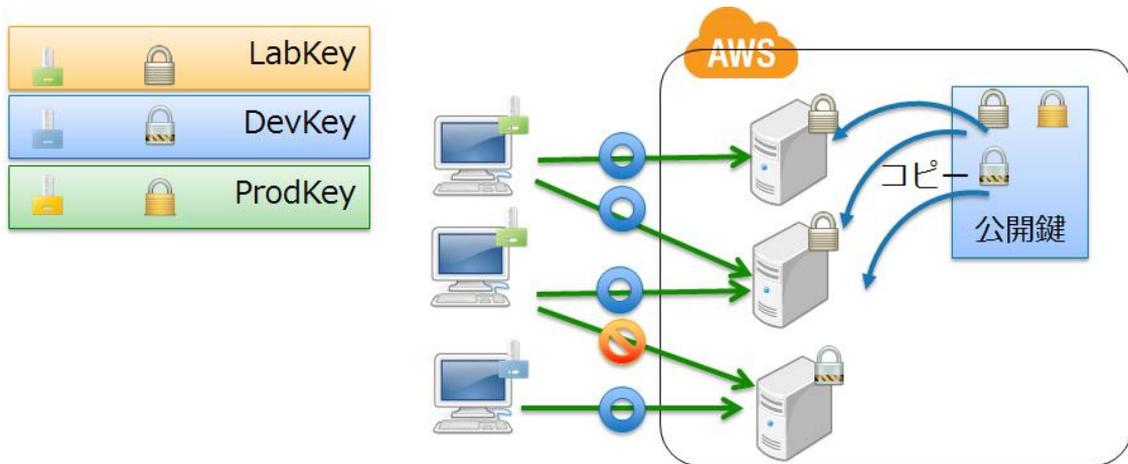
キーペアとは

- キーペアを作成すると、1組の公開鍵・秘密鍵が作成される
- 公開鍵はAWS側で保管。EC2起動時にインスタンスにコピー
- 秘密鍵はユーザー側で保管。ログイン時に必要
- リージョンごとに独立



EC2インスタンスへのログインに必要な鍵認証について

- ユーザ名・パスワードの認証よりも安全な認証方式
- 公開鍵・秘密鍵のペアが合っている場合のみログイン可能
 - Linuxの場合はSSHログイン時に認証
 - Windowsの場合は公開鍵によって暗号化されたAdministratorパスワードを秘密鍵で復号化することで、Administratorログインが可能



EC2インスタンスを起動する

The screenshot displays the AWS Management Console interface for the Asia Pacific (Tokyo) region. The top navigation bar shows the region as 'Tokyo'. The main content area is titled 'Resources' and lists various EC2 resources: 0 Running Instances, 1 Elastic IP, 0 Volumes, 0 Snapshots, 6 Key Pairs, 0 Load Balancers, 0 Placement Groups, and 50 Security Groups. Below this, there is a 'Create Instance' section with a 'Launch Instance' button highlighted in a red box. A callout box on the right side of the image, labeled '①リージョンの選択 Tokyoが選択されていることを確認', shows a dropdown menu for region selection. The 'Asia Pacific (Tokyo)' option is highlighted in a red box within this menu. Another callout box, labeled '②Launch Instanceを選択', points to the 'Launch Instance' button.

①リージョンの選択
Tokyoが選択されていることを確認

②Launch Instanceを選択

Step 1: イメージの選択



Step 1: Choose an Amazon Machine Image (AMI)

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. You can select an AMI provided by AWS, our user community, or the AWS Marketplace; or you can select one of your own AMIs.

Quick Start

- My AMIs
- AWS Marketplace
- Community AMIs
- Free tier only ⓘ

	Amazon Linux AMI 2014.09 (HVM) - ami-35072834 The Amazon Linux AMI is an EBS backed image. It includes the 3.14 kernel, Ruby 2.1, PHP 5.5, PostgreSQL 9.3, Docker 1.2, the AWS command line tools, and repository access to many other packages. Root device type: ebs Virtualization type: hvm
	Red Hat Enterprise Linux 7.0 (HVM) - ami-87206d86 Red Hat Enterprise Linux version 7.0 (HVM), EBS-backed Root device type: ebs Virtualization type: hvm
	SuSE Linux Enterprise Server 11 SP3 (HVM), SSD Volume Type - ami-4d... SuSE Linux Enterprise Server 11 Service Pack 3 (HVM), EBS General Purpose (SSD) automatically during startup for GPU instances. Root device type: ebs Virtualization type: hvm
	Ubuntu Server 14.04 LTS (HVM), SSD Volume Type - ami-0185ac00 Ubuntu Server 14.04 LTS (HVM), EBS General Purpose (SSD) Volume Type. Support (http://www.ubuntu.com/cloud/services). Root device type: ebs Virtualization type: hvm

今回はAmazon Linux、またはMicrosoft Windows Serverを選択する



Step 1: Choose an Amazon Machine Image (AMI)

	Microsoft Windows Server 2012 R2 Base - ami-c9466bc8 Microsoft Windows 2012 R2 Standard edition with 64-bit architecture. [English] Root device type: ebs Virtualization type: hvm	Select
	Microsoft Windows Server 2012 R2 with SQL Server Express - ami-cf416cce Microsoft Windows Server 2012 R2 Standard edition, 64-bit architecture, Microsoft SQL Server 2014 Express edition. [English] Root device type: ebs Virtualization type: hvm	Select
	Microsoft Windows Server 2012 R2 with SQL Server Web - ami-e5416ce4 Microsoft Windows Server 2012 R2 Standard edition, 64-bit architecture, Microsoft SQL Server 2014 Web edition. [English] Root device type: ebs Virtualization type: hvm	Select
	Microsoft Windows Server 2012 R2 with SQL Server Standard - ami-eb416cea Microsoft Windows Server 2012 R2 Standard edition, 64-bit architecture, Microsoft SQL Server 2014 Standard edition. [English] Root device type: ebs Virtualization type: hvm	Select
	Microsoft Windows Server 2012 Base - ami-55715c54	Select

Step 2: インスタンスタイプの選択

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の組み合わせによって構成されている

(例)

- ・t2.micro(無料利用枠): 1vCPU, 1GiBメモリ
- ・r3.8xlarge : 32vCPU, 244GiBメモリ

※スペックの高いインスタンスタイプほど、1時間あたりの起動料金が高くなる

Step 2: Choose an Instance Type

Amazon EC2 provides a wide selection of instance types optimized to fit different use cases. Instance types vary in CPU, memory, storage, and networking capacity, and give you the flexibility to choose the appropriate instance type for your computing needs.

Filter by: All instance types Current generation Show/Hide Columns

Currently selected: t2.micro (Variable ECUs, 1 vCPUs, 2.5 GHz, Intel Xeon Family, 1 GiB memory)

Family	Type	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance
General purpose	t2.micro <small>Free tier eligible</small>	1	1	EBS only	-	Low to Moderate
General purpose	t2.small	1	2	EBS only	-	Low to Moderate
General purpose	t2.medium	2	4	EBS only	-	Low to Moderate
General purpose	m3.medium	1	3.75	1 x 4 (SSD)	-	Moderate
General purpose	m3.large	2	7.5	1 x 32 (SSD)	-	Moderate
General purpose	m3.xlarge	4	15	2 x 40 (SSD)	Yes	High

Cancel Previous Review and Launch Next: Configure Instance Details

t2.microを選択

Step 3: 起動台数およびネットワークの選択

Services Edit Tokyo Help

1. Choose AMI 2. Choose Instance Type 3. Configure Instance 4. Add Storage 5. Tag Instance 6. Configure Security Group 7. Review

Step 3: Configure Instance Details

Configure the instance to suit your requirements. You can launch multiple instances from the same AMI, request Spot Instances to take advantage of the lower pricing, assign an access management role to the instance, and more.

