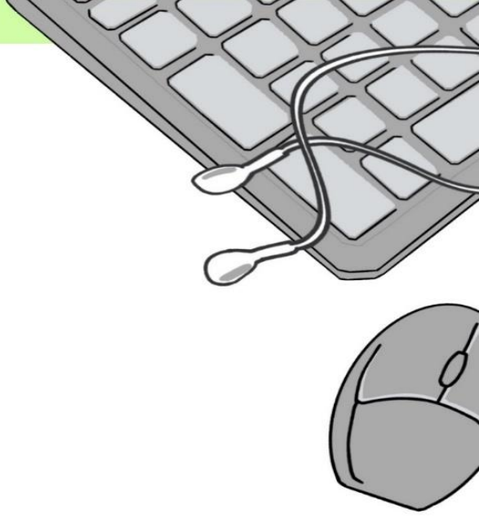
Amazon VPCの概要

VPCとは



- Amazon Virtual Private Cloud（Amazon VPC）
- AWSクラウド上で仮想ネットワークを作成するサービス
- インターネットや他のネットワークと分離された独立したネットワークを構築可能
- EC2やLambda、RDS等のAWSサービスをVPC上に適切に配置してアプリケーションを構築していく
- AWSアカウント作成時、各リージョンにデフォルトのVPCが作成されている
- ファイアウォールのような機能でセキュリティを高めることが可能

VPCの構成要素

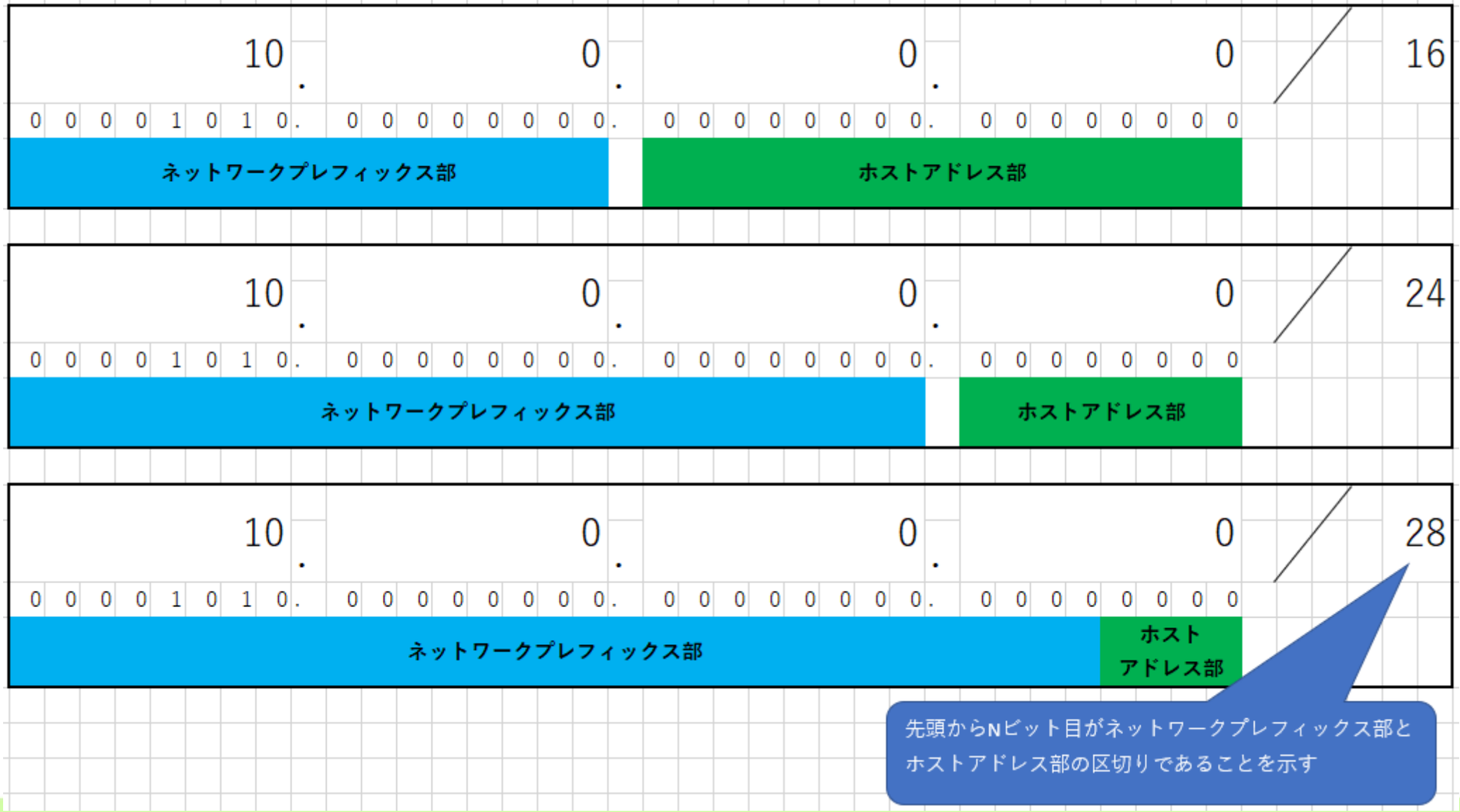
- CIDRブロック
- サブネット
- インターネットゲートウェイ
- NATゲートウェイ
- VPCエンドポイント
- ルートテーブル
- ネットワークACL
- セキュリティグループ

