

T3-2

## デモで理解する！ 基本の Web システムアーキテクチャ - スケーラビリティ向上編 -

小林 大樹

アマゾン ウェブ サービス ジャパン合同会社  
ソリューションアーキテクト

# 自己紹介



小林 大樹 (こばやし だいき)

ソリューションアーキテクト

- 業種・業態を問わず、様々なお客様の構成検討を支援
- 前職は金融機関にて、システムの開発や維持保守に従事

# 本セッションで取り上げる AWS のサービス・機能

本セッションで取り上げる AWS のサービス・機能は以下の通りです。  
詳細は、AWS クラウドサービス活用資料集 よりご確認ください。

- Elastic Load Balancing
- Amazon EC2
- Amazon EC2 Auto Scaling
- Amazon Relational Database Service ( RDS )
- Amazon Virtual Private Cloud ( VPC )
- AWS Systems Manager

## AWS クラウドサービス活用資料集

アマゾン ウェブ サービスの公式イベントのアーカイブおよびオンデマンドコンテンツの動画や資料がご利用いただけます。

[AWS Webinar お申込 »](#)

[AWS 初心者向け »](#)

[サービス別資料 »](#)

[ハンズオン資料 »](#)

<https://aws.amazon.com/jp/events/aws-event-resource/>



# 本セッションの対象になる方

- これから AWS 上にシステムの構築を検討される方
- システムの可用性やスケーラビリティの向上について検討される方

可用性って？  
スケーラビリティって？

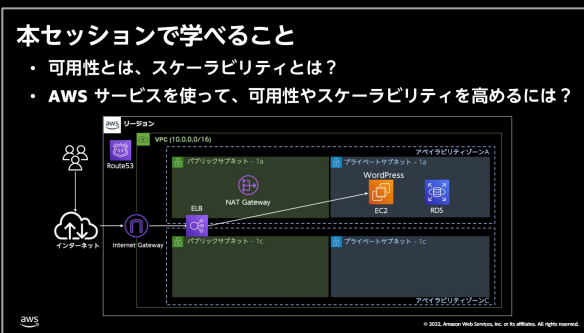


障害に強いシステム  
を構築していきたい

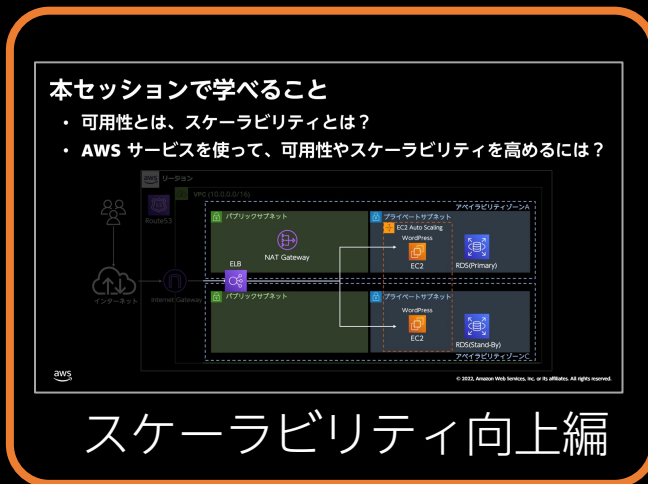


# 関連セッションについて

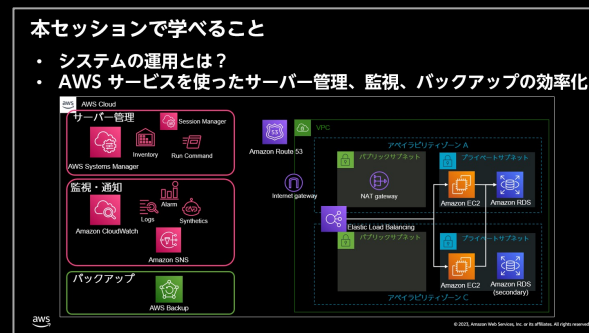
- 本セッションは、基礎編、スケーラビリティ向上編、運用編、セキュリティ編の4セッションで構成しています。
- スケーラビリティ向上編で例示するアーキテクチャの構築デモは、基礎編をご覧ください