Number of instances ⓘ ← インスタンスの起動台数を指定

Purchasing option ⓘ Request Spot Instances

Network ⓘ [Create new VPC](#)

Subnet ⓘ [Create new subnet](#) ← どのVPCのどのサブネットに起動するかを選択
今回はデフォルトのまま変更不要

Auto-assign Public IP ⓘ

IAM role ⓘ

Shutdown behavior ⓘ

Enable termination protection ⓘ Protect against accidental termination

Monitoring ⓘ Enable CloudWatch detailed monitoring
[Additional charges apply.](#)

Tenancy ⓘ
[Additional charges will apply for dedicated tenancy.](#)

Cancel Previous **Review and Launch**

© 2008 - 2014, Amazon Web Services, Inc. or its affiliates. All rights reserved. [Privacy Policy](#) [Terms of Use](#) [Feedback](#)

Step 4:ストレージの選択

1. Choose AMI 2. Choose Instance Type 3. Configure Instance 4. Add Storage 5. Tag Instance 6. Configure Security Group 7. Review

Step 4: Add Storage

Your instance will be launched with the following storage device settings. You can attach additional EBS volumes and instance store volumes to your instance, or edit the settings of the root volume. You can also attach additional EBS volumes after launching an instance, but not instance store volumes. [Learn more](#) about storage options in Amazon EC2.

Type	Device	Snapshot	Size (GiB)	Volume Type	IOPS	Delete on Termination	Encrypted
Root	/dev/xvda	snap-31c92fd3	8	General Purpose (SSD)	24 / 3000	<input checked="" type="checkbox"/>	Not Encrypted

Add New Volume

Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage. [Learn more](#) about free usage tier eligibility and usage restrictions.

ストレージの容量およびボリュームタイプを選択
起動時にボリュームの容量を増やすことが可能。
(減らすことは不可)
今回はデフォルトのまま次へ

Cancel Previous Review and Launch Next: Tag Instance

Feedback



Step 5:インスタンスへのタグ付け



- 1. Choose AMI
- 2. Choose Instance Type
- 3. Configure Instance
- 4. Add Storage
- 5. Tag Instance
- 6. Configure Security Group
- 7. Review

Step 5: Tag Instance

A tag consists of a case-sensitive key-value pair. For example, you could define a tag with key = Name and value = Webserver. [Learn more](#) about tagging your Amazon EC2 resources.

Key (127 characters maximum)	Value (255 characters maximum)
<input type="text" value="Name"/>	<input type="text" value="Demo"/>

(Up to 10 tags maximum)

Valueにインスタンスを判別するための名前を入力する。
主にAWSマネージメントコンソール上で複数インスタンスを起動したときの判別用
(インスタンス内部の設定とは無関係)

Step 6: セキュリティグループの設定



今回は新しいセキュリティグループを作成する

Step 6: Configure Security Group

A security group is a set of firewall rules that control the traffic for your instance. On this page, you can add rules to allow specific traffic to reach your instance. For example, if you want to set up a web server and allow Internet traffic to reach your instance, add rules that allow unrestricted access to the HTTP and HTTPS ports. You can create a new security group or select from an existing one below. [Learn more](#) about Amazon EC2 security groups.

Assign a security group: Create a new security group

Select an existing security group

Security group name:

Description:

任意のセキュリティグループの名前を入力

Type	Protocol	Port Range	Source
SSH	TCP	22	Anywhere 0.0.0.0/0
HTTP	TCP	80	Anywhere 0.0.0.0/0

Add Rule



Warning

Rules with source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

HTTP(80番)、および
LinuxであればSSH(22番)
WindowsであればRDP(3389番) を許可する

Cancel

Previous

Review and Launch

Step 7: 起動設定の確認

東京リージョンが選択されていることを確認

Services Edit Tokyo

1. Choose AMI 2. Choose Instance Type 3. Configure Instance 4. Add Storage 5. Tag Instance 6. Configure Security Group 7. Review

Step 7: Review Instance Launch

Please review your instance launch details. You can go back to edit changes for each section. Click **Launch** to assign a key pair to your instance.

⚠ Improve your instance's security. Your security group, launch-wizard-8, is open to the world.

Your instance may be accessible from any IP address. We recommend that you update your security group rules to allow access to the application or service you're running, e.g., by adding a rule that allows access from your own IP address. You can also open additional ports in your security group to facilitate access to the application or service you're running, e.g., by adding a rule that allows access on port 80.

▼ AMI Details

Free tier eligible Amazon Linux AMI 2014.09 (HVM) - ami-35072834

The Amazon Linux AMI is an EBS backed image. It includes the 3.14 kernel, Ruby 2.1, PHP 5.5, PostgreSQL 9.3, Docker 1.2, the Amazon CloudWatch Agent, and access to the Amazon Linux repository access to many other packages.

Root Device Type: ebs Virtualization type: hvm

▼ Instance Type [Edit instance type](#)

Instance Type	ECUs	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance
t2.micro	Variable	1	1	EBS only	-	Low to Moderate

▼ Security Groups [Edit security groups](#)

Security group name	Description
launch-wizard-8	launch-wizard-8 created 2014-09-29T18:30:06.698+09:00

Cancel Previous **Launch**

インスタンスタイプがt2.microが選択されていること、セキュリティグループがHTTP、SSH(Linux用)、RDP(Windows用)が許可されていることを確認

Step 8:使用するキーペアの作成または選択をして起動

Instance Launch

Launch details. You can go back to edit changes for each section. Click **Launch** to assign a key pair to your instance and com

Select an existing key pair or create a new key pair

A key pair consists of a **public key** that AWS stores, and a **private key file** that you store. Together, they allow you to connect to your instance securely. For Windows AMIs, the private key file is required to obtain the password used to log into your instance. For Linux AMIs, the private key file allows you to securely SSH into your instance.

Note: The selected key pair will be added to the set of keys authorized for this instance. Learn more about [removing existing key pairs from a public AMI](#).

Create a new key pair

Key pair name
demo11

Download Key Pair

You have to download the **private key file** (*.pem file) before you can continue. Store it in a secure and accessible location. You will not be able to download the file again after it's created.

Cancel Launch Instances

① Create a new key pairを選択
(作成済みであればChoose an existing key pairを選択してLaunch Instancesを選択)

Opening demo11.pem

You have chosen to open:

demo11.pem
which is: Microsoft Excel Comma Separated Values File
from: https://console.aws.amazon.com

What should Firefox do with this file?

Open with

Save File

Do this automatically for files like this from now on.

OK Cancel

② キーペア名を入力

③ 秘密鍵ファイルをダウンロードする

④ Launch Instances
を選択

インスタンスの起動開始

Launch Status

✓ **Your instance is now launching**
The following instance launch has been initiated **i-fa79f3e3** [View launch log](#)

☰ **Get notified of estimated charges**
[Create billing alerts](#) to get an email notification when estimated charges on your AWS bill exceed an amount

起動したインスタンスにインスタンスIDが割り当てられる。インスタンスIDを選択すると、起動ステータスを確認可能

How to connect to your instance

Your instance is launching, and it may take a few minutes until it is in the **running** state, when it will be ready for you to use. Usage hours on your new instance will start immediately and continue to accrue until you stop or terminate your instance.