CIDRブロック

- ネットワークを識別するために使用されるIPアドレスの範囲を表すために使用される記法
- 記述例：10.0.0.0/16、10.0.1.0/24、XXX.XXX.XXX.XXX/XX
- IPアドレスの使用量を削減し、ルーティングの効率化が目的
- VPCでは、CIDRブロックを指定してIPアドレス範囲を定義する
- サブネットやルートテーブル等もCIDRブロックに基づいて設定される
- VPCではXXX.XXX.XXX.XXX/16 ～ XXX.XXX.XXX.XXX/28の範囲で指定が可能

CIDRブロック

- 末尾の数字により、ネットワークプレフィックス部とホストアドレス部に分割される



CIDRブロック

- ホストアドレス部の領域の中で、IPアドレスを設定する

10	.	0	.	0	.	0	/	16
00001010	.	00000000	.	00000000	.	00000000		10.0.0.0
00001010	.	00000000	.	00000000	.	00000001		10.0.0.1
00001010	.	00000000	.	00000000	.	00000010		10.0.0.2
00001010	.	00000000	.	00000000	.	00000011		10.0.0.3
00001010	.	00000000	.	00000000	.	00000100		10.0.0.4
						...		...
						...		...
00001010	.	00000000	.	11111111	.	11111100		10.0.255.252
00001010	.	00000000	.	11111111	.	11111101		10.0.255.253
00001010	.	00000000	.	11111111	.	11111110		10.0.255.254
00001010	.	00000000	.	11111111	.	11111111		10.0.255.255
ネットワークプレフィックス部				ホストアドレス部				

65536個のIPアドレスを表現可能

CIDRブロック

- ホストアドレス部の領域の中で、IPアドレスを設定する

10	0	0	0	/ 16	
00001010.	00000000.	00000000.	00000000	~~10.0.0.0~~	ネットワークアドレスのため使用不可
00001010.	00000000.	00000000.	00000001	~~10.0.0.1~~	AWSが管理上使用するアドレスのため使用不可
00001010.	00000000.	00000000.	00000010	~~10.0.0.2~~	AWSが管理上使用するアドレスのため使用不可
00001010.	00000000.	00000000.	00000011	~~10.0.0.3~~	AWSが管理上使用するアドレスのため使用不可
00001010.	00000000.	00000000.	00000100	10.0.0.4	
			. . .	. . .	
			. . .	. . .	
00001010.	00000000.	11111111.	11111100	10.0.255.252	
00001010.	00000000.	11111111.	11111101	10.0.255.253	
00001010.	00000000.	11111111.	11111110	10.0.255.254	
00001010.	00000000.	11111111.	11111111	~~10.0.255.255~~	ブロードキャストアドレスのため使用不可
ネットワークプレフィックス部		ホストアドレス部			

65536個のIPアドレスを表現可能

CIDRブロック

- ホストアドレス部の領域の中で、IPアドレスを設定する

10	0	0	0	/ 24
00001010	00000000	00000000	00000000	10.0.0.0
00001010	00000000	00000000	00000001	10.0.0.1
00001010	00000000	00000000	00000010	10.0.0.2
00001010	00000000	00000000	00000011	10.0.0.3
00001010	00000000	00000000	00000100	10.0.0.4
			...	...
			...	...
00001010	00000000	00000000	11111100	10.0.0.252
00001010	00000000	00000000	11111101	10.0.0.253
00001010	00000000	00000000	11111110	10.0.0.254
00001010	00000000	00000000	11111111	10.0.0.255
ネットワークプレフィックス部			ホストアドレス部	

256個のIPアドレスを表現可能

CIDRブロック

- ホストアドレス部の領域の中で、IPアドレスを設定する

10	0	0	0	24	
0 0 0 0 1 0 1 0	0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0	~~10.0.0.0~~	ネットワークアドレスのため使用不可
0 0 0 0 1 0 1 0	0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 1	~~10.0.0.1~~	AWSが管理上使用するアドレスのため使用不可
0 0 0 0 1 0 1 0	0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0	0 0 0 0 0 0 1 0	~~10.0.0.2~~	AWSが管理上使用するアドレスのため使用不可
0 0 0 0 1 0 1 0	0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0	0 0 0 0 0 0 1 1	~~10.0.0.3~~	AWSが管理上使用するアドレスのため使用不可
0 0 0 0 1 0 1 0	0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0	0 0 0 0 0 1 0 0	10.0.0.4	
			. . .	. . .	
			. . .	. . .	
0 0 0 0 1 0 1 0	0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0	1 1 1 1 1 1 0 0	10.0.0.252	
0 0 0 0 1 0 1 0	0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0	1 1 1 1 1 1 0 1	10.0.0.253	
0 0 0 0 1 0 1 0	0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0	1 1 1 1 1 1 1 0	10.0.0.254	
0 0 0 0 1 0 1 0	0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0	1 1 1 1 1 1 1 1	~~10.0.0.255~~	ブロードキャストアドレスのため使用不可
ネットワークプレフィックス部			ホストアドレス部		

256個のIPアドレスを表現可能

CIDRブロック

- ホストアドレス部の領域の中で、IPアドレスを設定する

10	0	0	0	28
0 0 0 0 1 0 1 0	0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0	10.0.0.0
0 0 0 0 1 0 1 0	0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 1	10.0.0.1
0 0 0 0 1 0 1 0	0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0	0 0 0 0 0 0 1 0	10.0.0.2
0 0 0 0 1 0 1 0	0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0	0 0 0 0 0 0 1 1	10.0.0.3
0 0 0 0 1 0 1 0	0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0	0 0 0 0 0 1 0 0	10.0.0.4
			. . .	. . .
			. . .	. . .
0 0 0 0 1 0 1 0	0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0	0 0 0 0 1 1 0 0	10.0.0.12
0 0 0 0 1 0 1 0	0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0	0 0 0 0 1 1 0 1	10.0.0.13
0 0 0 0 1 0 1 0	0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0	0 0 0 0 1 1 1 0	10.0.0.14
0 0 0 0 1 0 1 0	0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0	0 0 0 0 1 1 1 1	10.0.0.15
ネットワークプレフィックス部			ホスト アドレス部	