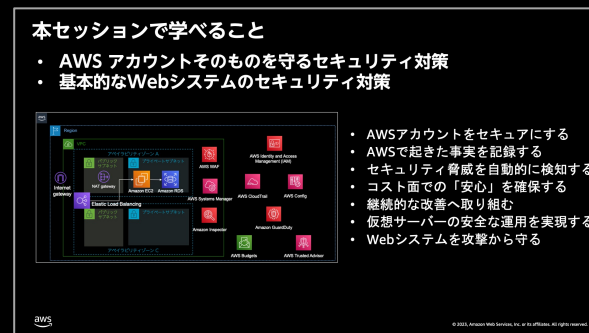
基礎編



スケーラビリティ向上編



運用編



セキュリティ編

※ 当日ご覧になれなかった場合でも、後日のオンデマンド配信でご覧頂くことができます

# 本セッションの内容

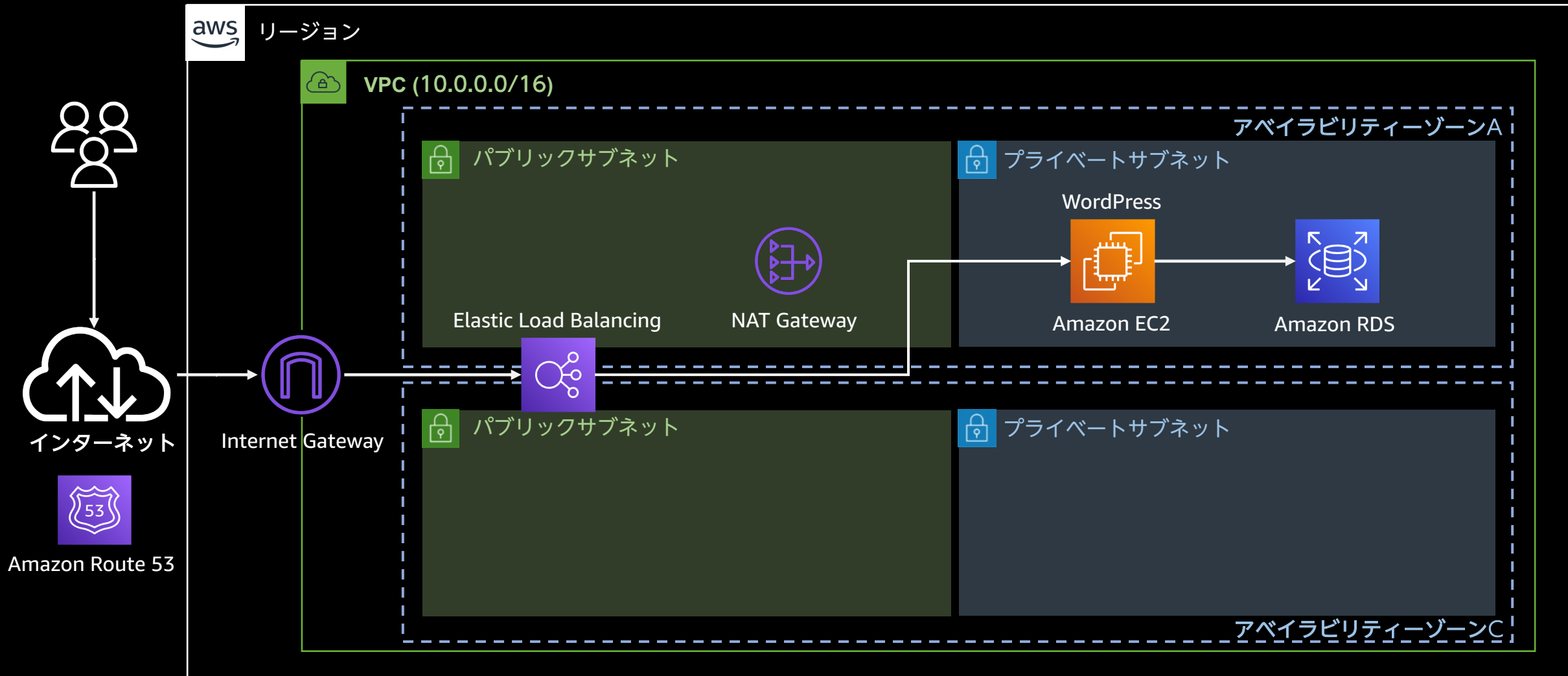
お話しすること

- 基本の Web システムアーキテクチャの振り返り
- 可用性、スケーラビリティとは何か
- Web システムの可用性/スケーラビリティを高める方法
  - EC2 / RDS の冗長化
  - EC2 Auto Scaling によるスケーラビリティの確保
- デモ