Click **View Instances** to monitor your instance's status. Once your instance is in the **running** state, you can **connect** to it from the Instances screen. [Find out](#) how to connect to your instance.

Here are some helpful resources to get you started

- [How to connect to your Linux instance](#)
- [Amazon EC2: User Guide](#)
- [Learn about AWS Free Usage Tier](#)
- [Amazon EC2: Discussion Forum](#)

While your instances are launching you can also

[Create status check alarms](#) to be notified when these instances fail status checks. (Additional charges may apply)

[Create and attach additional EBS volumes](#) (Additional charges may apply)

[Manage security groups](#)

[View Instances](#)

[Feedback](#)

インスタンスの起動確認およびインターネットから接続可能なPublic DNSを確認する

The screenshot shows the AWS Management Console interface for an EC2 instance. The instance name is 'Demo', ID is 'i-fa79f3e3', and it is in the 'running' state. The status checks show '2/2 checks passed'. The Public DNS is highlighted with a red box and is 'ec2-54-64-167-130.ap-northeast-1.compute.amazonaws.com'. The instance is located in the 'ap-northeast-1' availability zone.

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm
Demo	i-fa79f3e3	t2.micro	ap-northeast-1c	running	2/2 checks passed	None

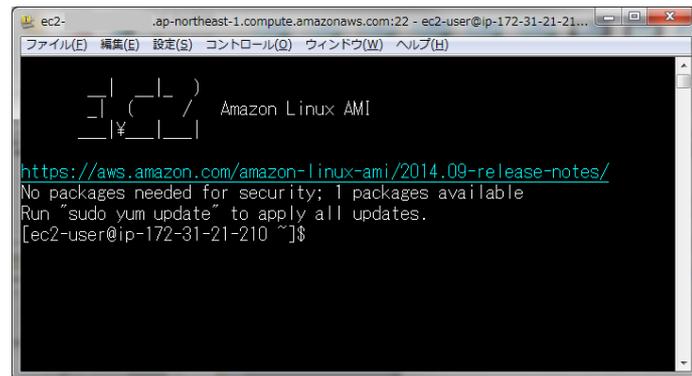
Property	Value
Instance ID	i-fa79f3e3
Instance state	running
Instance type	t2.micro
Private DNS	ip-172-31-21-210.ap-northeast-1.compute.internal
Private IPs	172.31.21.210
Secondary private IPs	
VPC ID	vpc-b4ad65dc
Subnet ID	subnet-b6ad65de
Network interfaces	eth0
Public DNS	ec2-54-64-167-130.ap-northeast-1.compute.amazonaws.com
Public IP	54.64.167.130
Elastic IP	-
Availability zone	ap-northeast-1c
Security groups	launch-wizard-8-...
Scheduled events	No scheduled events
AMI ID	amzn-ami-hvm-2014.09.0.x86_64-ami-35072834
Platform	-
IAM role	-

Instance Stateがrunningになっており、Status Checksが2/2 checks passedになっていれば、アクセス可能

インターネットからアクセス可能なPublic DNS（ホスト名）をコピーして、SSHログインやRDPログイン時の接続先に利用

LinuxインスタンスへSSHログインする

- Windowsからログインする場合
 - TeraTerm (ttssh.exe) を起動
<http://sourceforge.jp/projects/ttssh2/>
 - 接続するインスタンスのホスト名を入力
 - SSH2を指定
 - ユーザ名: **ec2-user**
 - KeyPair作成時に取得した秘密鍵ファイルを選択して接続

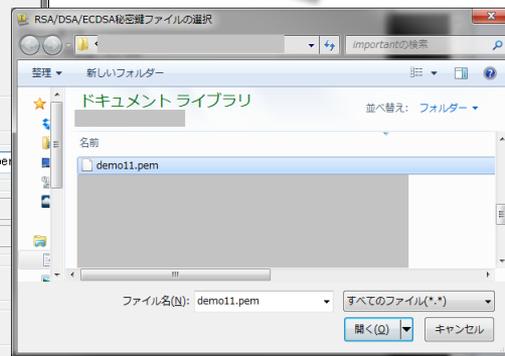
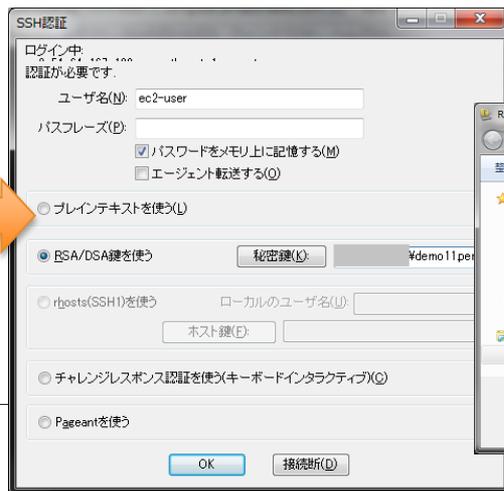


```
ec2- .ap-northeast-1.compute.amazonaws.com:22 - ec2-user@ip-172-31-21-210...
ファイル(F) 編集(E) 設定(S) コントロール(Q) ウィンドウ(W) ヘルプ(H)

  _ | _ | _ |
  | | | | |
  |_|_|_|_|

Amazon Linux AMI

https://aws.amazon.com/amazon-linux-ami/2014.09-release-notes/
No packages needed for security; 1 packages available
Run "sudo yum update" to apply all updates.
[ec2-user@ip-172-31-21-210 ~]$
```



LinuxインスタンスへSSHログインする

- Macからログインする場合
 - Terminalからコマンドラインでログイン(事前にパーミッションを600に!)

```
$ chmod 600 ~/.ssh/keypair_name.pem  
$ ssh -i ~/.ssh/keypair_name.pem ec2-user@インスタンスPublicDNSname
```

WindowsインスタンスへRDPでログインする

The image shows a sequence of steps in the AWS Management Console to retrieve a Windows Administrator password for an EC2 instance. The steps are annotated with numbered callouts:

- ① Windowsインスタンスを選択→Actions→Get Windows Passwordを選択**: In the EC2 console, the 'Demo1' instance is selected, and the 'Get Windows Password' option is chosen from the 'Actions' menu.
- ② 起動時に指定したキーペアに相当する秘密鍵ファイルをローカルから選択**: In the 'Retrieve Default Windows Administrator Password' dialog, the 'Key Pair Path' is set to 'demo.pem'.
- ③ Decrypt Password実行**: The 'Decrypt Password' button is clicked.
- ④ Public IP、ユーザー名およびPasswordを取得。こちらを元にリモートデスクトップを実行**: The retrieved information is shown in a summary box: Public IP, User name (Administrator), and Password.

The final step shows a Windows Server 2012 R2 desktop environment, indicating a successful RDP connection.

