# AWS Summits 2014

## エンタープライズ向けAWSクラウドデザインパターンのご紹介（ネットワーク編）

荒木靖宏, アマゾンデータサービスジャパン

July 18, 2014

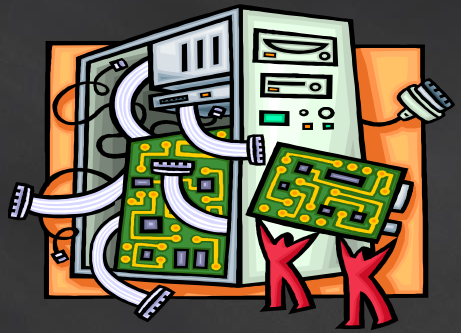
amazon web services

# 自己紹介

- 名前
  - 荒木 靖宏
- 所属
  - アマゾンデータサービスジャパン株式会社
    プリンシパルソリューションアーキテクト
- ID
  - Twitter: ar1
- 好きなAWSサービス
  - Amazon Virtual Private Cloud
  - AWS Direct Connect

# よく聞く話

- サービスが多くてよく分からない
  - 組み合わせ方が分からない
- 使い方が正しいのか分からない

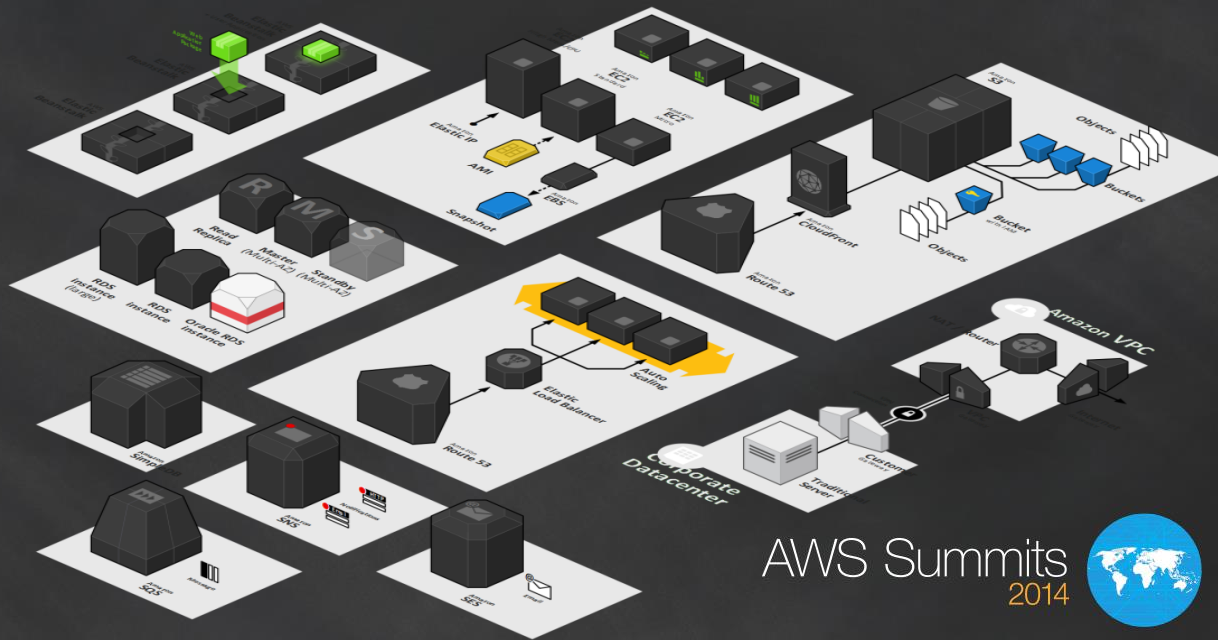


# AWSクラウドデザインパターンとは…

- AWSクラウドを使ったシステムアーキテクチャ設計を行う際に発生する、典型的な問題とそれに対する解決策・設計方法を、分かりやすく分類して、ノウハウとして利用できるように整理したもの。

# うまく組み合わせると・・

- ピーク対応が楽
- セキュアなシステム構築
- 安価に世界展開
- 障害にも強い



# CDPカテゴリ

**-基本パターン**
- Snapshot
- Stamp
- Scale Up
- Ondemand Disk

**-可用性を高める**
- Multi-Server
- Multi-Datacenter
- Floating IP
- Deep Health Check

**-動的コンテンツを処理**
- Scale Out
- Clone Server
- NFS Sharding
- NFS Replica
- State Sharing
- URL Rewriting
- Rewrite Proxy
- Cache Proxy

**-静的コンテンツを処理**
- Web Storage
- Direct Hosting
- Private Distribution
- Cache Distribution
- Rename Distribution

**-クラウドへのアップロード**
- Write Proxy
- Storage Index
- Direct Object Upload

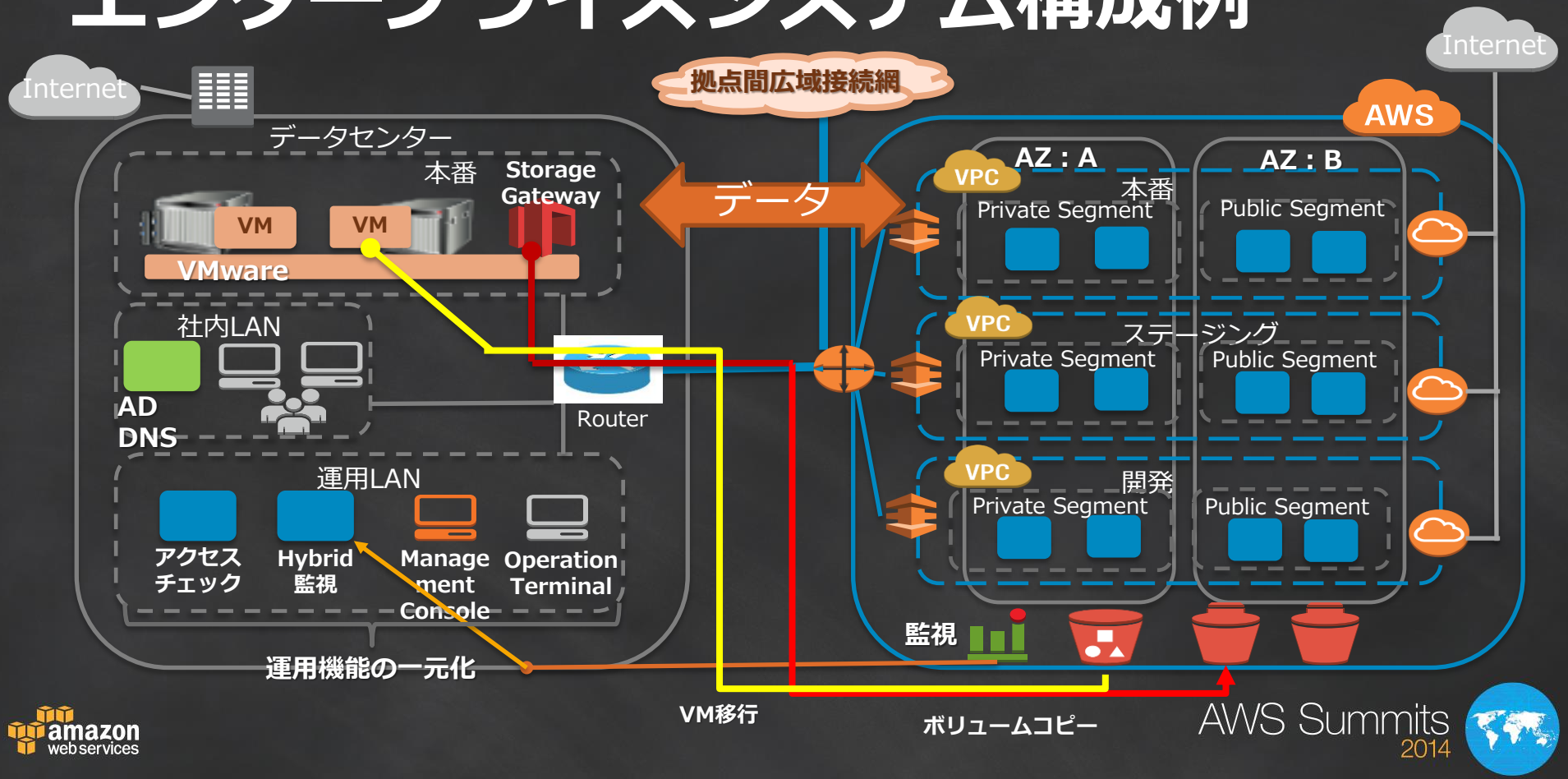
**-リレーショナル・データベース**
- DB Replication
- Read Replica
- Inmemory DB Cache
- Sharding Write

**-バッチ処理**
- Queuing Chain
- Priority Queue
- Job Observer
- Scheduled Autoscaling

**-運用保守**
- Bootstrap
- Cloud DI
- Stack Deployment
- Server Swapping
- Monitoring Integration
- Web Storage Archive

**-ネットワーキング**
- OnDemand NAT
- Backnet
- Functional Firewall
- Operational Firewall
- Multi Load Balancer
- WAF Proxy
- CloudHub

# エンタープライズシステム構成例



# エンタープライズシステム設計

## インフラ

- 対象アプリ決定
- リージョン設計
- VPC設計
- インターネット VPN
- 専用線

本セッションの対象

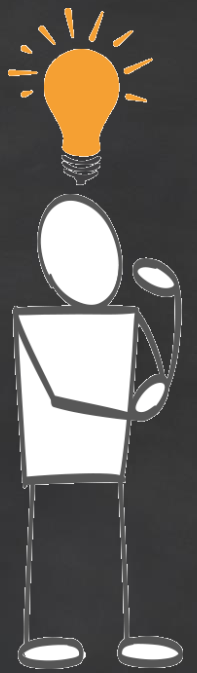
## データ：移行とバックアップ

- ブロックデバイス
- ファイル
- VM
- DB

## 運用

- ユーザ管理
- 機器変更管理
- 監視
- ログ
- ステージング環境テスト
- ユーザ誘導

# Agenda



- エンタープライズシステムへの準備
  - タグ、CloudFormation
- VPCの基本構成要素とパターン
- 内部向けアプリケーションパターン
- バックホームパターン

# エンタープライズシステムへの準備


Amazon VPC


Router


Internet Gateway


Customer Gateway


Subnet


Virtual Private Gateway


VPN Connection

Route Table


Elastic Network Interface

# Tag：最初から、頻繁に

- タグの利用は設計の初期から行う
- 使いそうな情報はなんでもタグ化
  - プロジェクトコード、コスト負担部署、実施チーム、環境、利用目的などなど
- リソースを作ってからでもにタグ付けできる
- タグはリソースの利用権限制御に使える
- AWS Billingもタグをサポートしている