16個のIPアドレスを表現可能

CIDRブロック

- ホストアドレス部の領域の中で、IPアドレスを設定する

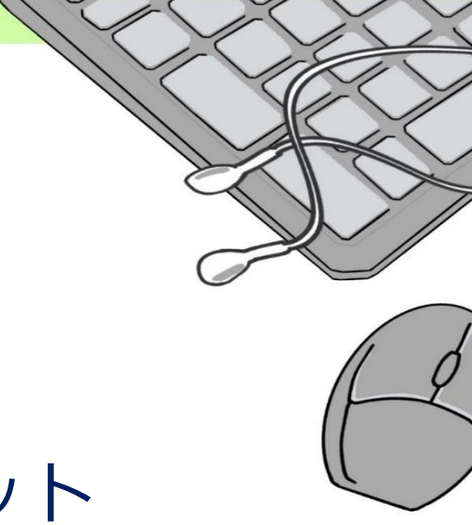
10	0	0	0	/	28	
00001010.	00000000.	00000000.	00000000			
00001010.	00000000.	00000000.	00000000	10.0.0.0		ネットワークアドレスのため使用不可
00001010.	00000000.	00000000.	00000001	10.0.0.1		AWSが管理上使用するアドレスのため使用不可
00001010.	00000000.	00000000.	00000010	10.0.0.2		AWSが管理上使用するアドレスのため使用不可
00001010.	00000000.	00000000.	00000011	10.0.0.3		AWSが管理上使用するアドレスのため使用不可
00001010.	00000000.	00000000.	00000100	10.0.0.4		
				...		
				...		
00001010.	00000000.	00000000.	00001100	10.0.0.12		
00001010.	00000000.	00000000.	00001101	10.0.0.13		
00001010.	00000000.	00000000.	00001110	10.0.0.14		
00001010.	00000000.	00000000.	00001111	10.0.0.15		ブロードキャストアドレスのため使用不可
ネットワークプレフィックス部			ホストアドレス部			

16個のIPアドレスを表現可能

サブネット

- VPC内のIPアドレス範囲をさらに分割して、複数のネットワークセグメントを作成するための仮想的なサブネットワーク
- サブネットはパブリックサブネットとプライベートサブネットの2種類がある
- サブネットはVPCとAZ、CIDRブロックを指定して作成する