LinuxまたはWindowsインスタンスへの接続について

- お客様の環境によっては、社内からインターネットへのSSHまたはRDPの接続が閉じられている可能性がある
 - 社内とAWSのネットワークをIPSec VPNで拠点間接続することで、VPN経由でSSHやRDP接続が可能（専用線接続も可能）
詳細は後述
- モバイルルータやポート利用制限のないGuest Wifi等を使うことで接続
- 起動したインスタンス自体は、VPCの設定でインターネットにルーティングされていれば、インターネット通信が可能

LinuxインスタンスにApache HTTP Serverをインストールしてみる

- Apache HTTP Serverをインストール

```
sudo yum install httpd
```

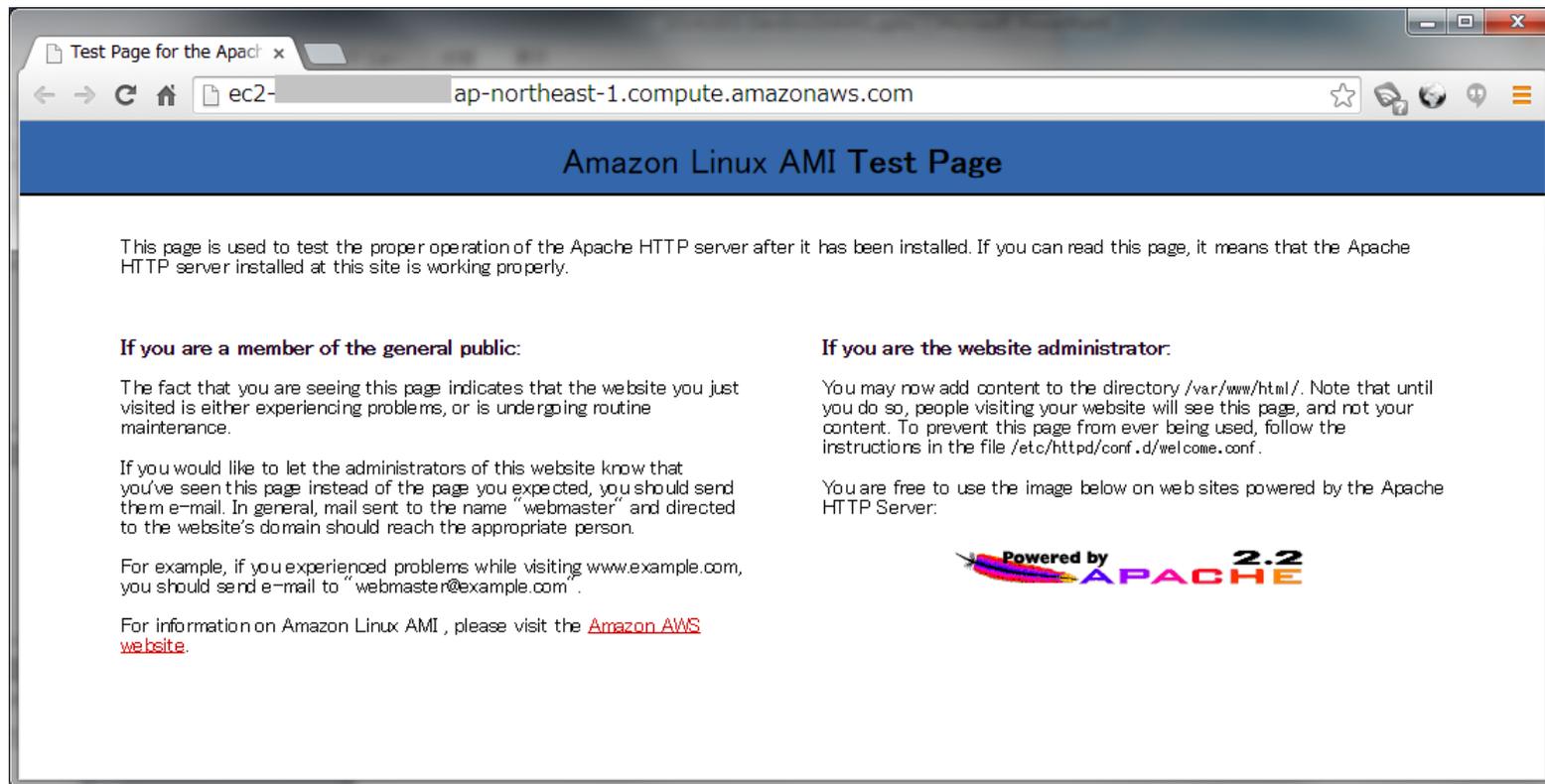
- Apache HTTP Serverを起動

```
sudo service httpd start
```