# AWS CloudFormation: Infrastructure as Code




CloudFormation
JSON (Text)
Template
Stack

# AWS CloudFormationで得るもの

- バージョン管理されたデータセンタ設計
- コマンド一発でインフラ展開
- いつでも世界中のリージョンで再現可能
- インフラとアプリケーションの明確な分離

# VPCの基本構成要素とパターン

Amazon VPC

Router

Internet Gateway

Customer Gateway

Subnet

Virtual Private Gateway

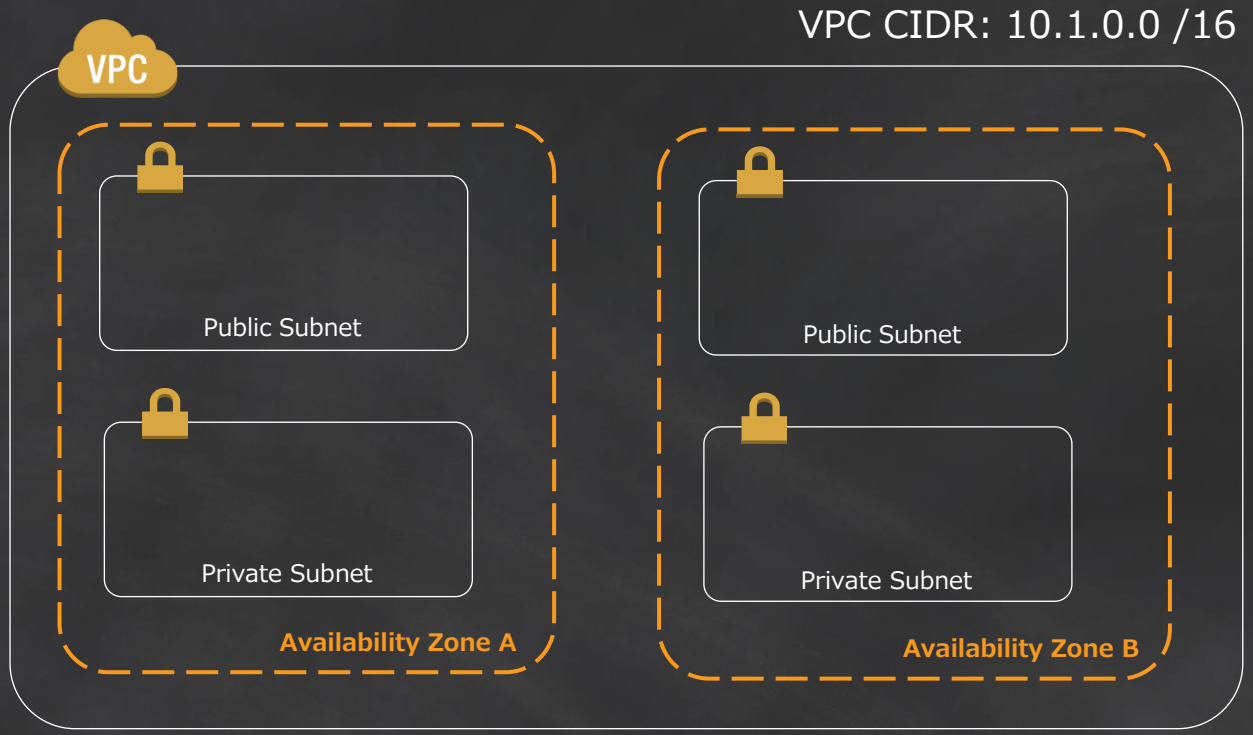
VPN Connection

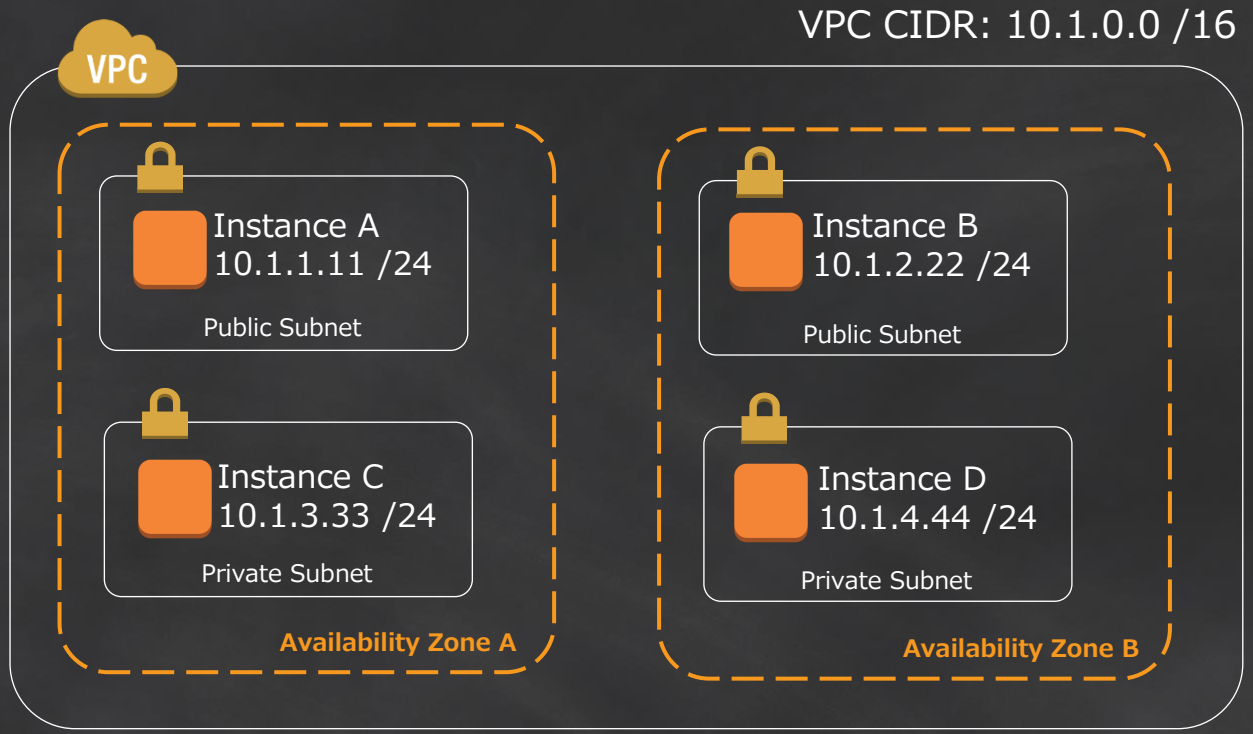

Route Table

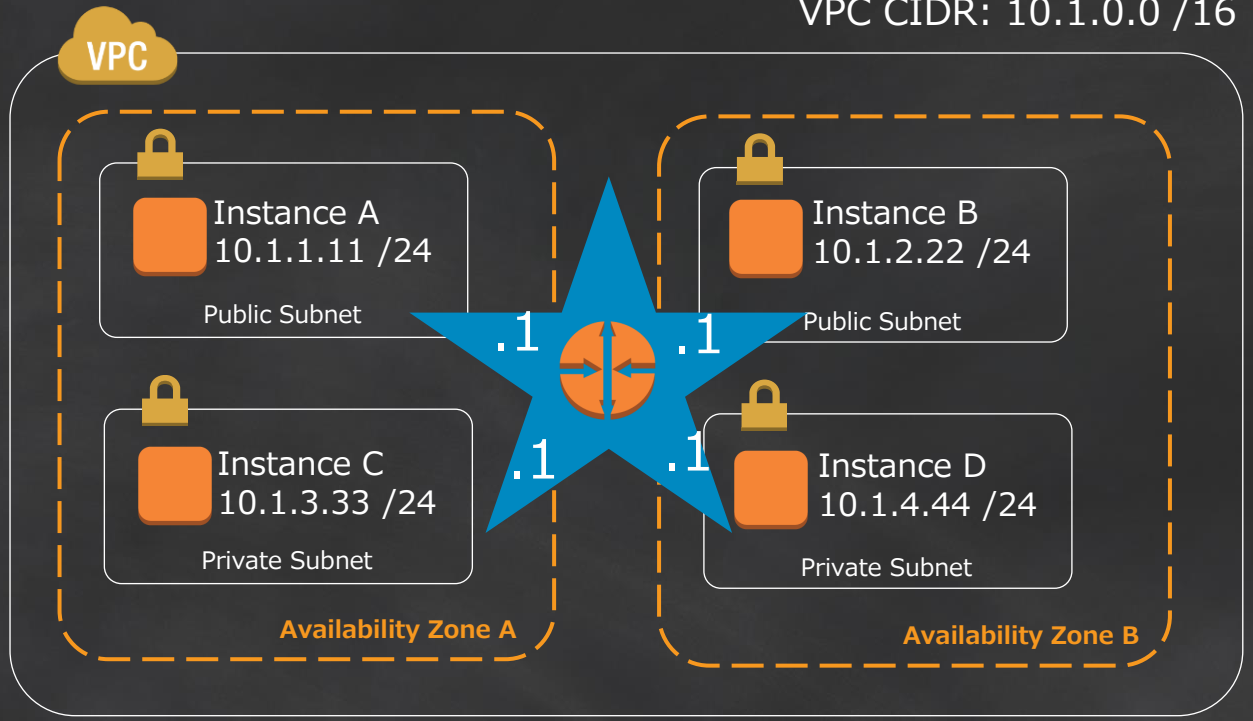
Elastic Network Interface

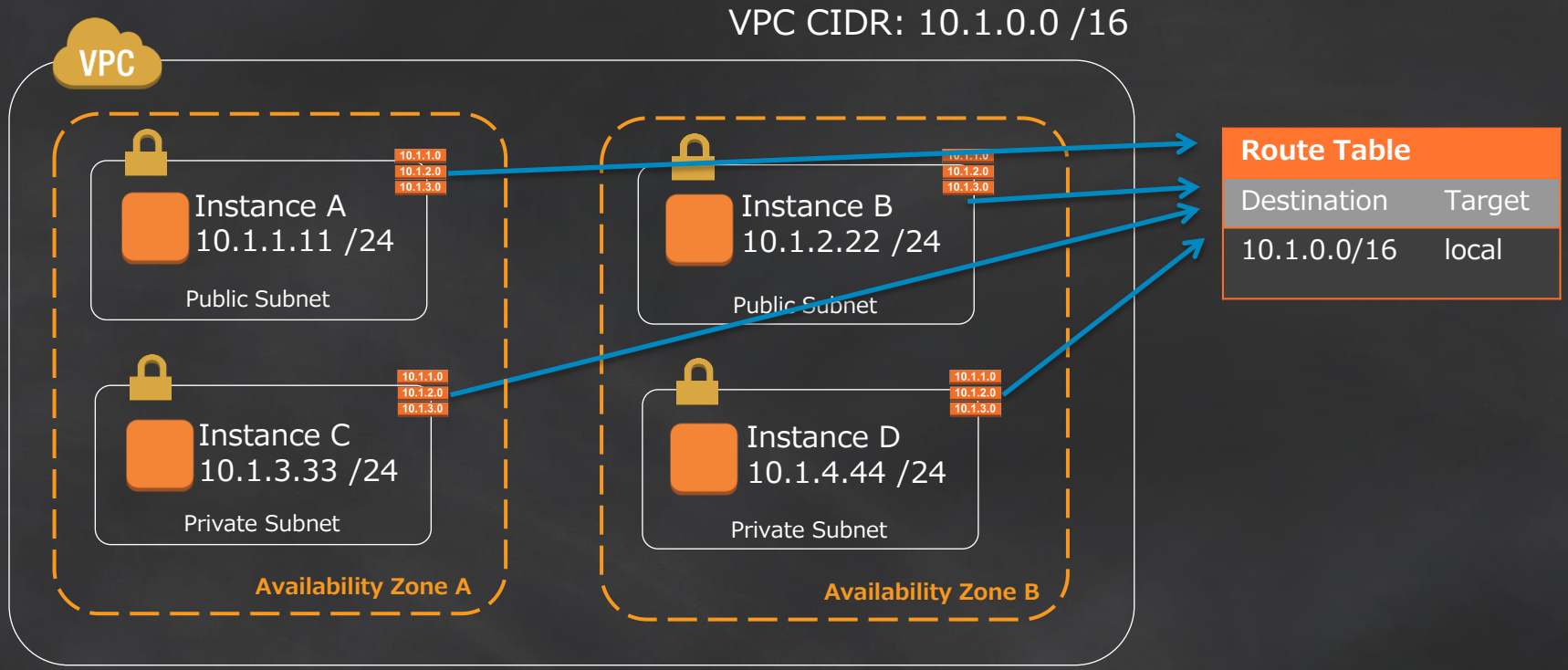
# VPCを作る前にIP空間を考える

- リージョンの将来拡張に備える
- 社内ネットワークの将来拡張に備える
- VPCの制約を理解する
  - VPCサイズは /16 （約6万5千）から /28（16）
  - IPアドレス空間は初期作成時から変更できない


VPC CIDR: 10.1.0.0 /16
VPC
Public Subnet
Private Subnet
Availability Zone A
Public Subnet
Private Subnet
Availability Zone B


VPC CIDR: 10.1.0.0 /16
VPC
Instance A
10.1.1.11 /24
Public Subnet
Instance B
10.1.2.22 /24
Public Subnet
Instance C
10.1.3.33 /24
Private Subnet
Instance D
10.1.4.44 /24
Private Subnet
Availability Zone A
Availability Zone B


VPC CIDR: 10.1.0.0 /16
VPC
Instance A
10.1.1.11 /24
Public Subnet
Instance B
10.1.2.22 /24
Public Subnet
.1
.1
.1
.1
Instance C
10.1.3.33 /24
Private Subnet
Instance D
10.1.4.44 /24
Private Subnet
Availability Zone A
Availability Zone B