サブネット



- **パブリックサブネット**

  インターネットからネットワーク内にアクセス可能なサブネット

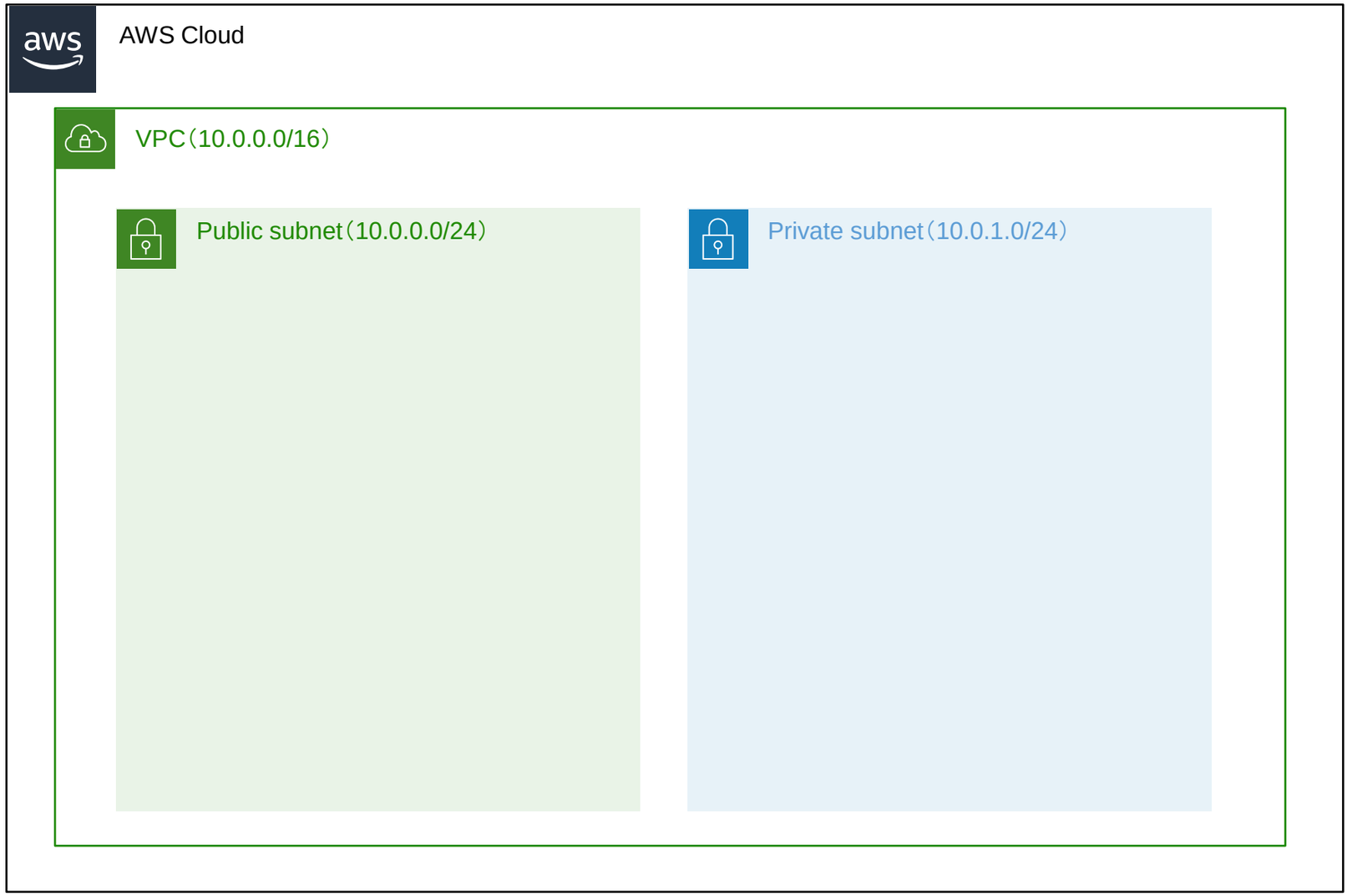
  Webサーバーを配置したりする

- **プライベートサブネット**

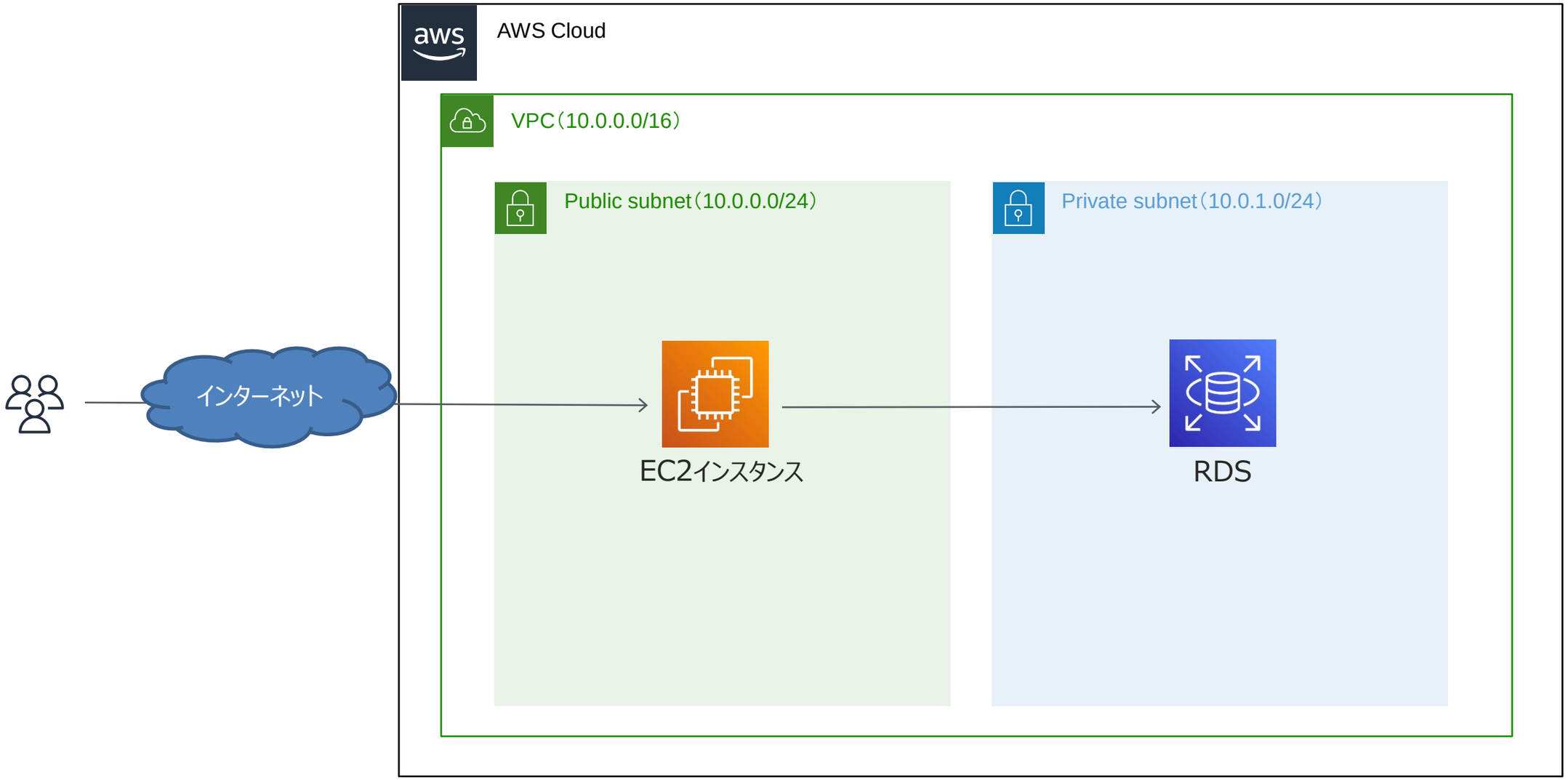
  インターネットと隔離されたサブネット

  DBサーバーを配置したりする

VPCとサブネットの構成例