※全体像を理解していただくことが目的のため、各サービスや機能の詳細説明は行っておりません

# 基本の Web システム アーキテクチャの振り返り

# 基本の Web システムのアーキテクチャ





# 可用性とスケーラビリティ

## 可用性

システム全体が障害で  
停止することなく、  
サービスを提供し続け  
ることができる能力

## スケーラビリティ

システムの規模の変化  
に柔軟に対応すること  
ができる能力

# 可用性とスケーラビリティ

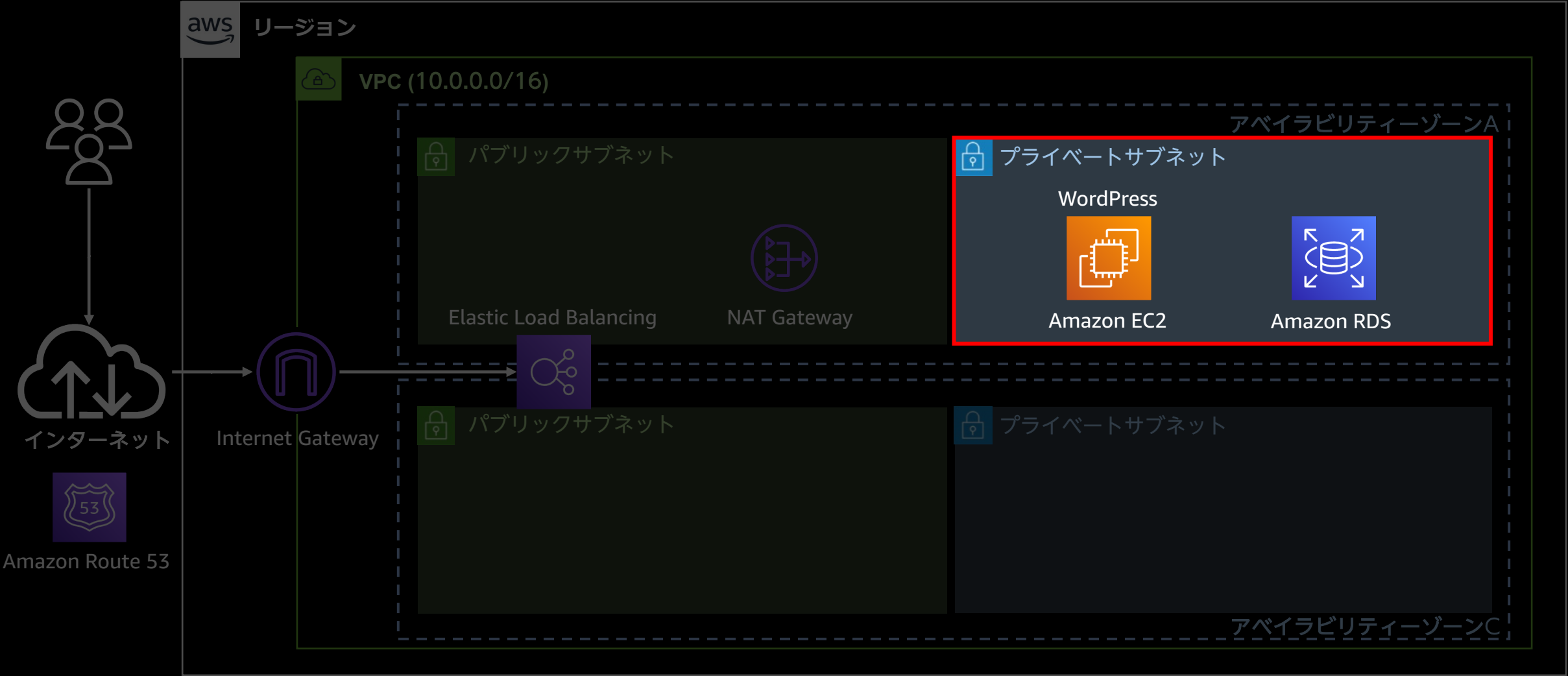
## 可用性

システム全体が障害で  
停止することなく、  
サービスを提供し続け  
ることができる能力

## スケーラビリティ

システムの規模の変化  
に柔軟に対応すること  
ができる能力

# 基本の Web システムのアーキテクチャ



# 基本の Web システムのアーキテクチャ



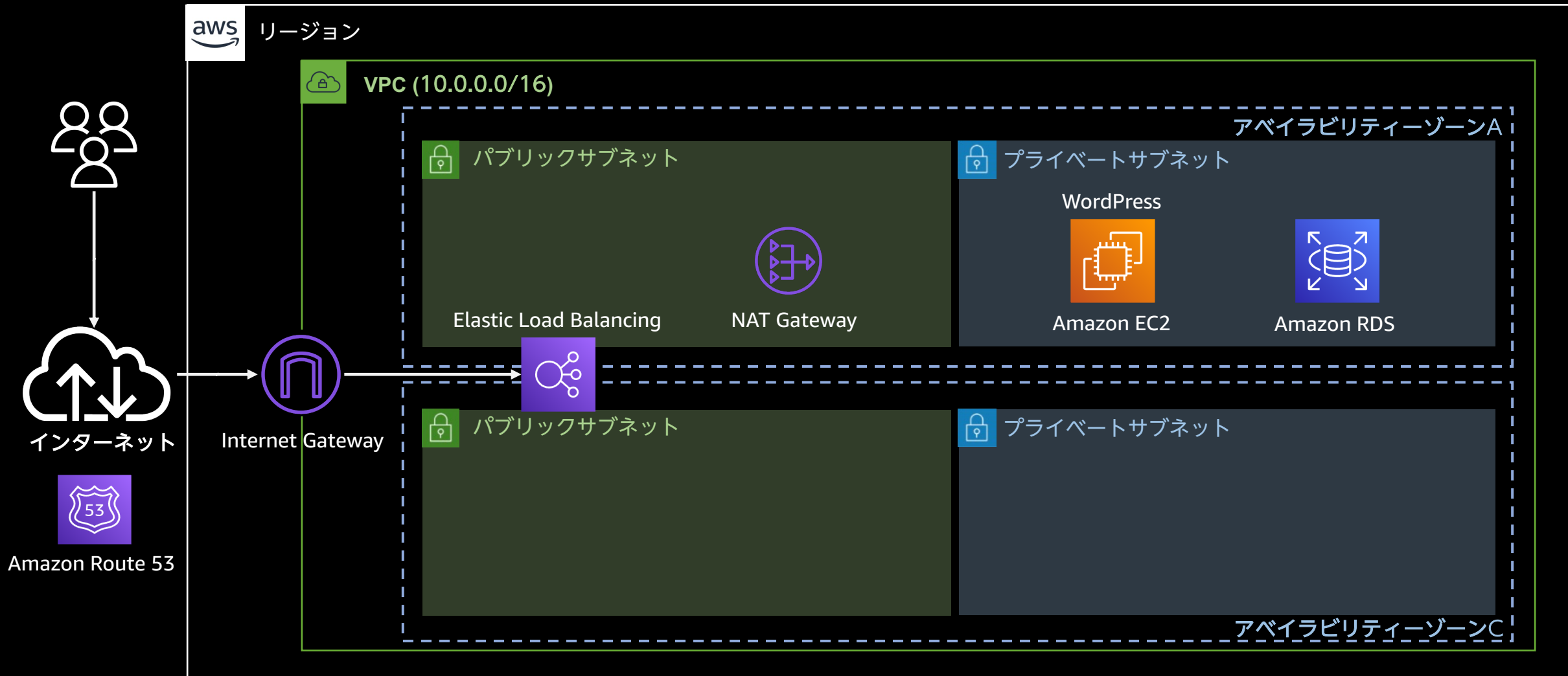