VPC CIDR: 10.1.0.0 /16
VPC
Instance A
10.1.1.11 /24
Public Subnet
10.1.1.0
10.1.2.0
10.1.3.0
Instance B
10.1.2.22 /24
Public Subnet
Instance C
10.1.3.33 /24
Private Subnet
Instance D
10.1.4.44 /24
Private Subnet
Availability Zone A
Availability Zone B
Route Table
Destination
Target
10.1.0.0/16
local

# Mainルートテーブルは使わない

| Route Table ID | Associated With | Main | VPC |
|---|---|---|---|
| rtb-39ca9d52 | 0 Subnets | Yes | vpc-3bca9d50 (10.1.0.0/16) |

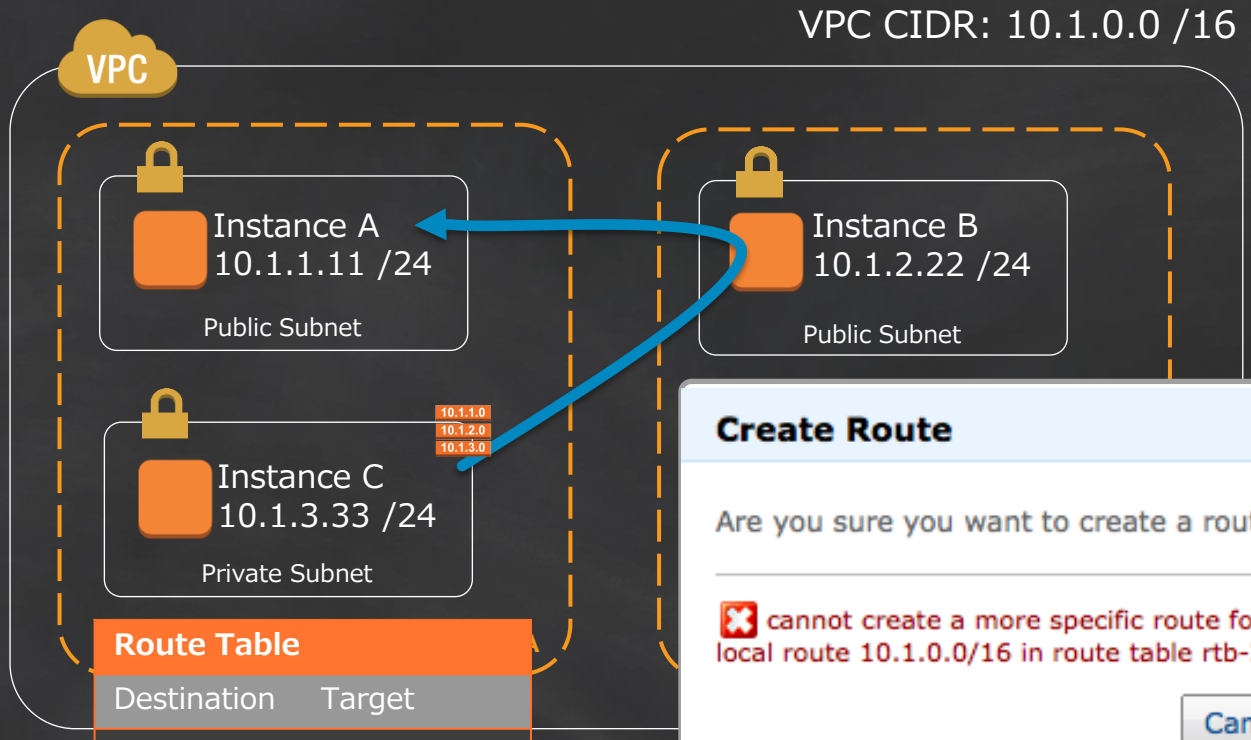
**Route Table:** rtb-39ca9d52

Routes | **Associations** | Route Propagation | Tags

| Subnet | Actions |
|---|---|
| Select a subnet | Associate |

The following subnets have not been associated with any route tables and are therefore using the Main table routes:

- subnet-6af6a101 (10.1.4.0/24)
- subnet-2ff7a044 (10.1.1.0/24)
- subnet-8ef7a0e5 (10.1.3.0/24)
- subnet-d4f7a0bf (10.1.2.0/24)

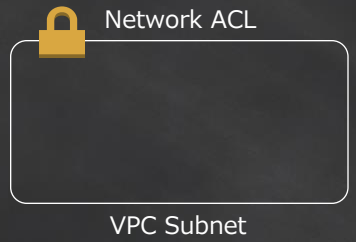



VPC内の通信経路は
編集も削除もできない

# Network ACLs と セキュリティグループ


## NACLs

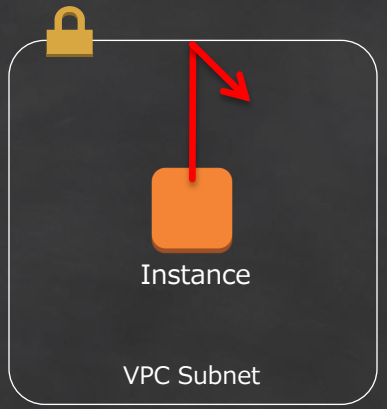
- サブネットに1つだけ適用
- ステートレス
- Allow & Deny (blacklist)
- ルール順に評価される




## Security Groups

- ENIに5つまで適用
- ステートフル
- Allowのみ (whitelist)
- ルール全体が評価される
- 同一VPC内の全てのセキュリティグループを参照可能

# Network ACLsの利用パターン




- 最低限のSecurityポリシー適用
  - 例:“全サブネットから外部へのSMTP禁止”
- インスタンス毎に適用されているセキュリティグループの漏れの補完
- ネットワーク運用と、開発者の分離

# VPC Network ACLsのベストプラクティス

- 慎重かつシンプルに
- VPCから出て行く方向のポリシーに使う
- Rule番号は成長を見越して間をあける
- 修正・削除できるユーザはIAMを使って制限

何も考えずに押さないで!


Default Network ACL:

| Rule # | Port (Service) | Protocol | Source | Allow/Deny | Action |
|---|---|---|---|---|---|
| 100 | ALL | ALL | 0.0.0.0/0 | ALLOW | Delete |
| * | ALL | ALL | 0.0.0.0/0 | DENY | |

# VPC管理用のIAM

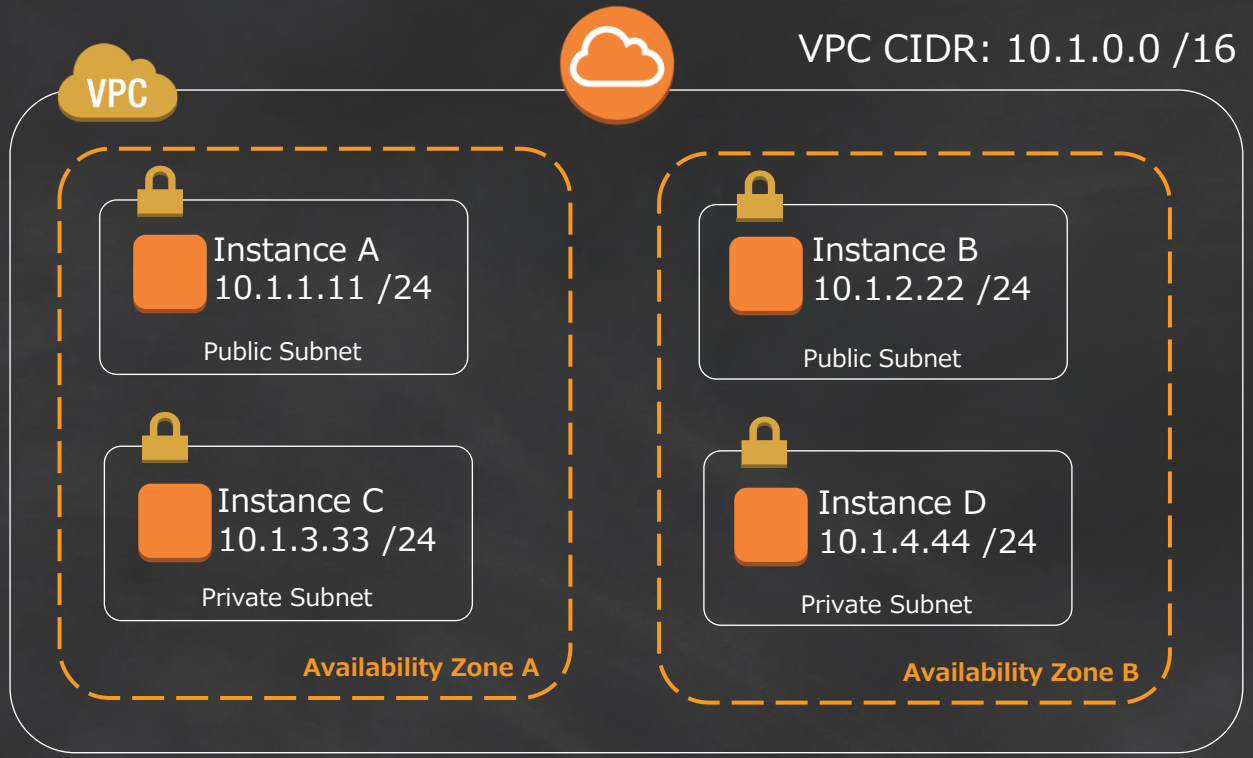
致命的な影響を与えかねないVPCのAPIコール

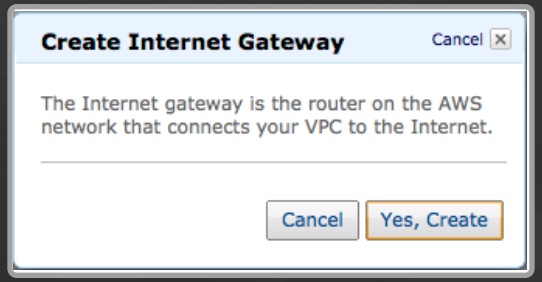
AttachInternetGateway
AssociateRouteTable
CreateRoute
DeleteCustomerGateway
DeleteInternetGateway
DeleteNetworkAcl
DeleteNetworkAclEntry
DeleteRoute
DeleteRouteTable
DeleteDhcpOptions
ReplaceNetworkAclAssociation
DisassociateRouteTable