- (ご参考) インスタンス起動時に常時Apache HTTP Serverが起動するよう設定

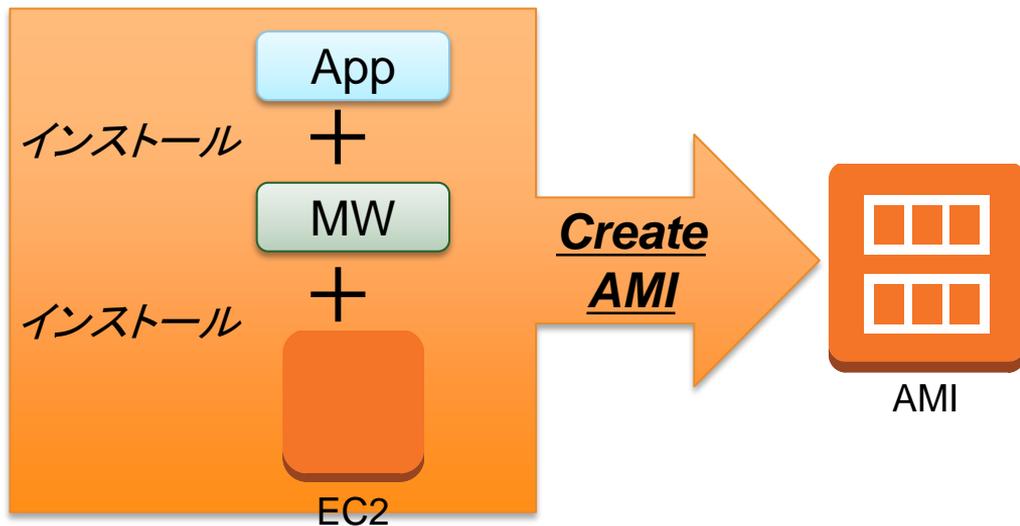
```
sudo chkconfig httpd on
```

Linuxインスタンス上のApache HTTP Serverにアクセスしてみる

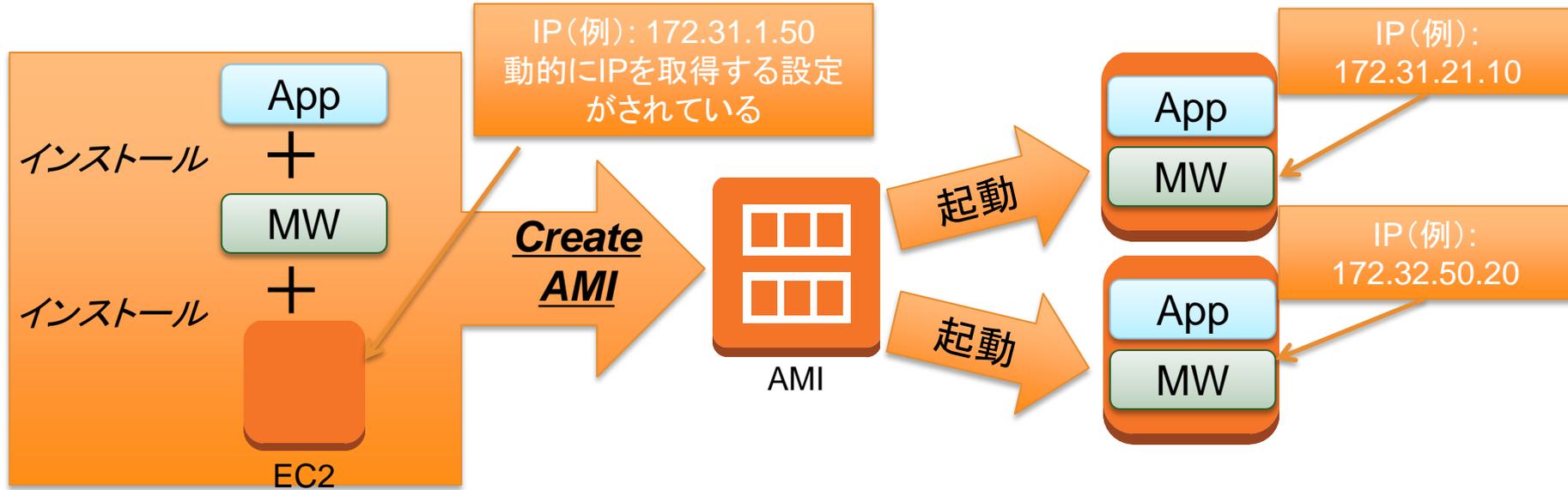


インストール・設定したインスタンスをイメージとして保存する

- アプリケーションやミドルウェアをインストール/設定をした環境をAMI(Amazon Machine Image)としてテンプレート化することが可能

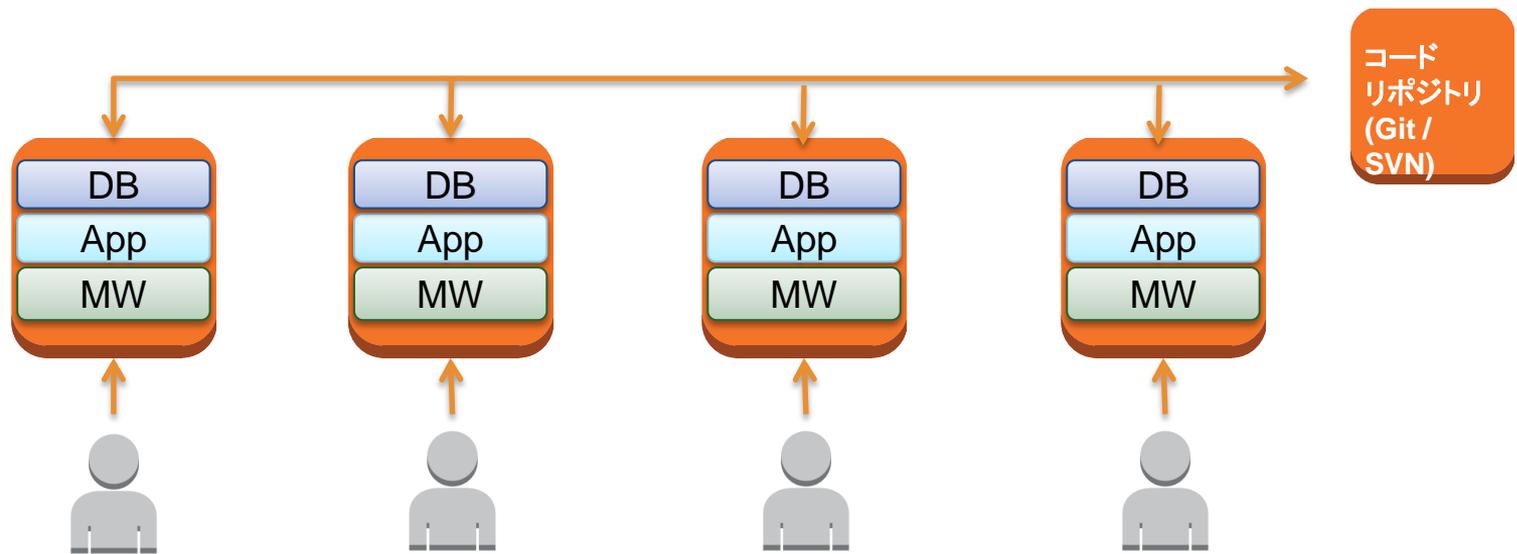


以前構築したものと同一インスタンスをイメージから起動



- IPアドレスは異なるが、同じOSやMW,Appがインストール、設定されたインスタンスを複数起動可能
- VPC内のどのサブネットに起動するかを指定することで、そのネットワークアドレスレンジに基づくIPアドレスが割り当てられる。

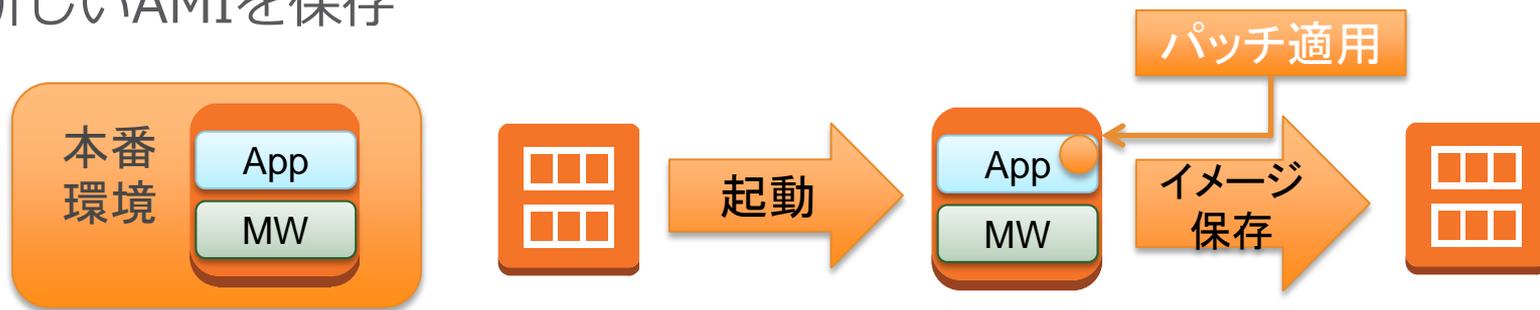
開発者一人一人に同等の開発環境をすぐに配布可能



- t2.microインスタンスを利用、開発時のみ起動することでコストを最適化可能
- 個人専用のインスタンスのため、自由にカスタマイズが可能
- デプロイを自動化することで、さらに開発スピード向上が可能

インスタンスへのパッチ適用について

1. 本番環境とは別に、AMIからインスタンスを起動。パッチを適用して新しいAMIを保存

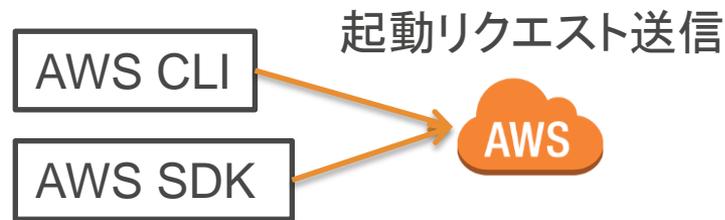


2. 新しいAMIからインスタンスを起動して、こちらを本番環境にする



インスタンスを昼間だけ稼働させて、夜間は停止させる 処理を自動化する

- AWSマネージメントコンソールで実行する操作と同等の処理をSDKやCLIで実行が可能
- LinuxのcronやWindowsのタスクスケジューラ等で上記SDKまたはCLIを含めたバッチスクリプトで自動実行
 - 例：朝の10時に開発用のEC2インスタンスを自動起動、夕方18時に自動終了
- AWS CLI
 - <http://aws.amazon.com/jp/cli/>
- AWS SDK
 - Java / Python / PHP / .NET / Ruby
 - nodeJS / JavaScript / iOS / Android