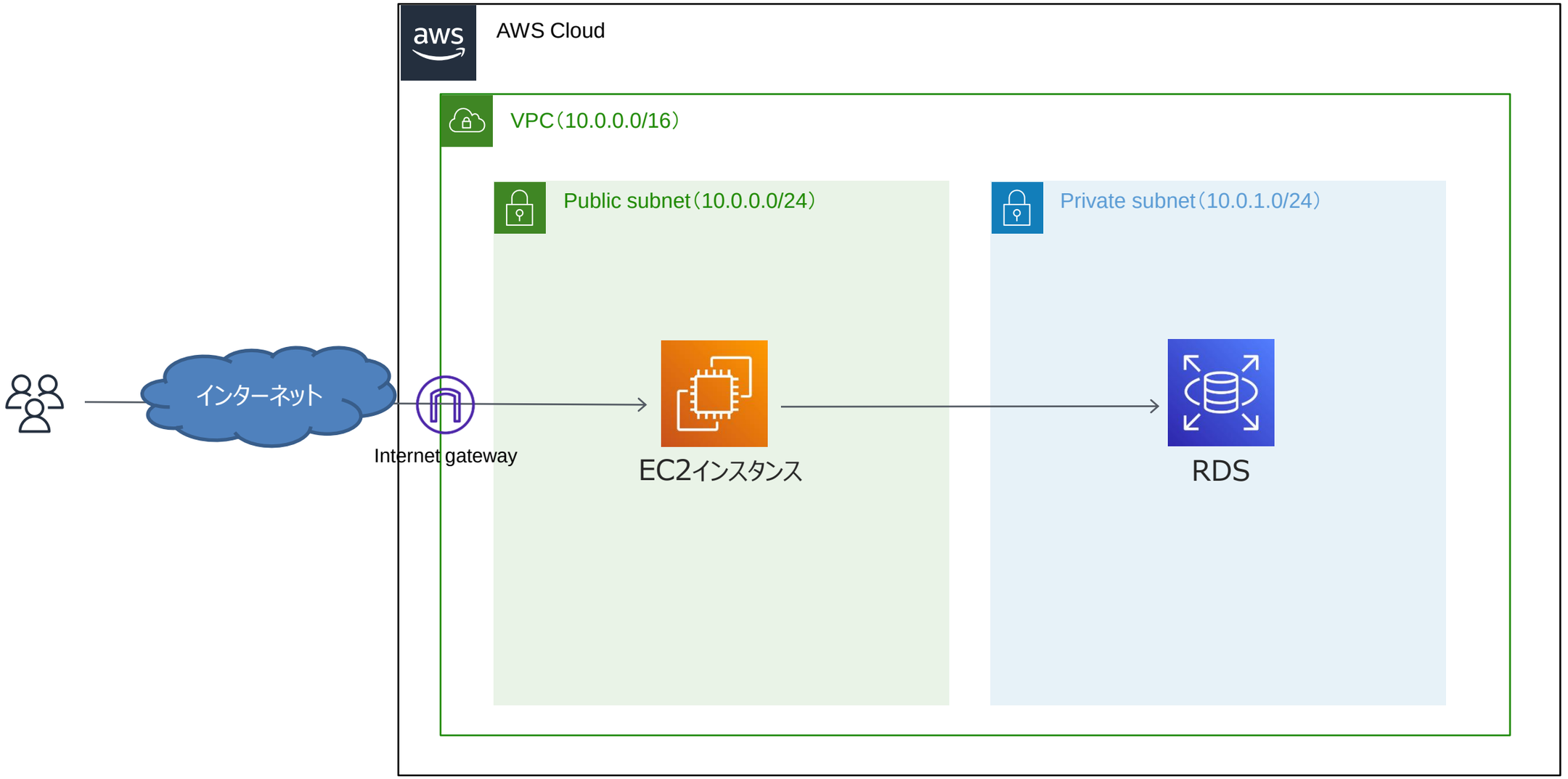
VPCとサブネットの構成例



インターネットゲートウェイ

- インターネットからサブネット内のリソースにアクセスするためのゲートウェイ
- ルートテーブル（後述）でインターネットゲートウェイが設定されているサブネットをパブリックサブネットという
- ルートテーブル（後述）でインターネットゲートウェイが設定されていないサブネットをプライベートサブネットという