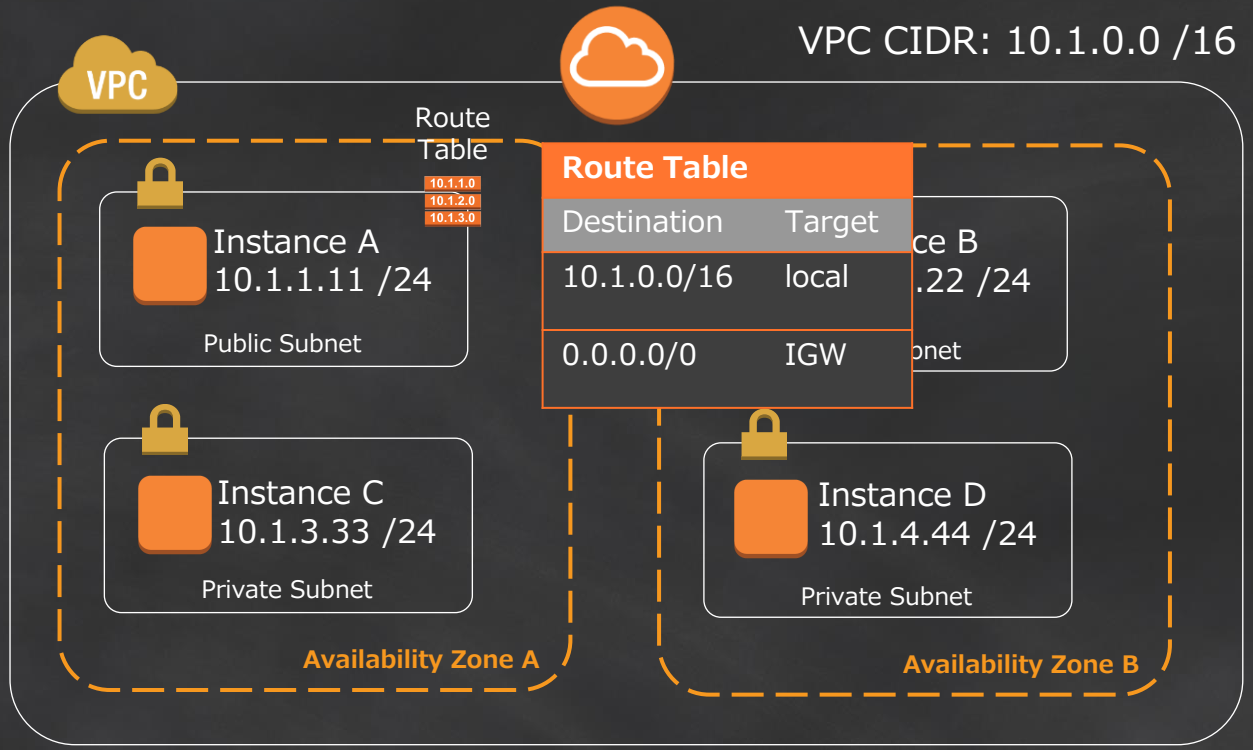
# NACL管理者の設定例：MFAの適用

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
     "Effect": "Allow",
     "Action": [
       "ec2:DeleteNetworkAcl",
       "ec2:DeleteNetworkAclEntry"
     ],
     "Resource": "arn:aws:ec2:us-west-2:123456789012:network-acl/*",
     "Condition": {
       "StringEquals": {
         "ec2:ResourceTag/Environment": "prod"
       },
                "Null": {
              "aws:MultiFactorAuthAge": "false"
       }
     }
    }
  ]
}
```

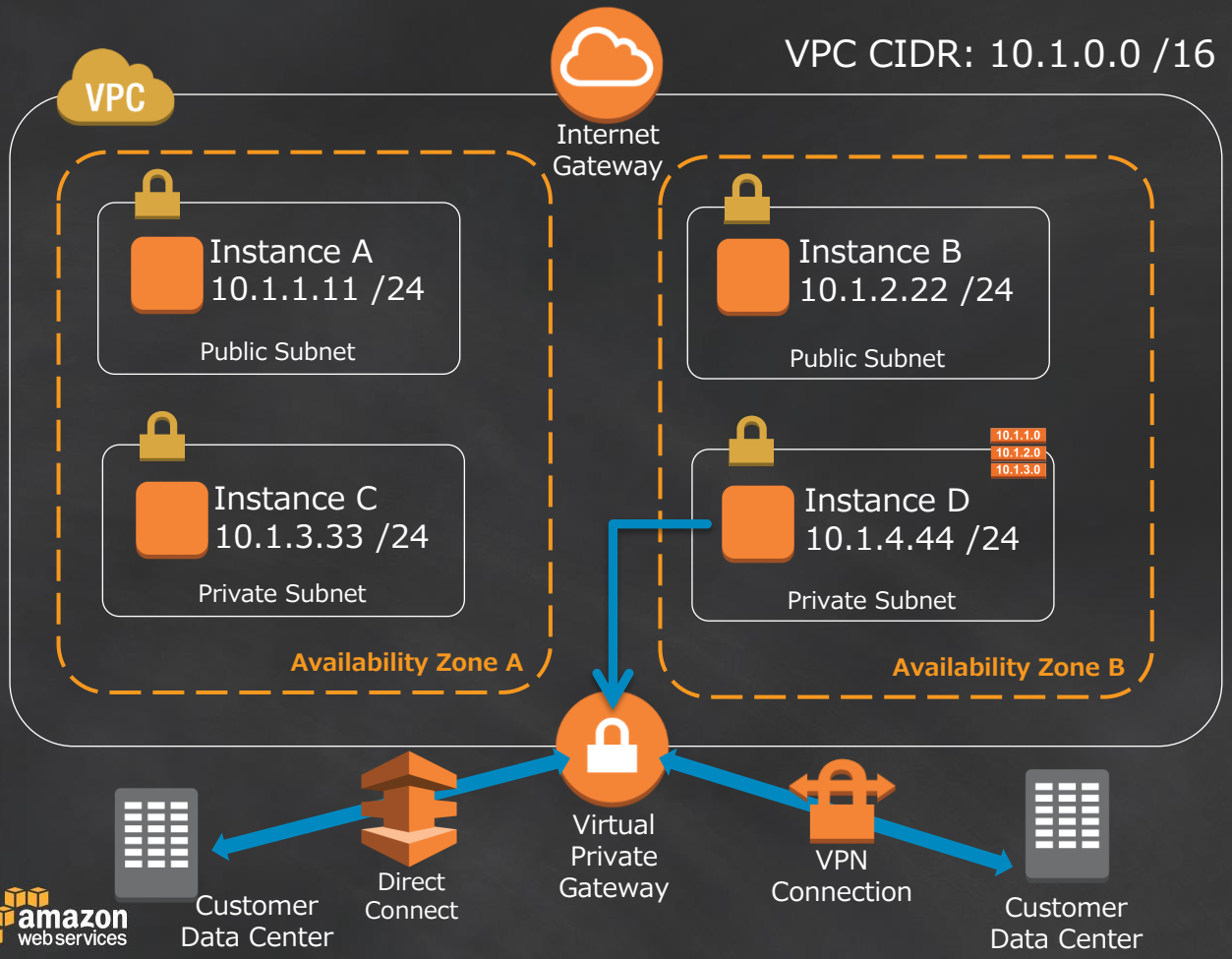
# VPCからの出口の作成




VPC CIDR: 10.1.0.0 /16
VPC
Route Table
10.1.1.0
10.1.2.0
10.1.3.0
Route Table
Destination
Target
10.1.0.0/16
local
0.0.0.0/0
IGW
Instance A
10.1.1.11 /24
Public Subnet
Instance C
10.1.3.33 /24
Private Subnet
Availability Zone A
Instance D
10.1.4.44 /24
Private Subnet
Availability Zone B

# インターネットゲートウェイとバーチャルゲートウェイの利用





| Route Table | |
|---|---|
| Destination | Target |
| 10.1.0.0/16 | local |
| Corp CIDR | VGW |

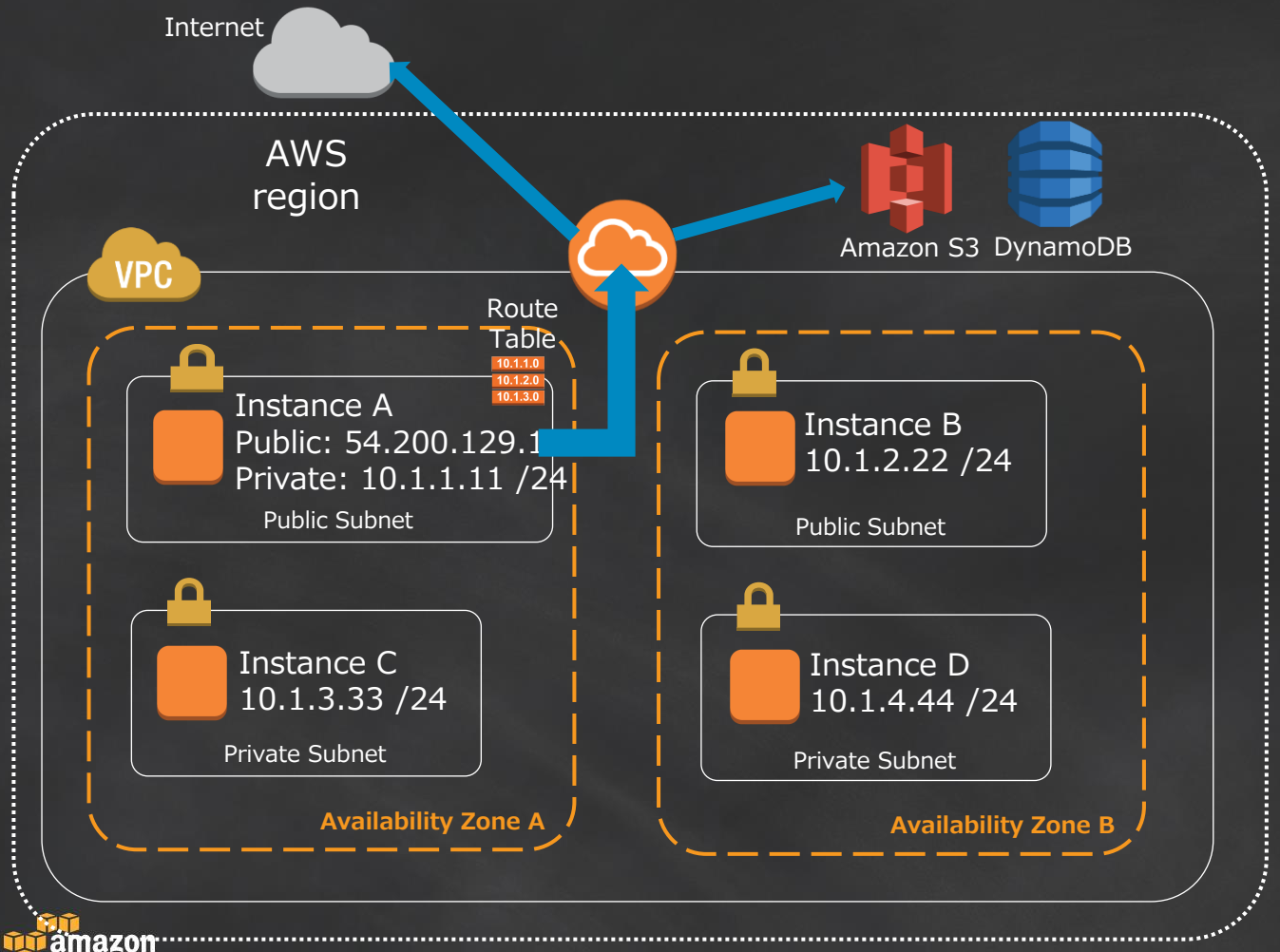
# Elastic IPを使った パブリックIPアドレスの付与

- AWSアカウントに対して付与される
  - インスタンスとの紐付けはいつでも可能
  - インスタンス間やENI間の移動も可能
- パブリックIPアドレスとプライベートIPアドレスは1対1にマッピングされる
- 1 Public IP to 1 Private IP static NAT mapping
- インスタンスOSはEIPとマッピングされているかはわからない

# 動的パブリックIPアドレス付与

- サブネット毎に有効化する
- インスタンスが起動するときに自動付与
- 1つめのENIでのみ有効

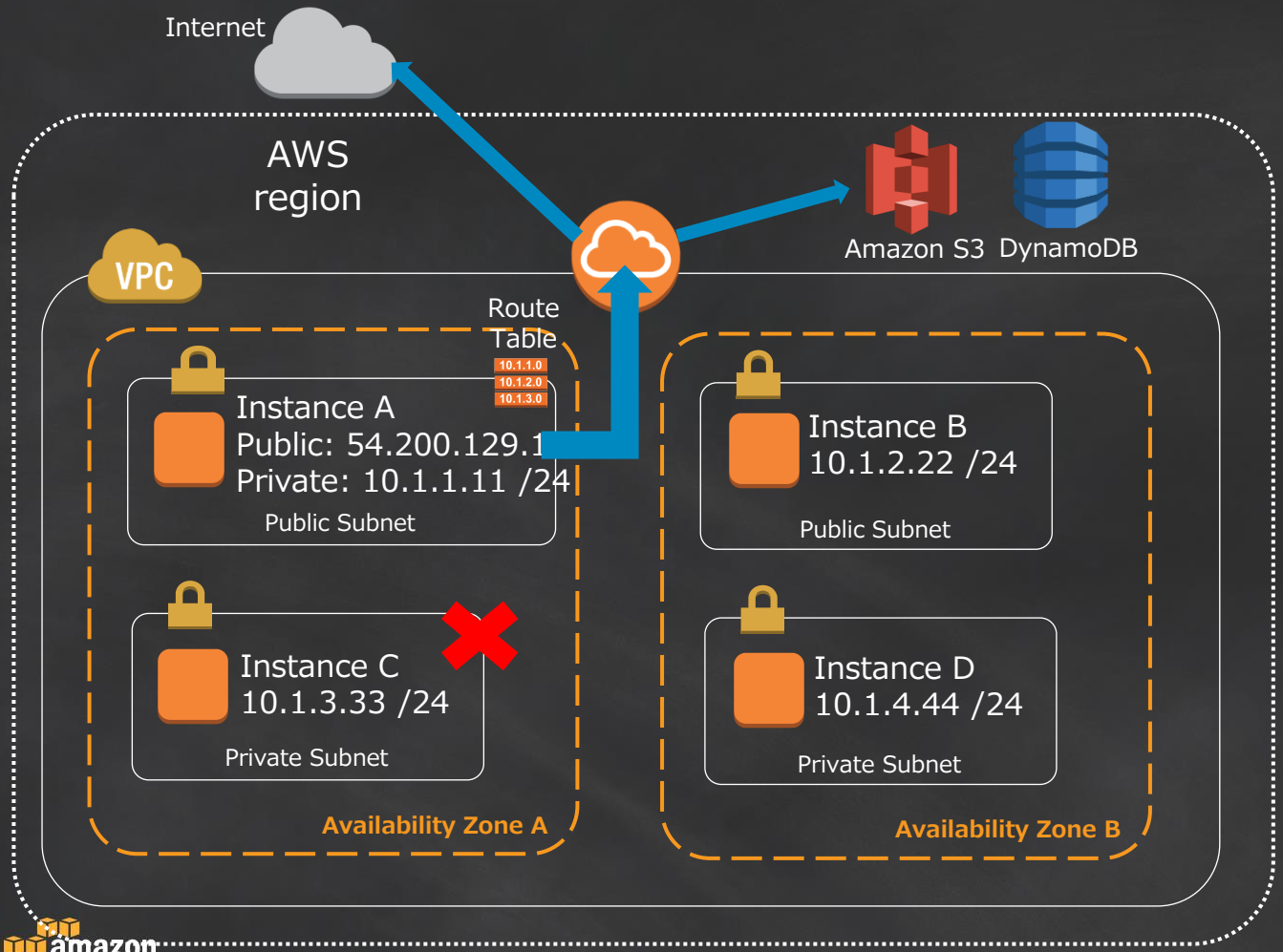
# VPCの外にあるAWS



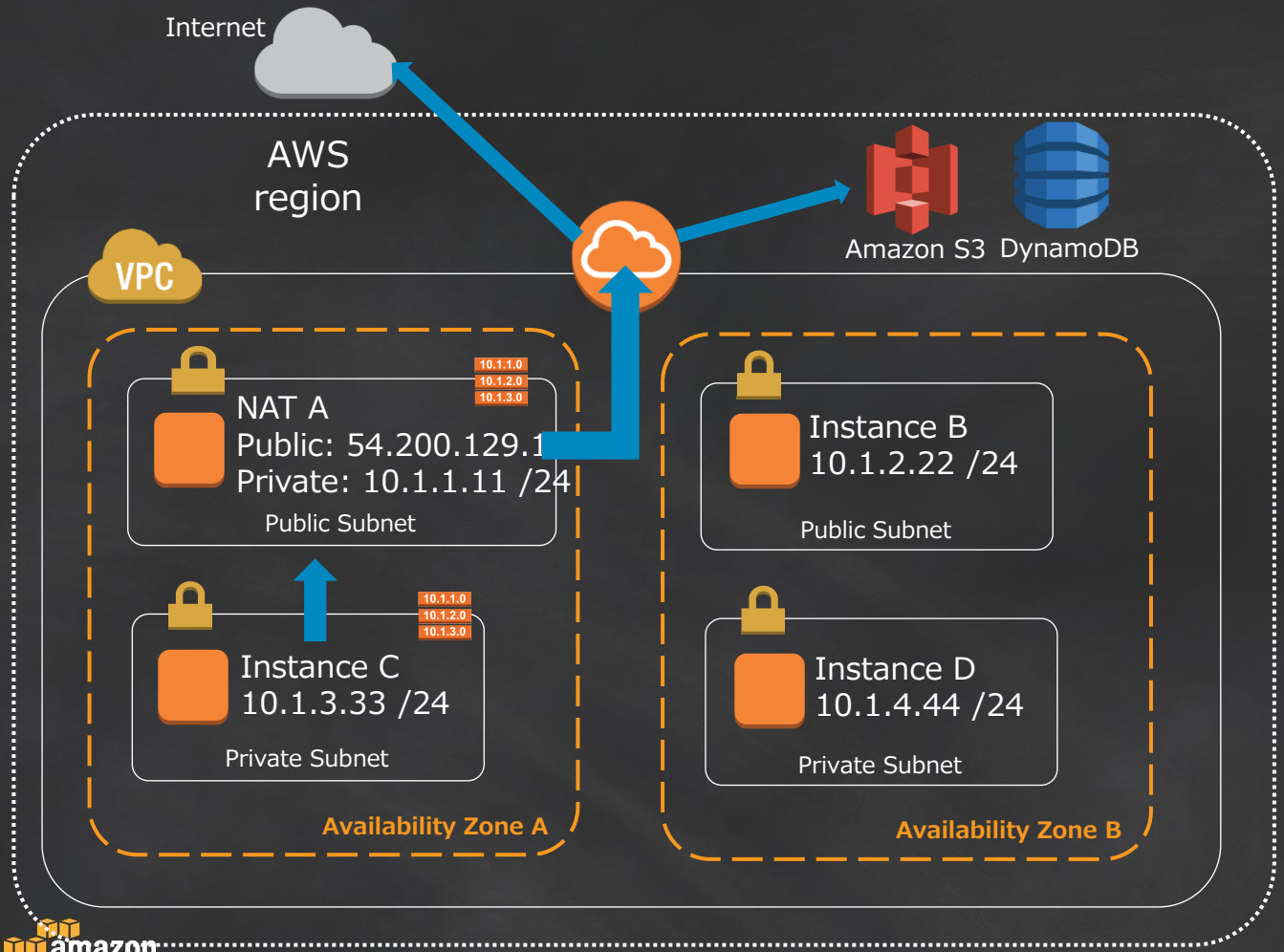

Public IPがあれば
アクセス可能

# VPCの外にあるAWSの例

- 常に使うAPIエンドポイント
  - 例：Amazon S3、Amazon Dynamo DB
- 制御時のみ使うAPIエンドポイント
  - 例：AutoScaling
- ソフトウェアレポジトリへのアクセス
  - 例：Amazon Linux repoはAWSのPublic IPからのみに限定




プライベートサブネットにあるインスタンスCがVPC外にアクセスするには？




# パブリックサブネット内インスタンスにNAT機能をもたせる

| Route Table | |
|---|---|
| Destination | Target |
| 10.1.0.0/16 | local |
| 0.0.0.0/0 | NAT instance |

# NATインスタンスは何をしているか

IP forwardの有効化