# Design for Failure

障害が発生したとしても、  
システム全体が動き続けるように設計する

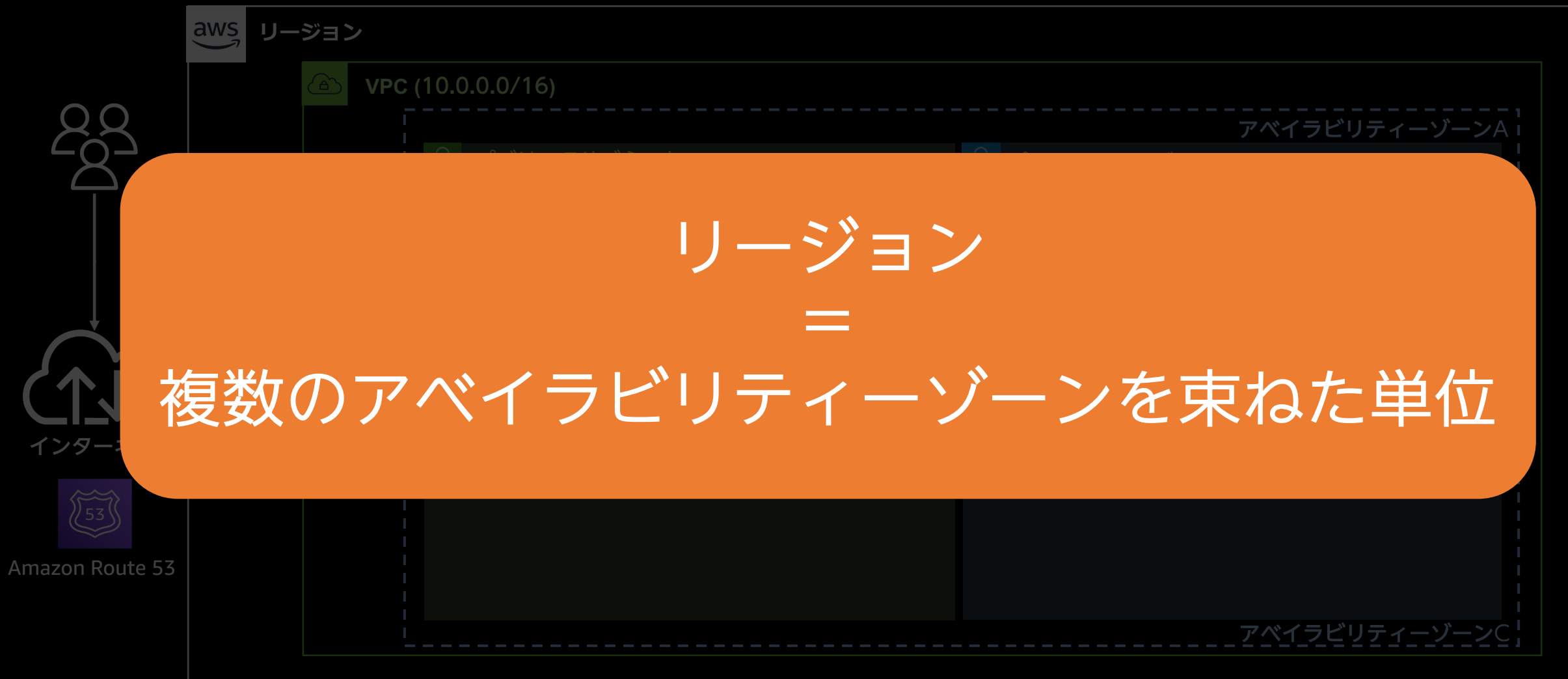
障害が発生させない = ×

障害が発生しても問題ない = ○

# 基本の Web システムのアーキテクチャ

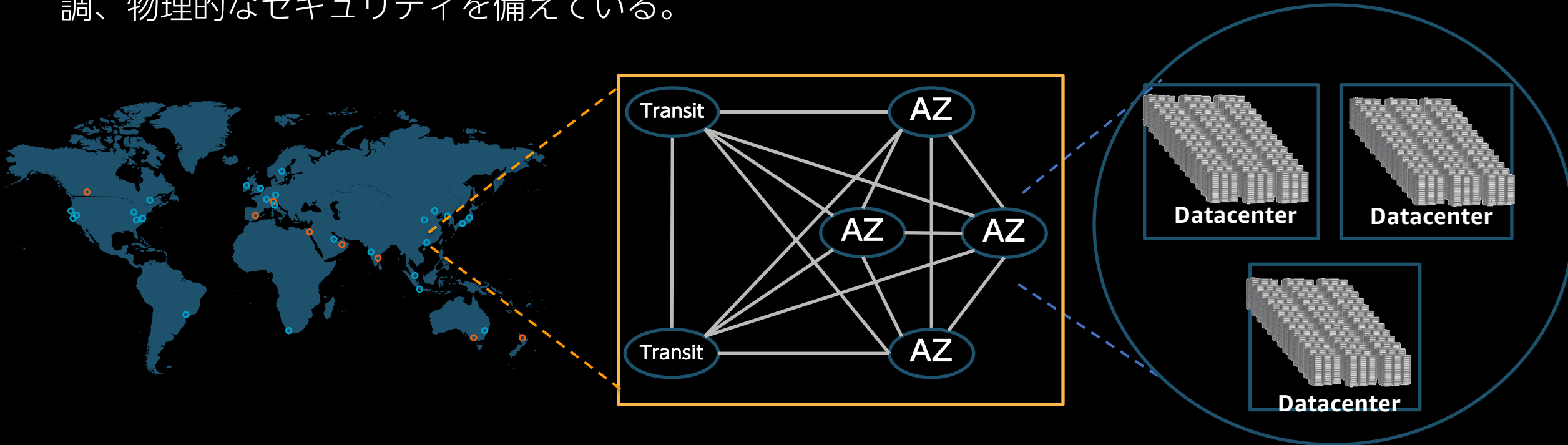


# 基本の Web システムのアーキテクチャ



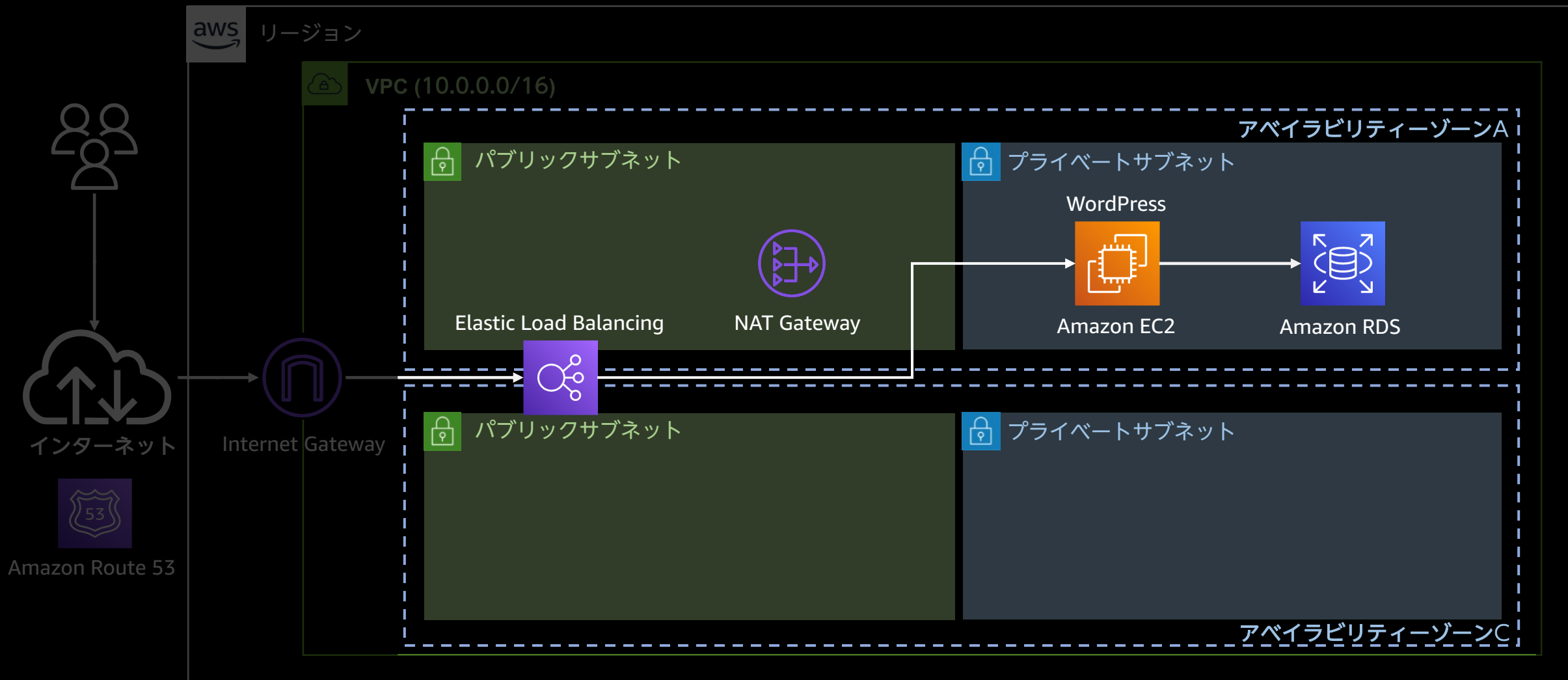
# AWS のリージョンにおける アベイラビリティーゾーン (AZ)

- それぞれのリージョンは、複数のアベイラビリティーゾーン (AZ) で構成される。
- AZ は自然災害やデータセンター単位の障害などビジネスに影響を与えるリスクを最小化するように地理的に影響を受けない十分離れた場所 (100 km (60 マイル) 以内) にあり、独立した電源、空調、物理的なセキュリティを備えている。