EC2インスタンスへのファイル転送方法の例

- 標準的な方法
 - scp / sftp / rsync / ftp など
- 高速転送ソリューション
 - TsunamiUDP
 - Aspera
 - Skeeid
 - Riverbed
- S3へファイル転送して、S3からEC2インスタンス側にダウンロード
 - AWS CLIを使って、ローカルとS3間のファイルコピー
 - S3のマルチパートアップロード、マルチパートダウンロード機能を使えば、大きなサイズのファイルを分割して、それらをマルチスレッドで高速転送が可能
- 参考 : AWS Direct Connect
 - 1Gbps/10GbpsでAWSへ専用線接続

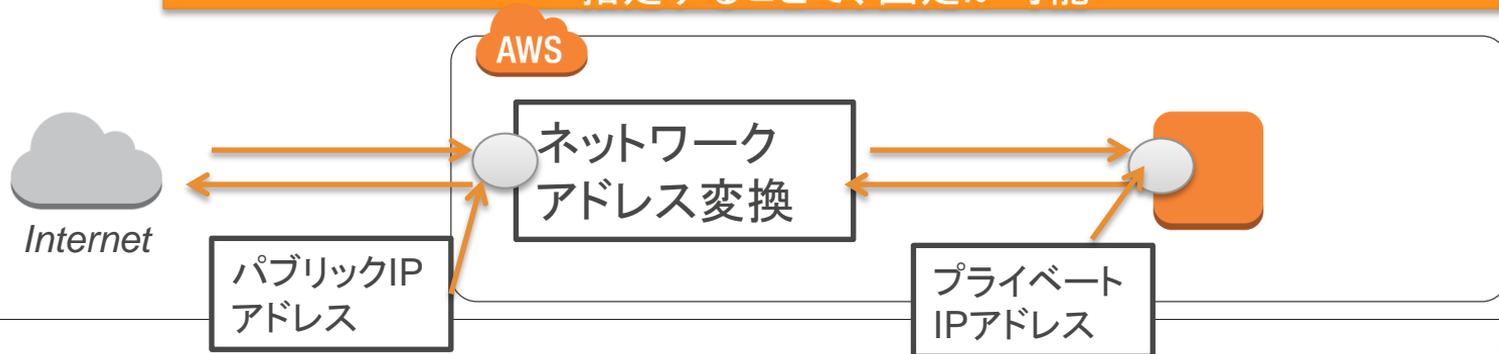
インスタンスで固定IPを使用する方法

- パブリックIPアドレスとプライベートIPアドレスが1:1でマッピングされている。
- インターネットとの通信に使用するパブリックIPアドレスを固定

Elastic IPをインスタンスに割り当てることで、固定が可能

- プライベートIPアドレスを固定

VPC内であれば、インスタンス起動時にプライベートIPを指定することで、固定が可能



社内とVPN接続して、セキュアに社内のユーザーがアクセスできるようにする



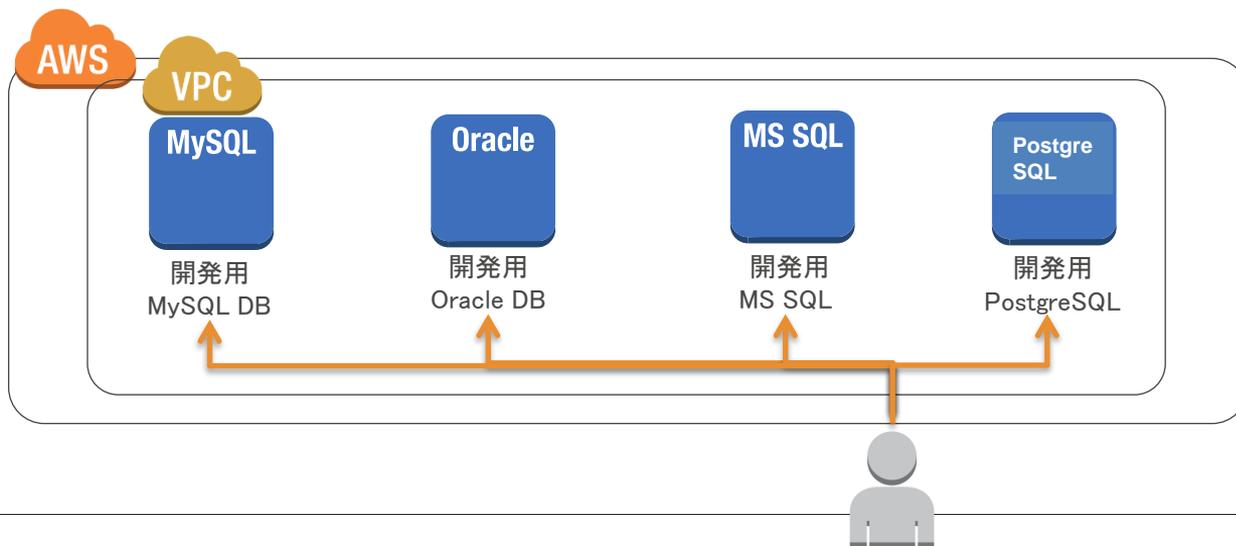
- VPCとのVPN接続の詳細
http://docs.aws.amazon.com/ja_jp/AmazonVPC/latest/UserGuide/VPC_VPN.html
- VPCとの接続で動作すると知られているVPNルーターについて
<http://aws.amazon.com/jp/vpc/faqs/>

社内ユーザーのAWSマネージメントコンソールやAPI, CLIの操作の制限をつける

- AWS管理者アカウントは、AWSアカウント内のすべての操作が可能のため、出来る限り使用しない
- IAM(Identity Access Management)を利用することで、ユーザーごとのアクセスコントロールが可能
 - 例：退職される社員のアクセス権限を簡単に削除可能

開発・検証用RDBをRDSを使ってすぐさま構築

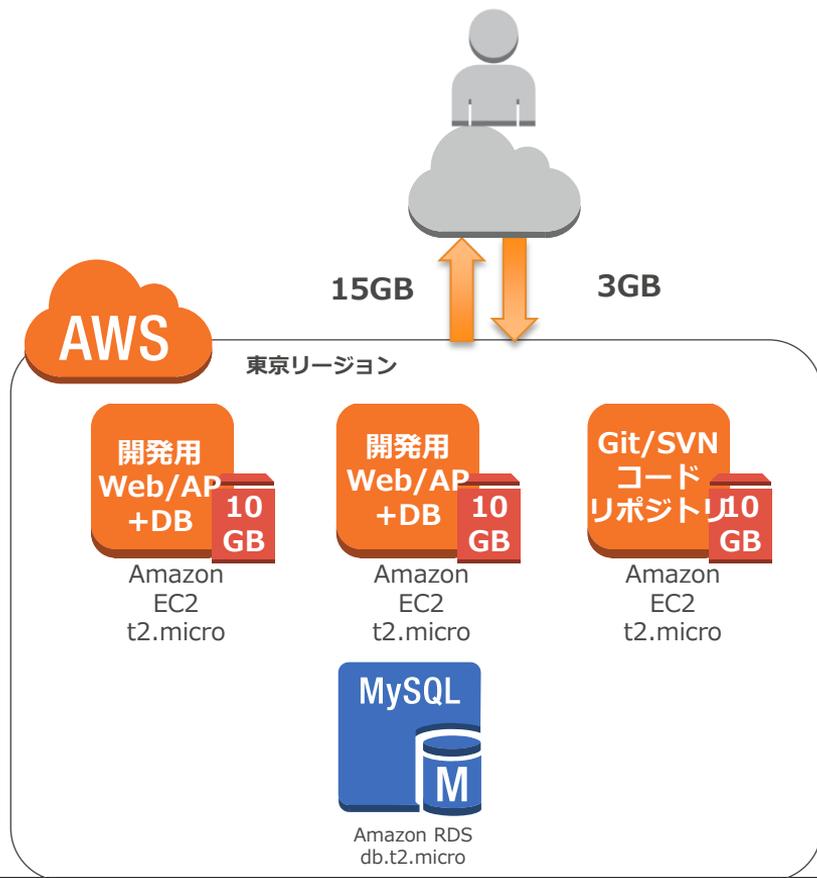
- RDBMSのインストール不要
- RDSを起動後すぐに利用可能。不要になれば削除も可能
- 有償のRDBも従量課金で利用が可能
- パブリックアクセスまたはプライベートアクセスの設定が可能