```
$echo 1 >  /proc/sys/net/ipv4/ip_forward
$echo 0 >  /proc/sys/net/ipv4/conf/eth0/send_redirects

$/sbin/iptables -t nat -A POSTROUTING -o eth0 –s 10.1.0.0/16 -j MASQUERADE
$/sbin/iptables-save
```

IPマスカレードの有効化

```
$aws ec2 modify-instance-attributes –instance-id i-xxxxxxxx –source-dest-check
“{¥”Value¥”:false}”
```

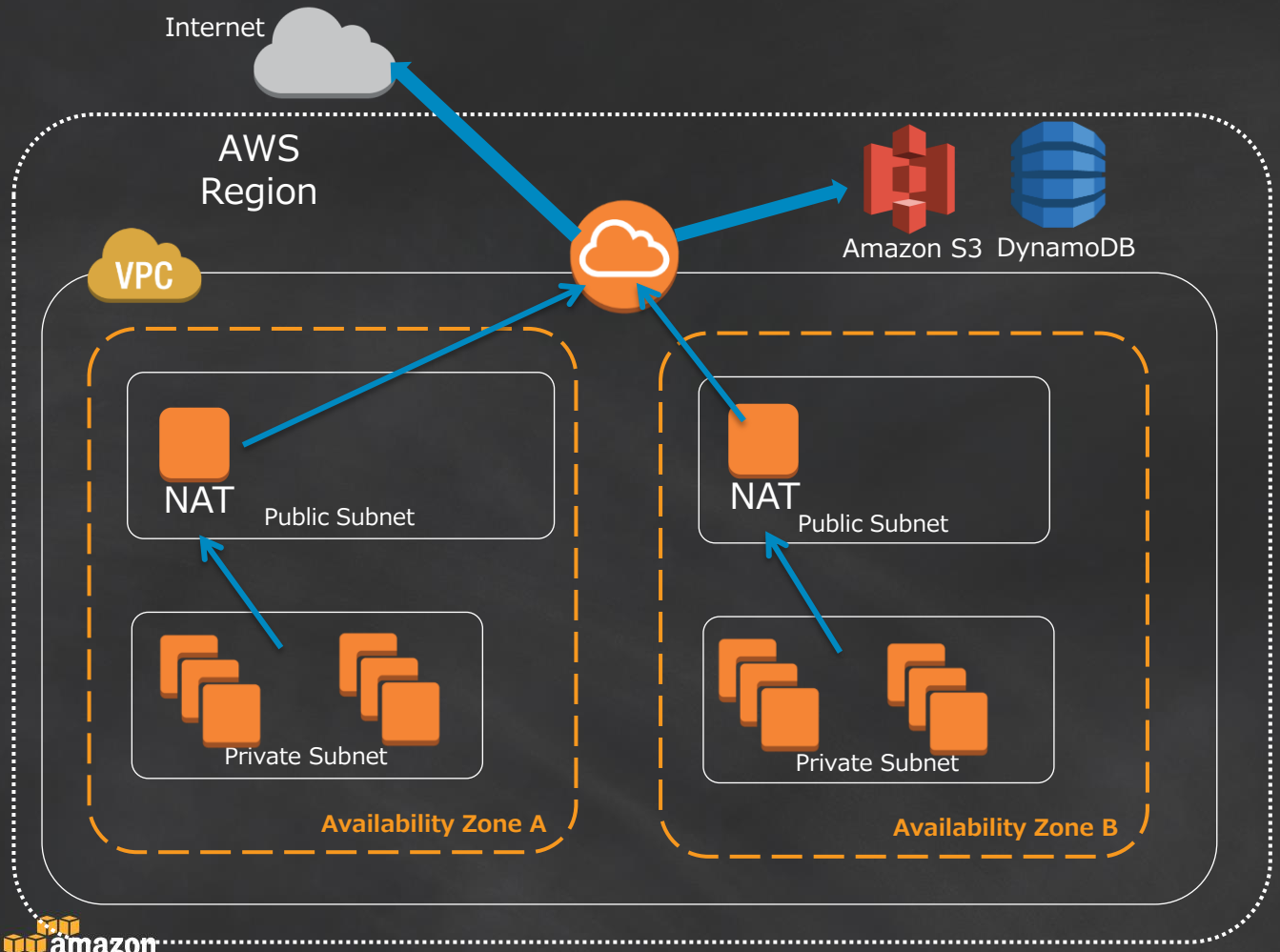
EC2での出先・宛先チェックの無効化

＋HAの機能