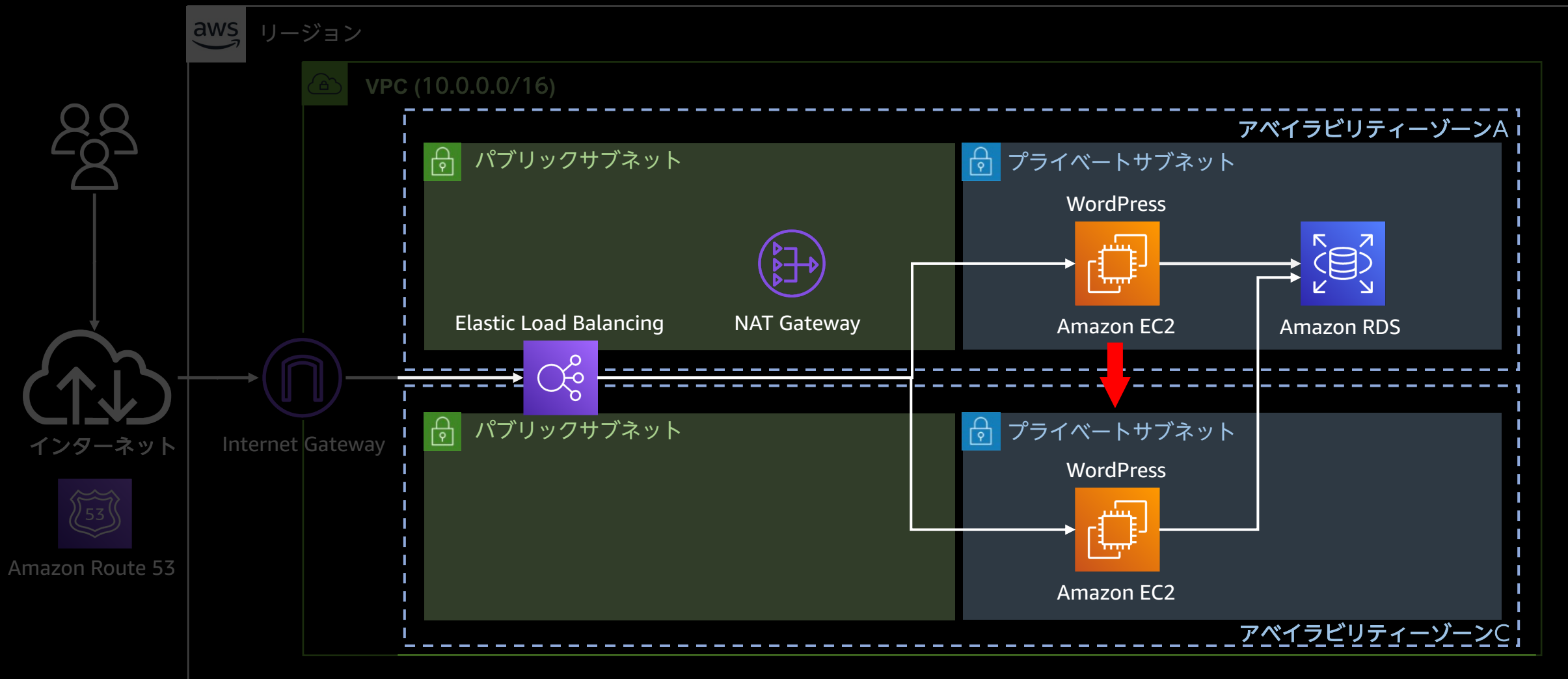




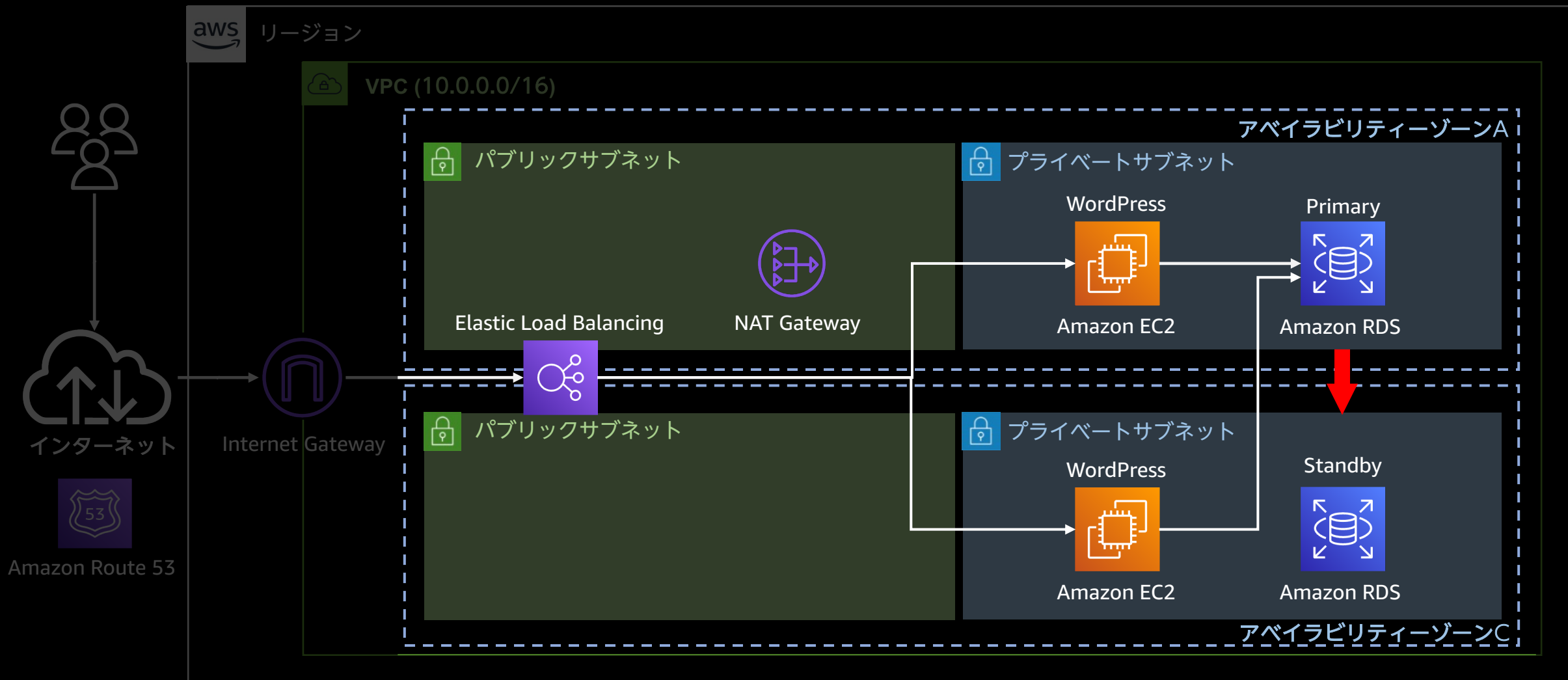
# 基本の Web システムのアーキテクチャ



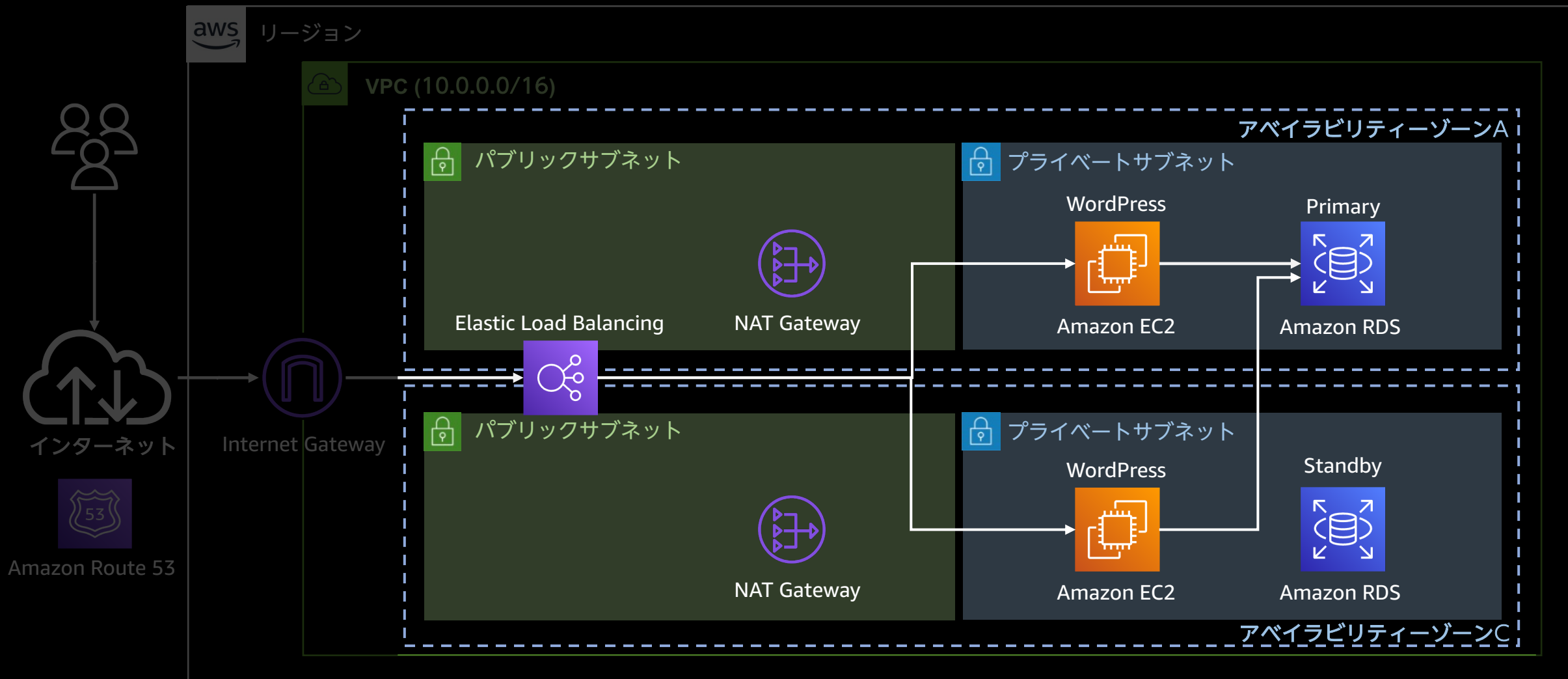
# EC2 を複数の AZ に配置する



# RDS を複数の AZ に配置する



# マルチ AZ を利用したアーキテクチャ



# 可用性とスケーラビリティ

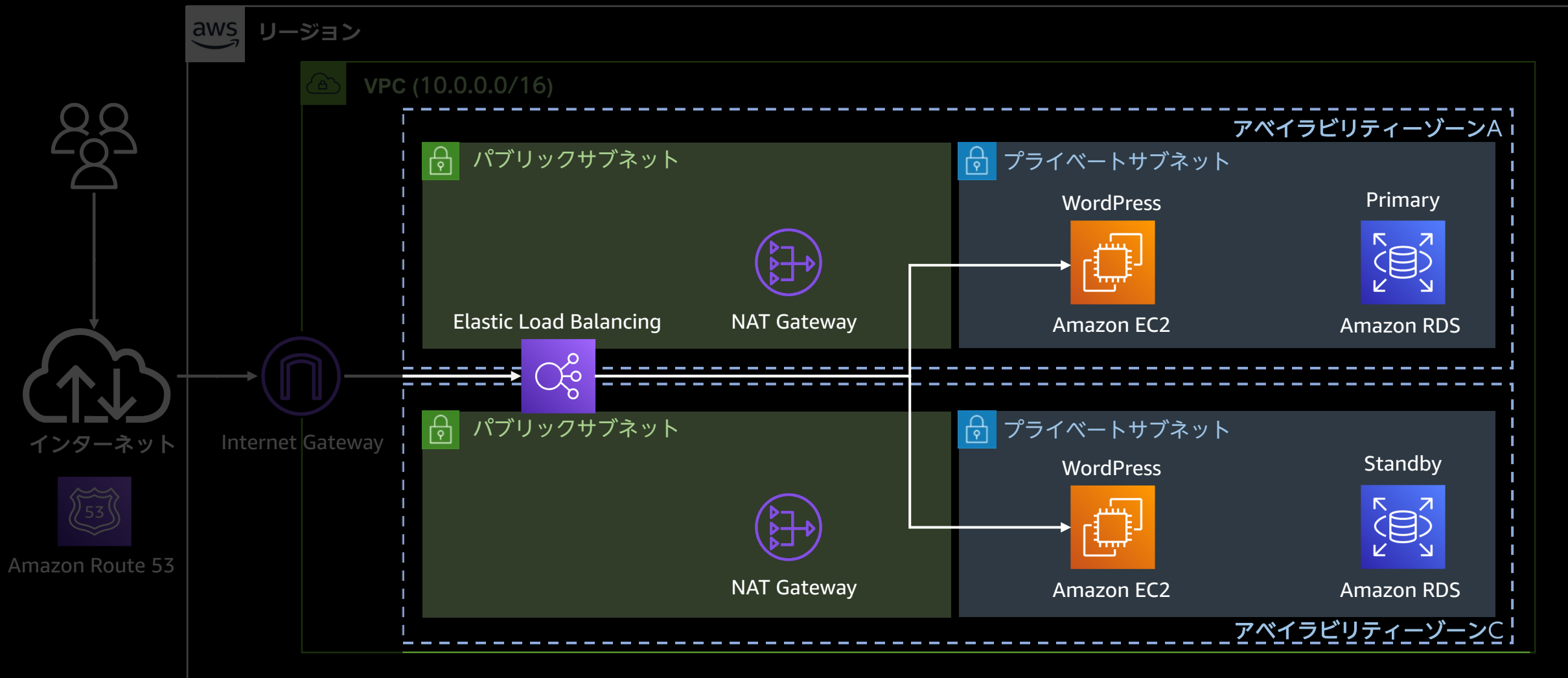
## 可用性

システム全体が障害で  