AWS上で開発環境を構築したときのコストについて

- AWSの利用料金は従量課金
- 以下のように利用量に応じて2パターンの環境にかかるコスト例を紹介する
 - パターンA：無料利用枠内で利用
 - パターンB：開発者10名で個別インスタンスを利用

パターンA：無料利用枠内で利用



構成のポイント

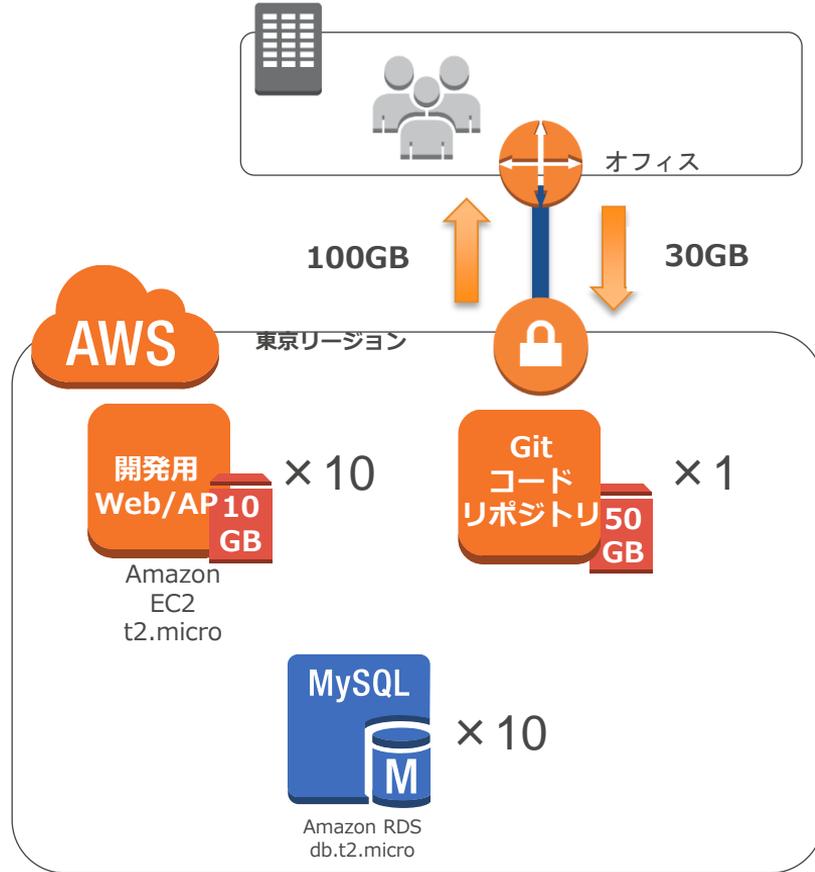
- 開発用Web/AP+DBインスタンスは8時間/日のみ動作
(3台合わせて月1台常時稼働分の時間)
- RDSは常時稼働
- AWSへのアップロードは無料
- AWSからのダウンロードは15GB/月まで無料
- EBSは30GB/月まで無料
- RDSのストレージは20GB/月まで無料
- EC2は全てt2.microを使用
- RDSはdb.t2.microを使用

AWSアカウント取得後、12か月間（無料利用枠適用）
月額：\$ 0

※実際の見積り内訳については以下参照

http://calculator.s3.amazonaws.com/index.html?lng=ja_JP#r=NRT&key=calc-5EF37058-30D7-41FD-AD79-20C596EBD815

パターンB：開発者10名で個別EC2インスタンスを利用



構成のポイント

- 開発用Web/APインスタンスは10台常時稼働
- RDSは10台常時稼働
- AWSへのアップロードは無料
- AWSからのダウンロードは15GB/月まで無料
- EBSは10GB x 10 = 100GB/月
- RDSのストレージは10GB x 10 = 100GB/月
- EC2は全てt2.microを使用
- RDSはdb.t2.microを使用
- 1VPN常時接続

月額：※実際の価格については以下参照

http://calculator.s3.amazonaws.com/index.html?lng=ja_JP#r=NRT&key=calc-F95215A9-8BDE-4A9D-8E66-77CF4F0938D1

(ご参考) デプロイやプロビジョニングの自動化

- ビジネスは変化するもの
 - 顧客とビジネスの間のフィードバックループを加速
 - 利用者の反応やシステムの安定性を見ながら小規模単位でリリース
 - 繰り返し高頻度でデプロイするには自動化が必要！！
-
- Provisioning & Deploy on AWS
http://adsj-contents.s3.amazonaws.com/solution-casestudy/20140702_AWS-BlackBelt-Deploy-public.pdf

アジェンダ

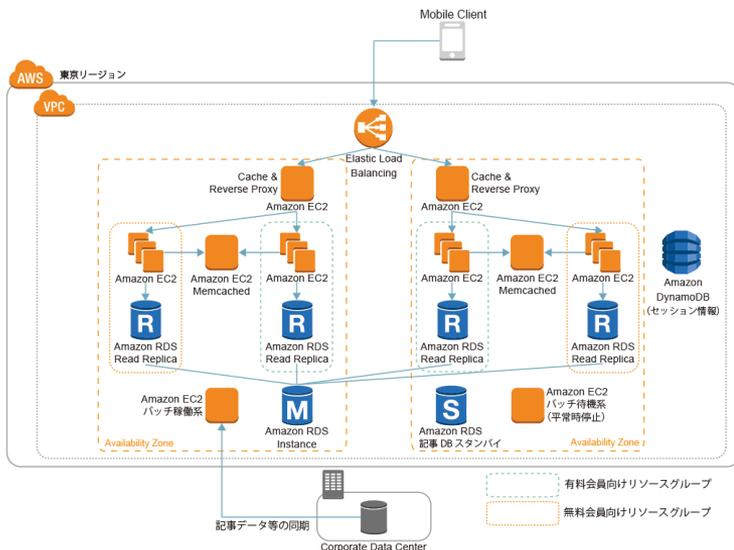
Agenda

- Introduction
- AWS概要おさらい
- 開発環境をAWSで構築するメリット
- 開発環境をAWSで構築するには
- **事例紹介**
- まとめ