```
$aws autoscaling create-auto-scaling-group --auto-scaling-group-name ha-nat-
asg --launch-configuration-name ha-nat-launch --min-size 1 --max-size 1 --vpc-
zone-identifier subnet-xxxxxxxx
```

サブネット内に1台維持するオートスケール

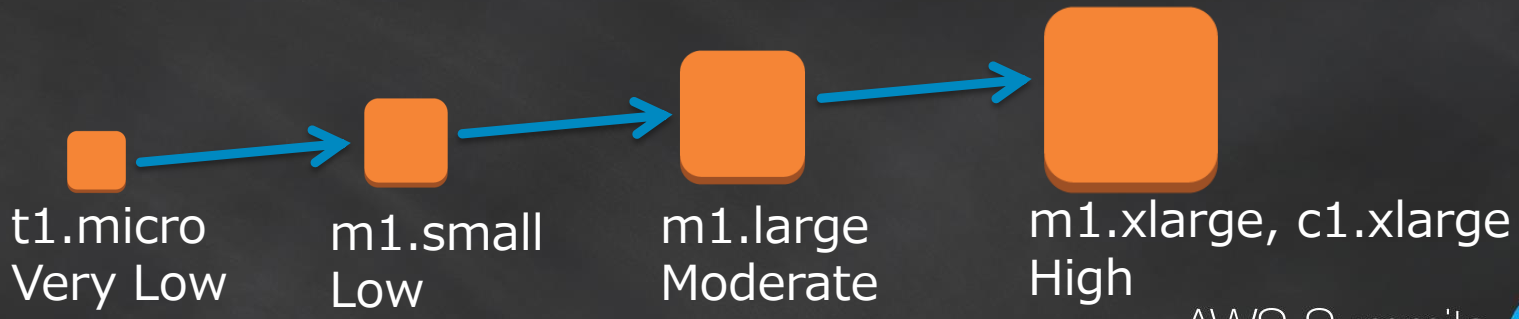
# Autoscale HA NAT




- NATインスタンスにオートスケールを設定（min=1,max=1）し、AZ毎に1NAT
- プライベートサブネットのルートテーブルは同じAZのNATにむける

# NAT下で広帯域が必要なとき

- AZ毎にHA NATを配置する
- スケールアップ！
- ネットワーク関連メトリクスを見る
- プロトコル別アプリケーションプロキシも有効



# 内部向けアプリケーションパターン


Amazon VPC


Router


Internet Gateway


Customer Gateway


Subnet

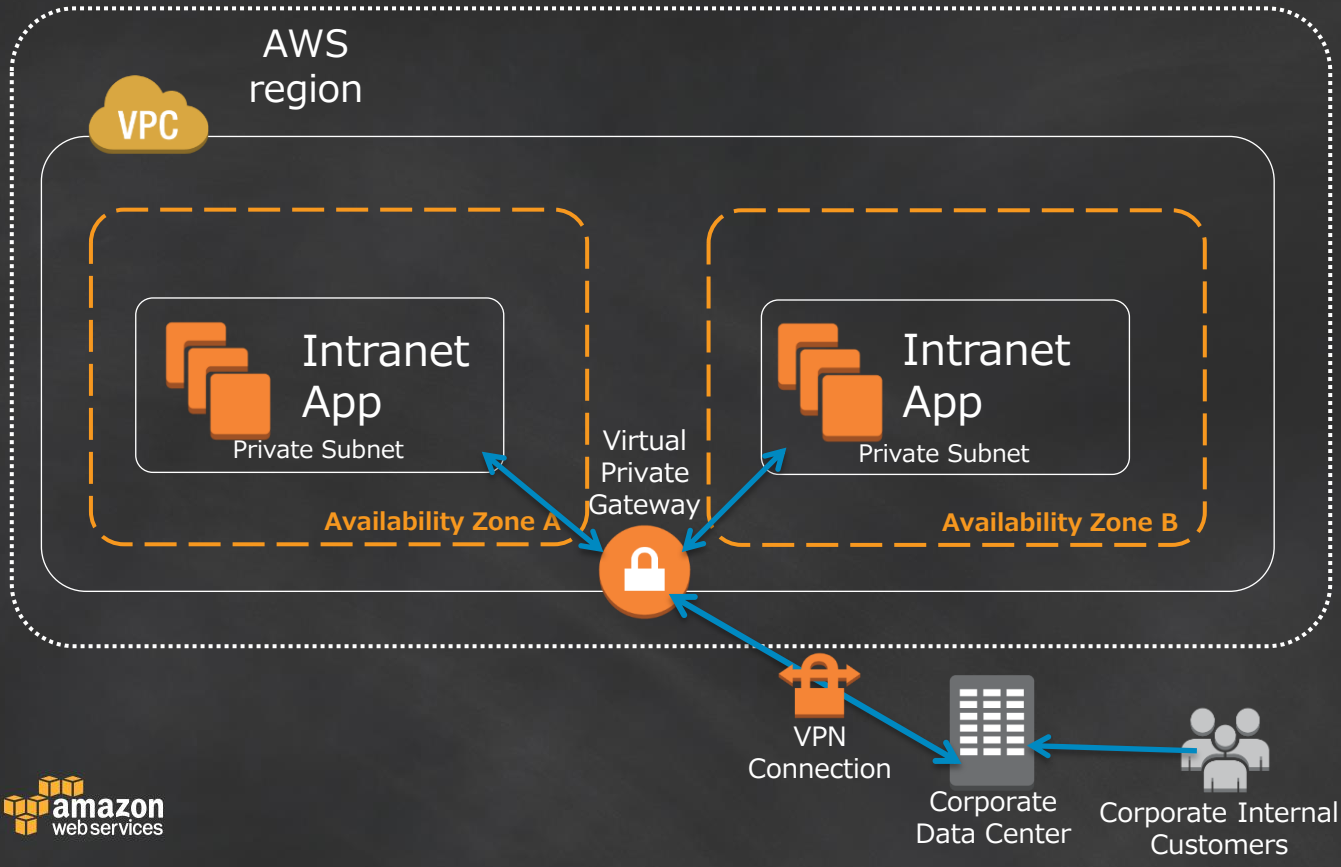

Virtual Private Gateway


VPN Connection

Route Table


Elastic Network Interface

# VPN接続とルートテーブル設定がポイント

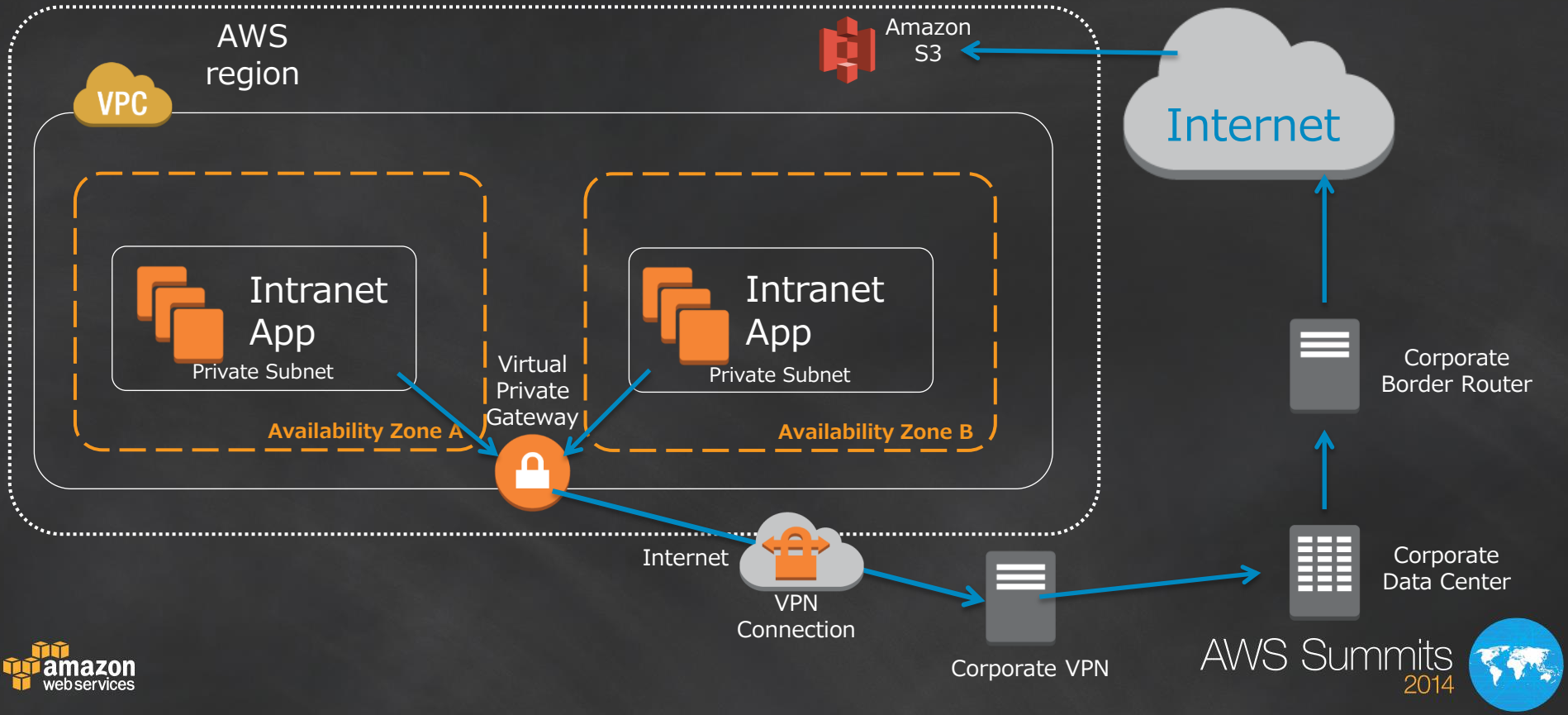



| Route Table | |
|---|---|
| Destination | Target |
| 10.1.0.0/16 | local |
| Corp CIDR | VGW |

# VPC内でデータが完結しない例

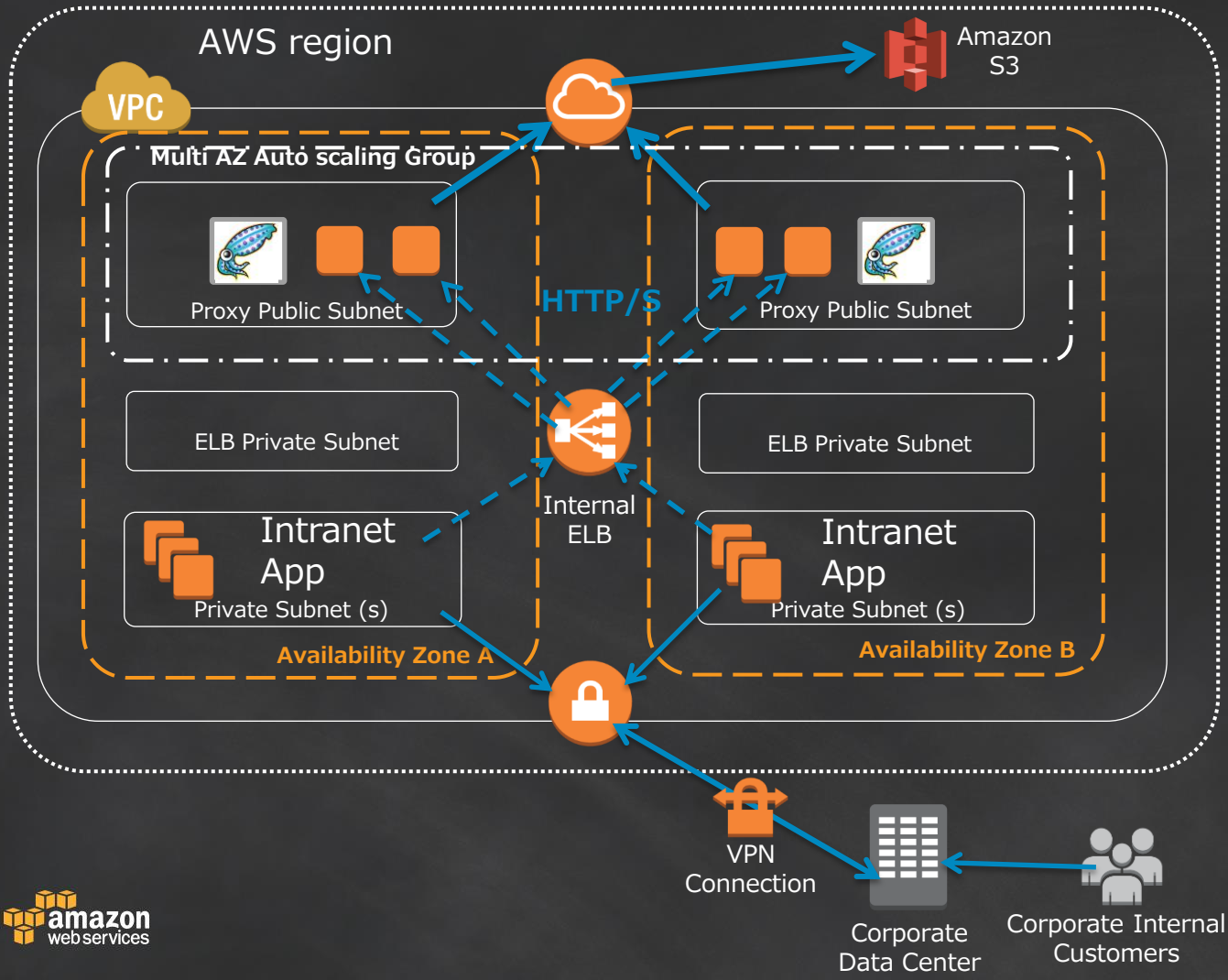
Amazon S3

# 基本となるデータフロー 嫌われる構成でもある



# インターネットゲートウェイを通すためにプロキシを設置する

- アプリ（プライベートサブネットやオンプレミス）とIGWの間にプロキシを配置
- 全てのHTTP(S)トラフィックはS3のように特定のURL宛のみ許可
- プライベートサブネットからはIGWに経路を設定しない
- プロキシへはセキュリティグループで制御する




Proxyとアプリの間にInternal ELBを配置した例

# バックホームパターン


Amazon VPC


Router


Internet Gateway


Customer Gateway

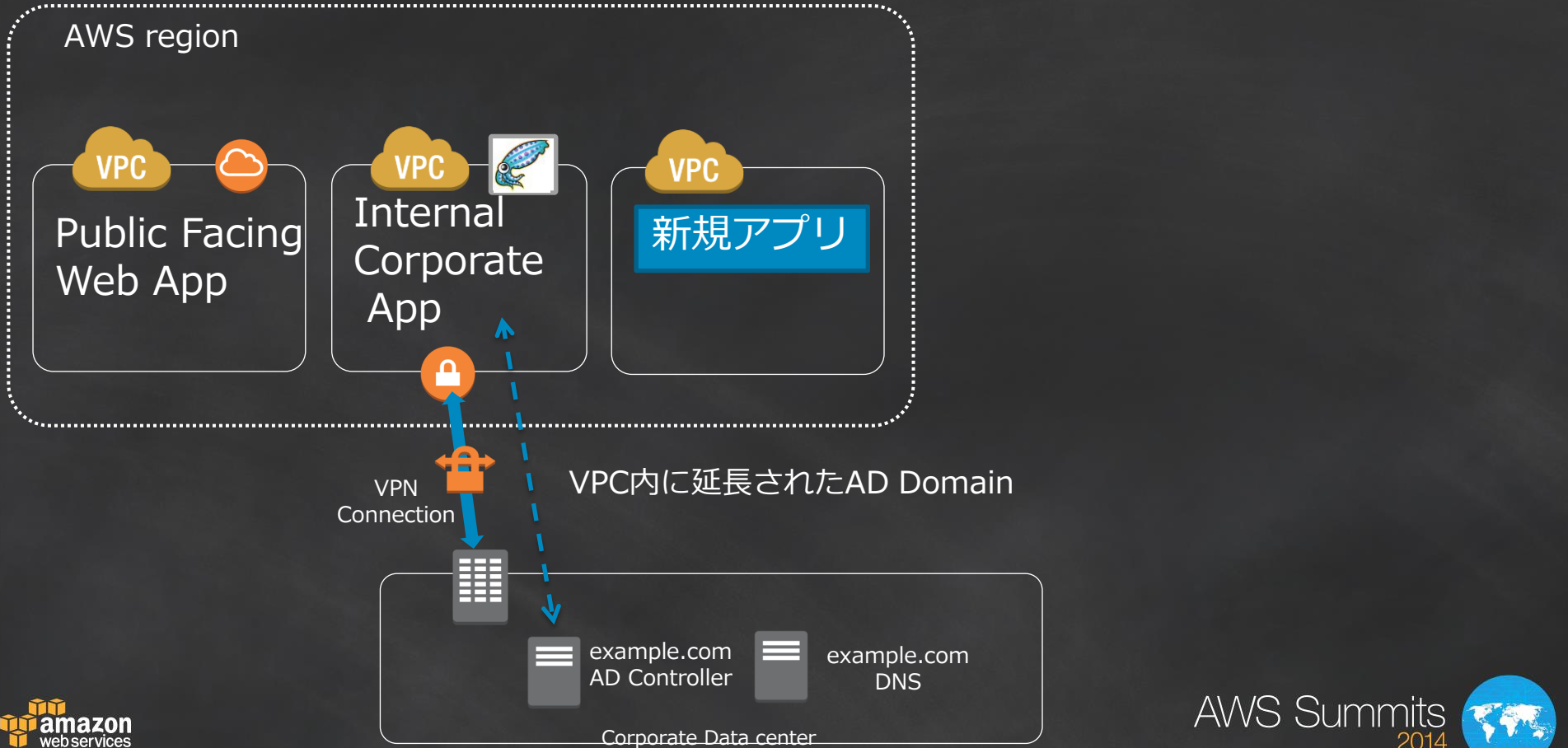

Subnet

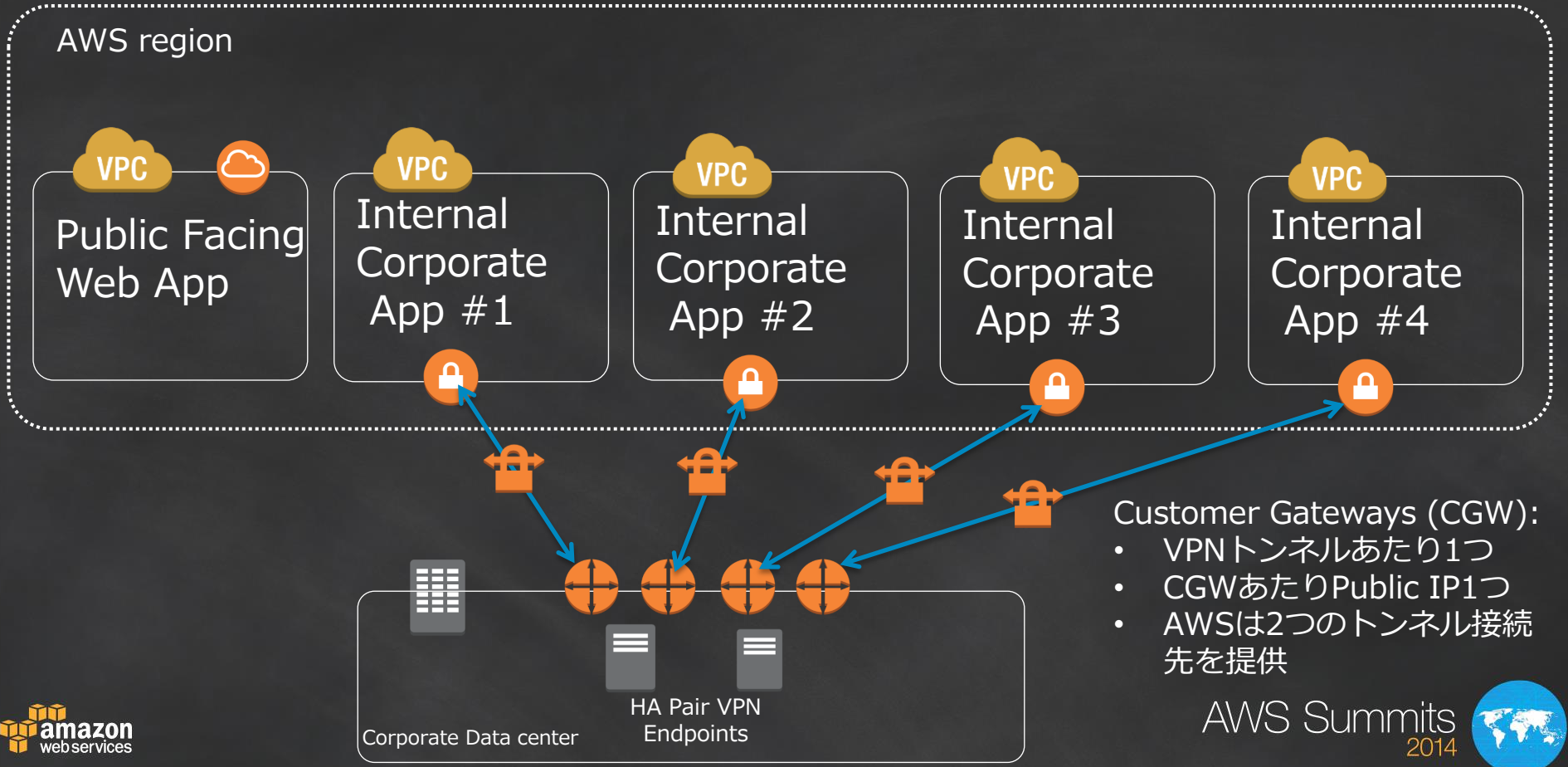

Virtual Private Gateway


VPN Connection

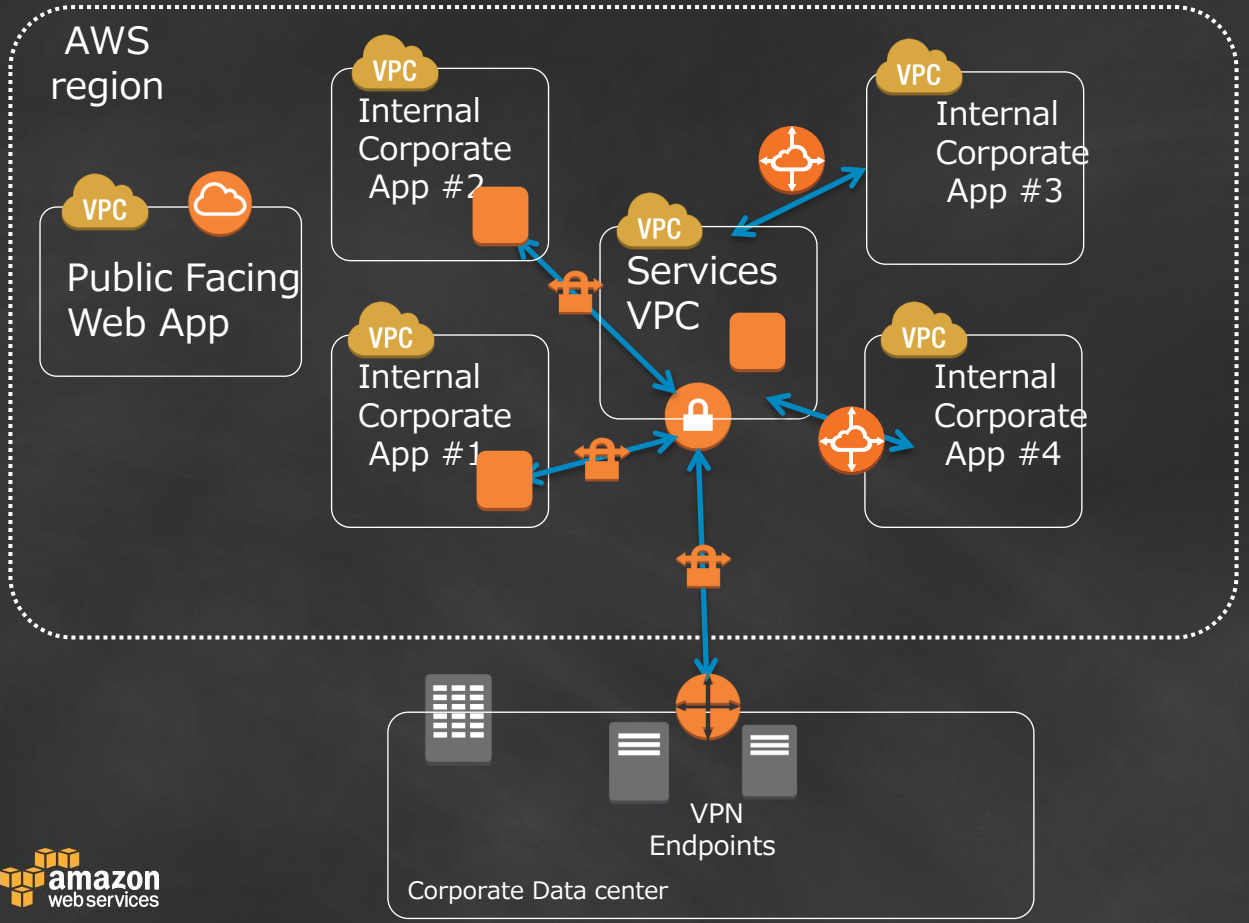
Route Table


Elastic Network Interface


AWS region
VPC
Public Facing Web App
VPC
Internal Corporate App
VPC
新規アプリ
VPN Connection
VPC内に延長されたAD Domain
example.com AD Controller
example.com DNS
Corporate Data center