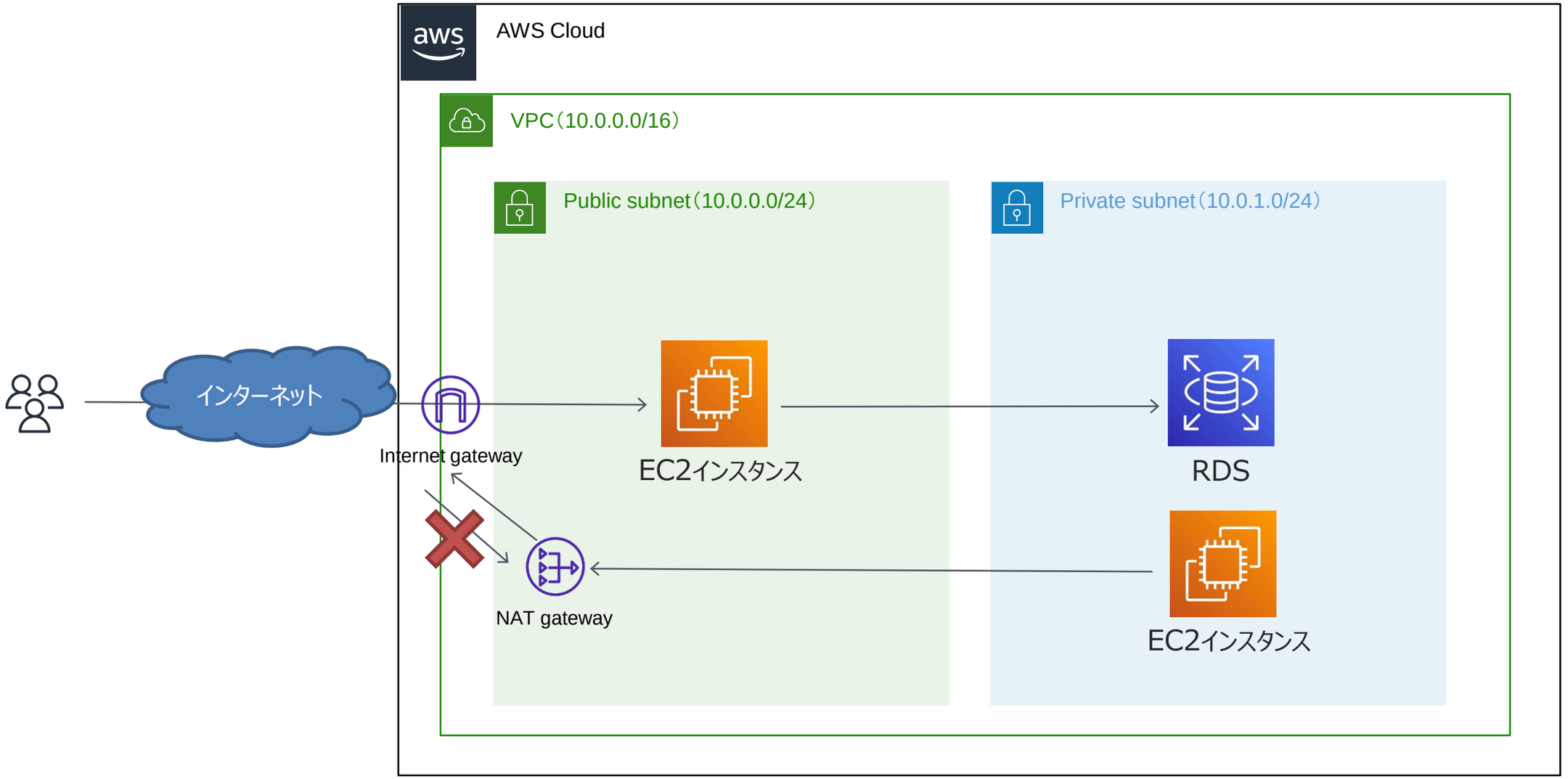
VPCとサブネットの構成例



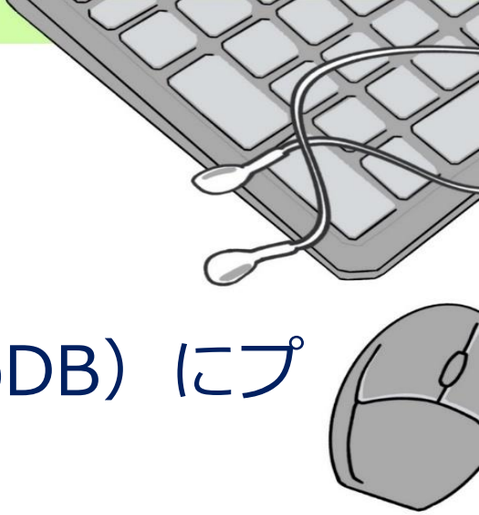
NATゲートウェイ

- プライベートサブネット内のリソースがインターネットにアクセスするためのゲートウェイ
- NATゲートウェイを使用することで、プライベートサブネットからインターネットにアクセスはできるが、インターネットからプライベートサブネットにはアクセスできない環境を構築可能
- プライベートサブネット内のリソースのソフトウェアインストールや、セキュリティパッチ適用等で使用する
- NATゲートウェイはパブリックサブネットに設置される