AWS導入事例：株式会社日本経済新聞社

～簡単にシステム構築ができるという意味で時間とコストの削減を実現できる点がAWSの最大のメリットと感じています～

- AWS上で「日経電子版モバイル」の開発、サービスインにAWSを利用
- システム運用負荷の軽減、システム構築



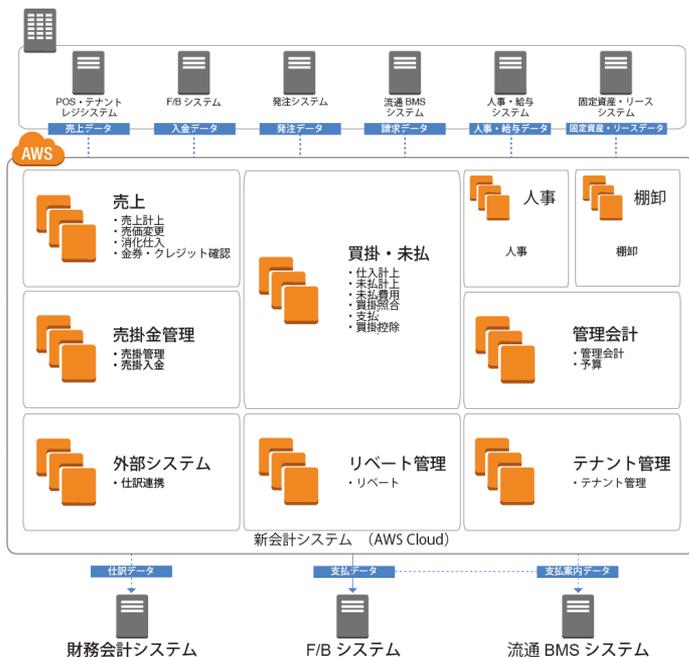
株式会社日本経済新聞社様
システム構成図

リンク

<http://aws.amazon.com/jp/solutions/case-studies/nikkei/>

AWS導入事例：西鉄ストア株式会社

- AWSを会計システムの開発環境として利用
- 売上締め処理や債券計上・改修処理などの業務のためのシステムが、本番・開発環境ともにAWS上で稼働



西鉄ストア株式会社様
システム構成図

リンク

<http://aws.amazon.com/jp/solutions/case-studies/nishitetsu-store/>

アジェンダ

Agenda

- Introduction
- AWS概要おさらい
- 開発環境をAWSで構築するメリット
- 開発環境をAWSで構築するには
- 事例紹介
- まとめ

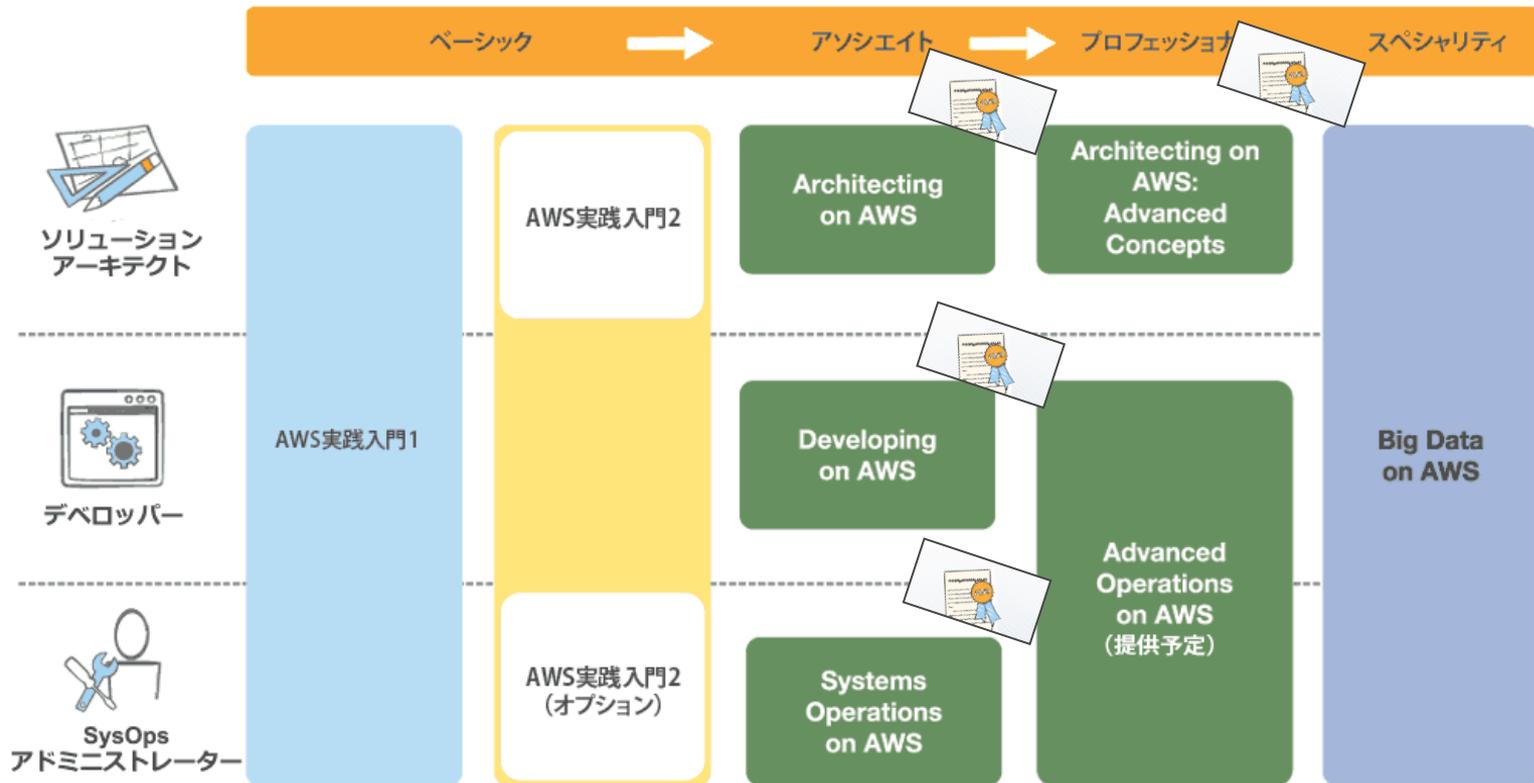
まとめ

- 今まではサーバー枯渇のため開発者は共有環境で開発することが多かった
- AWS上で開発環境を構築することで、開発者1人1人が個別にインスタンスを利用できるので、開発スピードの向上が可能
- AWSは従量課金のため、使わない時は削除することで、料金を最適化可能
- 柔軟にインスタンスを追加、スペック変更が可能のため、アプリの機能テストや速度テストがしやすい
- 1度構築した環境をイメージとして保存、いつでも起動できるため、特定のバージョンのアプリ、あるいは本番環境と同等の環境をいつでも利用・再現が可能
 - トラブルシュート用途などに便利

参照リンク

- AWSアカウント作成の流れ
<http://aws.amazon.com/jp/register-flow/>
- AWSアカウント作成やEC2インスタンス起動方法についての動画
(日本語字幕付き)
<http://aws.amazon.com/jp/getting-started/>
- AWSクラウド活用資料集
<http://aws.amazon.com/jp/aws-jp-introduction/>
- 国内のお客様のAWS活用事例
<http://aws.amazon.com/jp/solutions/case-studies-jp/>

AWSをより深く理解したい方向けに クラスルームトレーニングを提供しています。



詳細: aws.amazon.com/training



認定資格試験