AWS region
VPC
Public Facing Web App
VPC
Internal Corporate App #1
VPC
Internal Corporate App #2
VPC
Internal Corporate App #3
VPC
Internal Corporate App #4
Corporate Data center
HA Pair VPN Endpoints
Customer Gateways (CGW):
• VPNトンネルあたり1つ
• CGWあたりPublic IP1つ
• AWSは2つのトンネル接続先を提供

# VPNハブ＆スポーク

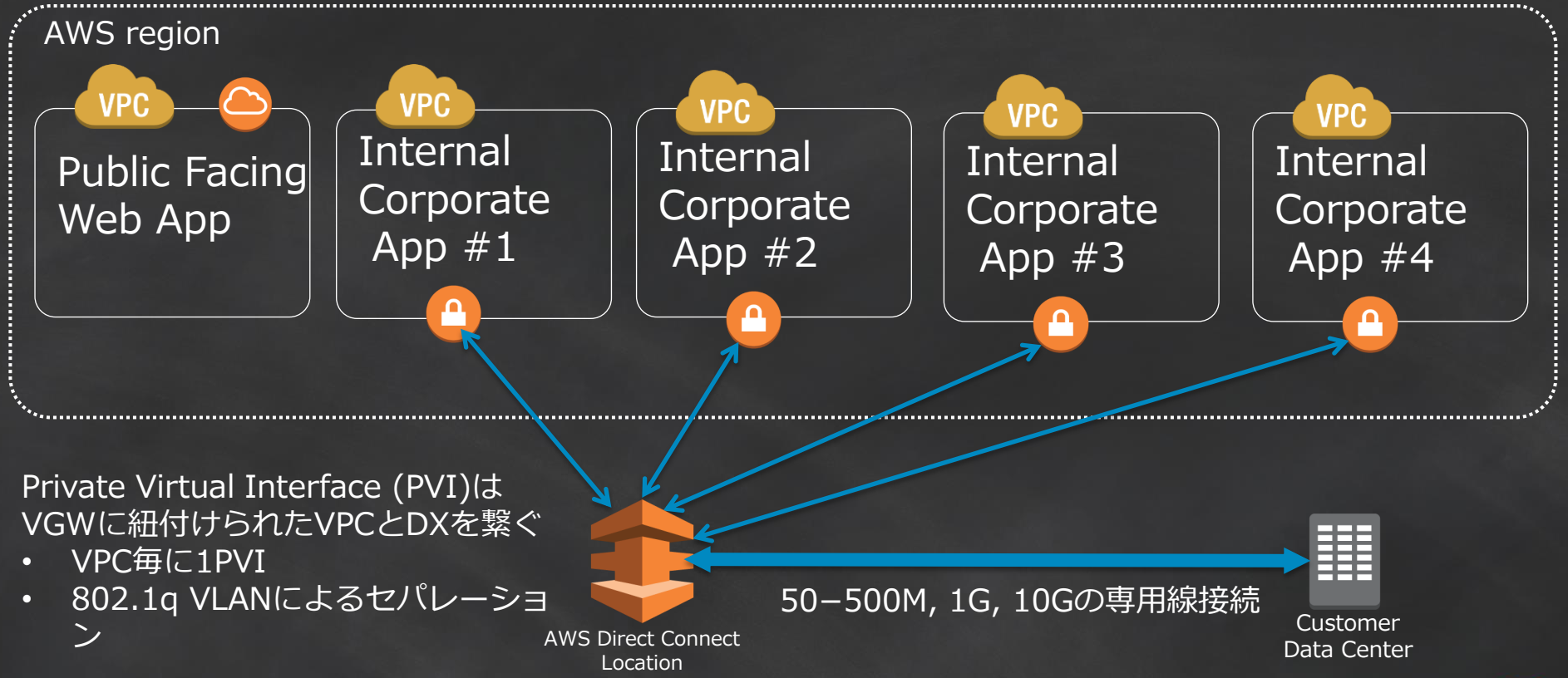



- Service VPCには他VPCで共通で使う機能を持たせる
- VPN
  - アプリ用VPC内インスタンスをCGWとする
  - HAのためには2つ目のインスタンスを追加する
- VPC Peering
  - ピアリング先を経由した先とは通信できない

# AWS Direct Connect (DX)

- 各リージョンへの専用線接続口
- 複数のVPCまたは、複数のパブリック接続インターフェースをのせられる
- インターネットに比べて
  - 安価なアウトバウンドトラフィック料金（インは同じく無料です）
  - 安定したパフォーマンス
- 複数のアカウントで共有可能
- 冗長接続を選択可能

# AWS Direct Connect（DX）によるシンプルな方法



# Multiple VPCs Over AWS Direct Connect

| Customer Interface 0/1.104 | |
|---|---|
| VLAN Tag | 104 |
| BGP ASN | 65004 |
| BGP Announce | Customer Internal |
| Interface IP | 169.254.251.6/30 |

| Private Virtual Interface 1 | |
|---|---|
| VLAN Tag | 101 |
| BGP ASN | 10124 |
| BGP Announce | 10.1.0.0/16 |
| Interface IP | 169.254.251.1/30 |

| Private Virtual Interface 2 | |
|---|---|
| VLAN Tag | 102 |
| BGP ASN | 10124 |
| BGP Announce | 10.2.0.0/16 |
| Interface IP | 169.254.251.5/30 |

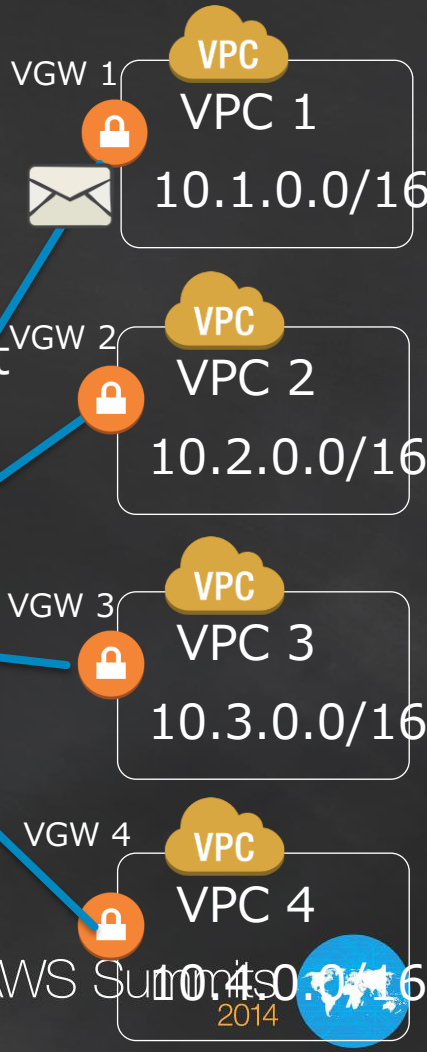
| Private Virtual Interface 3 | |
|---|---|
| VLAN Tag | 103 |
| BGP ASN | 10124 |
| BGP Announce | 10.3.0.0/16 |
| Interface IP | 169.254.251.9/30 |

| Private Virtual Interface 4 | |
|---|---|
| VLAN Tag | 104 |
| BGP ASN | 10124 |
| BGP Announce | 10.4.0.0/16 |
| Interface IP | 169.254.251.13/30 |

Customer Internal Network

**Route Table**

| Destination | Target |
|---|---|