VPCとサブネットの構成例

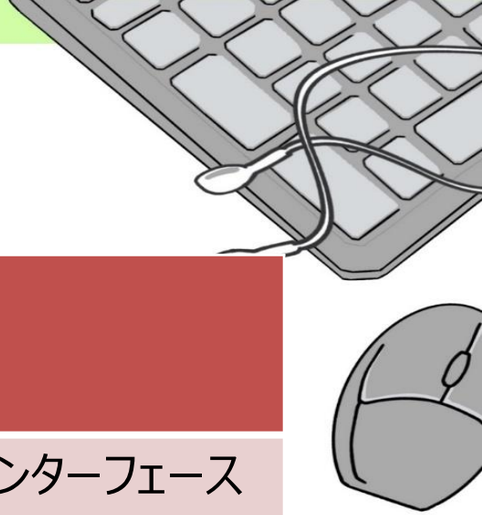


VPCエンドポイント



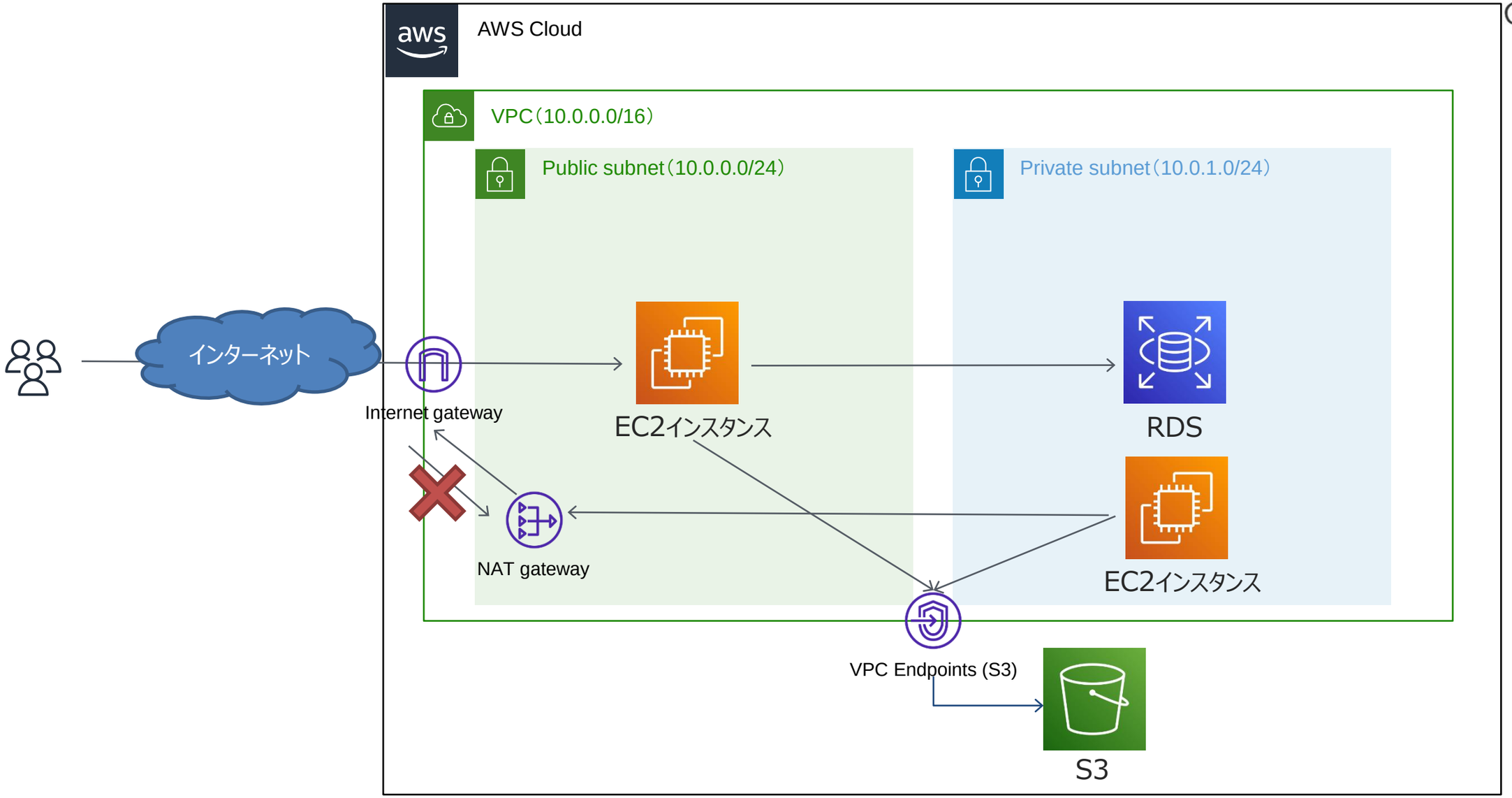
- VPC内のリソースが、AWSのサービス（例えばS3やDynamoDB）にプライベートにアクセスするための仕組み
- VPCエンドポイントを使用することで、VPC内のリソースがインターネット経由で外部のサービスにアクセスする必要が無く、セキュリティを高めることが可能
- VPCエンドポイントにはゲートウェイ型とインターフェース型の2種類が存在する

VPCエンドポイント



	ゲートウェイ型	インターフェース型
特徴	インターネットゲートウェイやNATゲートウェイと同じように利用できるエンドポイント	AWS PrivateLinkを使用したインターフェース型のエンドポイント
利用できるサービス	S3,DynamoDBのみ	S3,DynamoDB含む50以上のサービス
料金	無料	有料
設定	ルートテーブル	セキュリティグループ
VPC外からのアクセス	できない	できる

VPCとサブネットの構成例

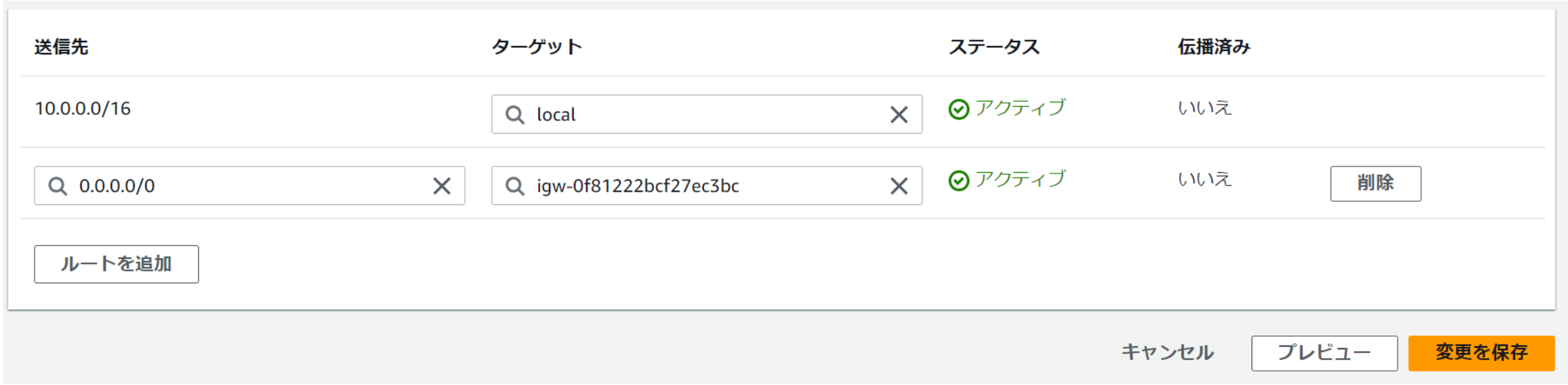


ルートテーブル

- サブネット内のトラフィックを転送するためのルーティング規則
- 送信先のCIDRブロックに基づくトラフィックを、どのターゲットに転送するのかを定義する
- サブネットは必ず1つのルートテーブルに関連付けられる
- インターネットゲートウェイやNATゲートウェイを使うには、ルートテーブルへのルート追加が必要