停止することなく、  
サービスを提供し続け  
ることができる能力

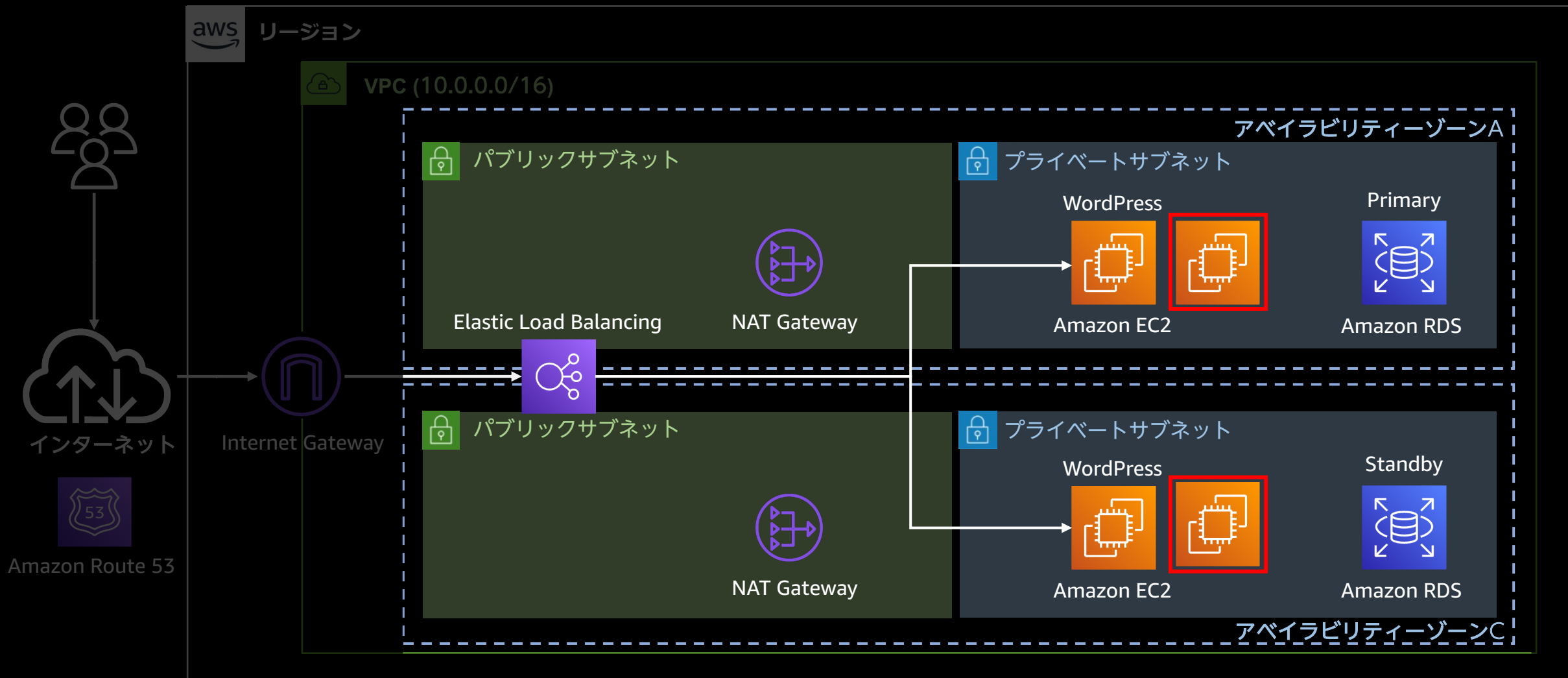
## スケーラビリティ

システムの規模の変化  
に柔軟に対応すること  
ができる能力

# 高負荷にどうやって対処するか？



# 高負荷にどうやって対処するか？



# スケーラビリティを実現するには

- スケールアウト
  - EC2の起動台数を増やしていく
- スケールイン
  - EC2の起動台数を減らしていく

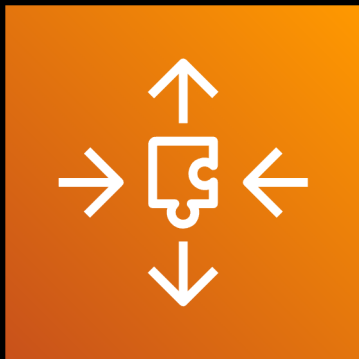