| 10.1.0.0/16 | PVI 1 |
| 10.2.0.0/16 | PVI 2 |
| 10.3.0.0/16 | PVI 3 |
| 10.4.0.0/16 | PVI 4 |

# Public AWS + VPCs Over AWS Direct Connect

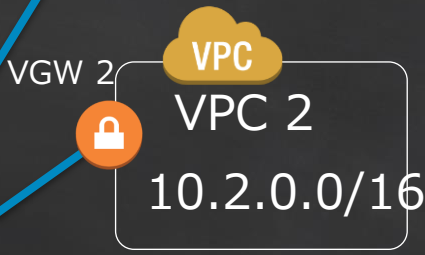
| Customer Interface 0/1.501 | |
|---|---|
| VLAN Tag | 501 |
| BGP ASN | 65501 (or Public) |
| BGP Announce | Customer Public |
| Interface IP | Public /30 Provided |

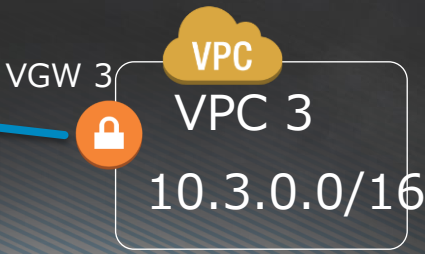
| Public Virtual Interface 1 | |
|---|---|
| VLAN Tag | 501 |
| BGP ASN | 10124 |
| BGP Announce | AWS Regional Public CIDRs |
| Interface IP | Public /30 Provided |

Customer Internal Network

| Route Table | |
|---|---|
| Destination | Target |
| 10.1.0.0/16 | PVI 1 |
| 10.2.0.0/16 | PVI 2 |
| 10.3.0.0/16 | PVI 3 |
| 10.4.0.0/16 | PVI 4 |
| Public AWS | PVI 5 |

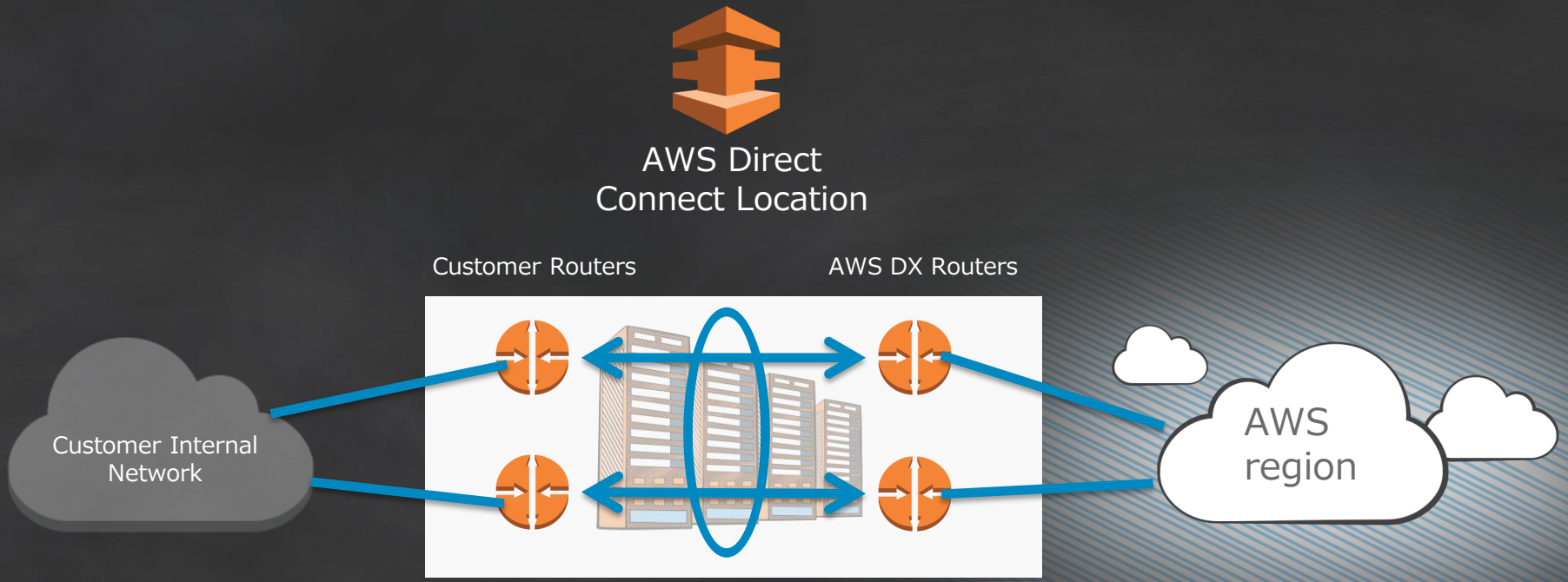

NAT / PAT Security Layer



amazon web services




# 複数のDXおよびVPN接続にはBGPで対応

- Active / Active
- Active / Passive

# キャリアWANでの拡張

# パターン：Directory and Name Services


Amazon VPC


Router


Internet Gateway


Customer Gateway


Subnet


Virtual Private Gateway


VPN Connection

Route Table


Elastic Network Interface

- Active Directory on AWS (TA-04)：吉松龍輝
  - 本セッションでは Active Directory を Amazon Web Services (AWS) 上で運用する際の考慮事項についてお話します。自社データ センターと AWS との相互運用、ドメインとサイトに関する設計の考え方、名前解決の設定、バックアップにおける注意事項など、Active Directory の運用時に必要となるノウハウについてご紹介いたします。

# まとめ

- リソースを可視化し再現可能なシステム開発にはタグ、CloudFormation
- ネットワークと開発担当の権限分離をIAMで行う
- VPCはサブネット単位で制御する
- VPN、専用線で拡張可能

# エンタープライズシステムはすでに現実

- AWSを使えばエンタープライズシステムの構築をあらゆるユーザが
  - もっと簡単に
  - もっと安全に
- AWSをつかったエンタープライズシステムならば
  - 今あるシステムに加えて
  - 現実的な時間
  - 現実的なコスト