# スケーラビリティを実現するには

- スケールアウト
  - EC2の起動台数を増やしていく
- スケールイン
  - EC2の起動台数を減らしていく

負荷に対して必要となる分だけリソースを  
起動することでコスト最適化が可能

# 負荷に応じて柔軟にスケーリングさせる



Amazon EC2 Auto Scaling

EC2 インスタンスの柔軟なスケールと可用性確保

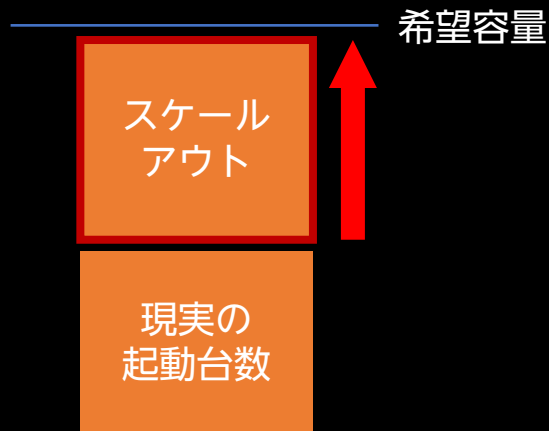
希望容量と、現在の起動台数との差を監視  
希望容量の値と一致するように、インスタンスの台数を増減

希望容量の増減方法はさまざまな設定が可能  
Ex) 平均 CPU 使用率、時間指定予約など

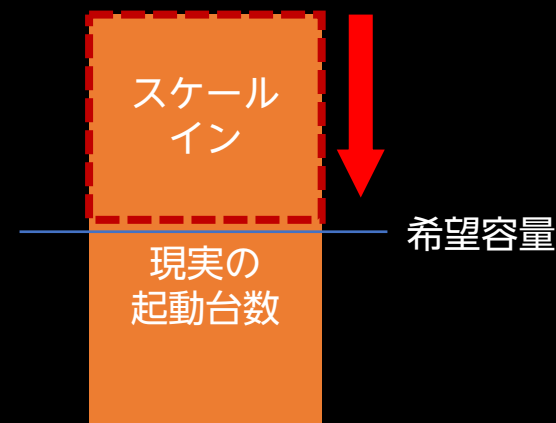
## 1) 静観



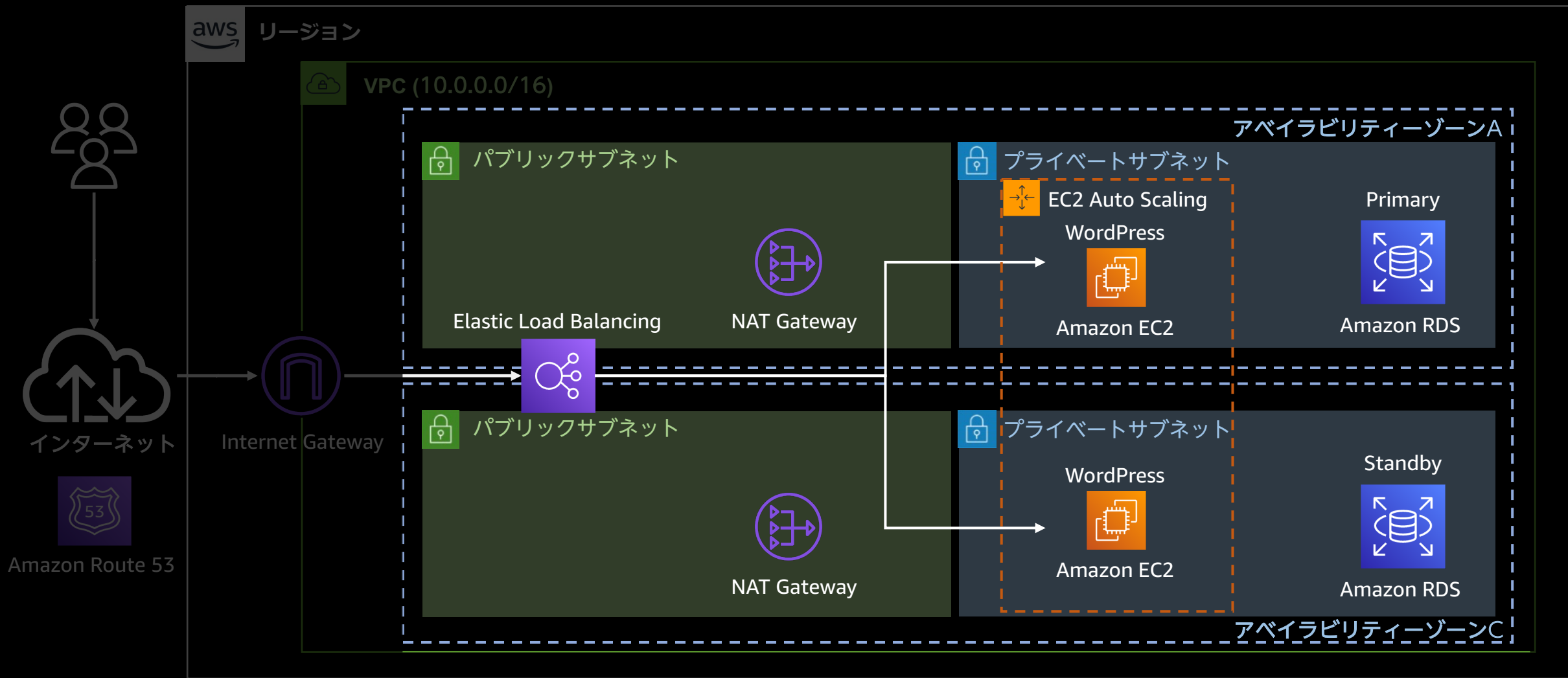
## 2) スケールアウト



## 3) スケールイン



# Auto Scaling を導入したアーキテクチャ



# まとめ

# おすすめしたいハンズオン

本セッションでご紹介した内容を、  
具体的な画面とデモを見ながら進めることができるハンズオンです

【ハンズオン】  
スケーラブルウェブ  
サイト構築編



【ハンズオン】  
Amazon EC2  
Auto Scaling  
スケーリング基礎編



「Z1 関連資料」にて、今すぐお試し可能です

# さいごに

- 可用性とスケーラビリティを向上させるための方法をデモを通して紹介
  - Design for Failure の考え方
  - 複数の AZ を使った EC2 インスタンス / RDS インスタンスの冗長化
  - EC2 Auto Scaling を利用したスケーラビリティの確保
- 今回ご紹介した内容を、お客様の環境の可用性とスケーラビリティを高めるための「基礎」としてご活用ください

# AWS TRAINING & CERTIFICATION

## AWS Skill Builder の 500+ の 無料デジタルコースで学ぼう

30以上のAWSソリューションの中から、自分に最も関係のあるクラウドスキルとサービスにフォーカスし、自習用のデジタル学習プランとRamp-Upガイドで学ぶことができます。

- 自分のペースでAWSクラウド上を活用した未来を切り開く
- 学習プランでスキルや知識を向上
- AWS認定資格でクラウドの専門知識を証明する

自分に合ったスキルアップ方法で学びましょう  
[EXPLORE.SKILLBUILDER.AWS](https://explore.skillbuilder.aws) »



# AWS Builders Online Series に ご参加いただきありがとうございます

楽しんでいただけましたか? ぜひアンケートにご協力ください。  
本日のイベントに関するご意見/ご感想や今後のイベントについてのご希望や改善のご提案などがございましたら、ぜひお聞かせください。



[aws-apj-marketing@amazon.com](mailto:aws-apj-marketing@amazon.com)



[twitter.com/awscloud\\_jp](https://twitter.com/awscloud_jp)



[facebook.com/600986860012140](https://facebook.com/600986860012140)



<https://www.youtube.com/user/AmazonWebServicesJP>



<https://www.linkedin.com/showcase/aws-careers/>



[twitch.tv/aws](https://twitch.tv/aws)



# Thank you!