ルートテーブル

- 以下のルートテーブル設定は以下の意味を持つ
- 10.0.0.0/16(10.0.0.0～10.0.255.255)のトラフィックをlocalに転送する
- 0.0.0.0/0のトラフィックをインターネットゲートウェイに転送する
- 最初にマッチしたルートに基づいてトラフィックが転送される

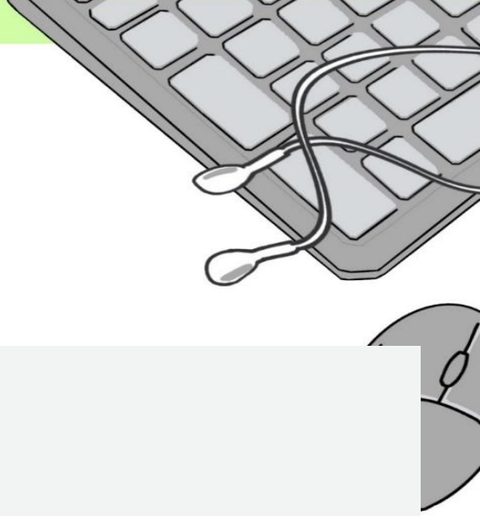


ネットワークACL

- サブネットのセキュリティを強化する目的で使用され、サブネット内のトラフィック転送を許可する／拒否するを定義する
- インバウンドトラフィック（サブネットに来るトラフィック）とアウトバウンドトラフィック（サブネットから出るトラフィック）の両方に対して設定する（ステートレス）
- CIDRブロック、ポート番号等の条件に基づいてトラフィックを許可または拒否することができる
- 例えば、Webサイトをホストするサブネットには、HTTPやHTTPSのトラフィックを許可するルールを設定し、それ以外のトラフィックは全て拒否するような設定を行ったりする
- セキュリティグループ（後述）と非常に似ているので紛らわしい

ネットワークACL

- インバウンドルールの設定一例



インバウンドルールを編集 情報

インバウンドルールは VPC への到達が許可された受信トラフィックを制御します。

ルール番号 情報	タイプ 情報	プロトコル 情報	ポート範囲 情報	送信元 情報	許可/拒否 情報	
100	HTTP (80)	TCP (6)	80	0.0.0.0/0	許可	削除
200	HTTPS (443)	TCP (6)	443	0.0.0.0/0	許可	削除
300	SSH (22)	TCP (6)	22	1.2.3.4/32	許可	削除
*	すべてのトラ...	すべて	すべて	0.0.0.0/0	拒否	

新しいルールを追加　ルール番号で並べ替え

・サブネットに来る、HTTP、HTTPSのトラフィックを許可する
・サブネットに来る、自宅（1.2.3.4とする）からのSSHトラフィックを許可する
・サブネットに来る、それ以外の全てのトラフィックを拒否する

ネットワークACL

- アウトバウンドルールの設定一例

アウトバウンドルールを編集 情報

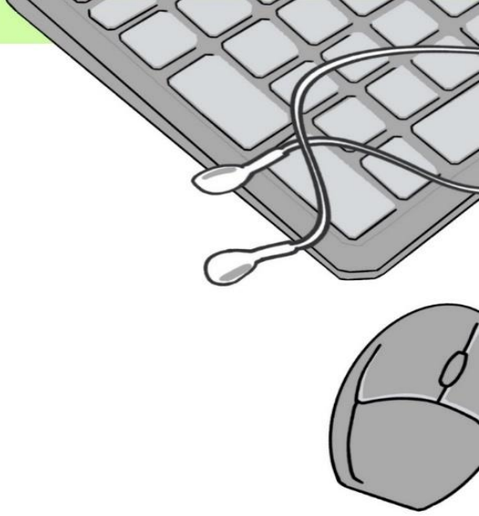
アウトバウンドルールは VPC からの出力を許可された送信トラフィックを制御します。

ルール番号 情報	タイプ 情報	プロトコル 情報	ポート範囲 情報	送信先 情報	許可/拒否 情報	
100	すべてのトラフ...	すべて	すべて	0.0.0.0/0	許可	削除
*	すべてのトラフ...	すべて	すべて	0.0.0.0/0	拒否	

新しいルールを追加　ルール番号で並べ替え

・サブネットから出る全てのトラフィックを許可する

セキュリティグループ



- AWSリソース（EC2やRDS）のセキュリティを強化する目的で使用され、AWSリソースのトラフィック転送許可を定義する（拒否は設定できない）
- Linuxのiptablesと似たような機能を提供する
- インバウンドトラフィック（サブネットに来るトラフィック）とアウトバウンドトラフィック（サブネットから出るトラフィック）の両方に対して設定する（ステートフル）
- CIDRブロック、ポート番号等の条件に基づいてトラフィック許可を設定することができる
- 例えば、Webサーバーが起動しているEC2にはHTTPやHTTPSのトラフィックを許可するルールを設定し、MySQLが起動しているEC2はMySQLのトラフィックのみ許可する設定を行ったりする
- ネットワークACL（前述）と非常に似ているので紛らわしい

セキュリティグループ

- インバウンドルールの設定一例

インバウンドルール 情報

セキュリティグループルール ID	タイプ 情報	プロトコル 情報	ポート範囲 情報	ソース 情報		説明 - オプション 情報	
sgr-0c9c27093dd6a4ecd	SSH	TCP	22	カスタム	1.2.3.4/32		削除
sgr-07ee4a919d047a554	HTTPS	TCP	443	カスタム	0.0.0.0/0		削除
sgr-03c87558f5128fdf0	HTTP	TCP	80	カスタム	0.0.0.0/0		削除

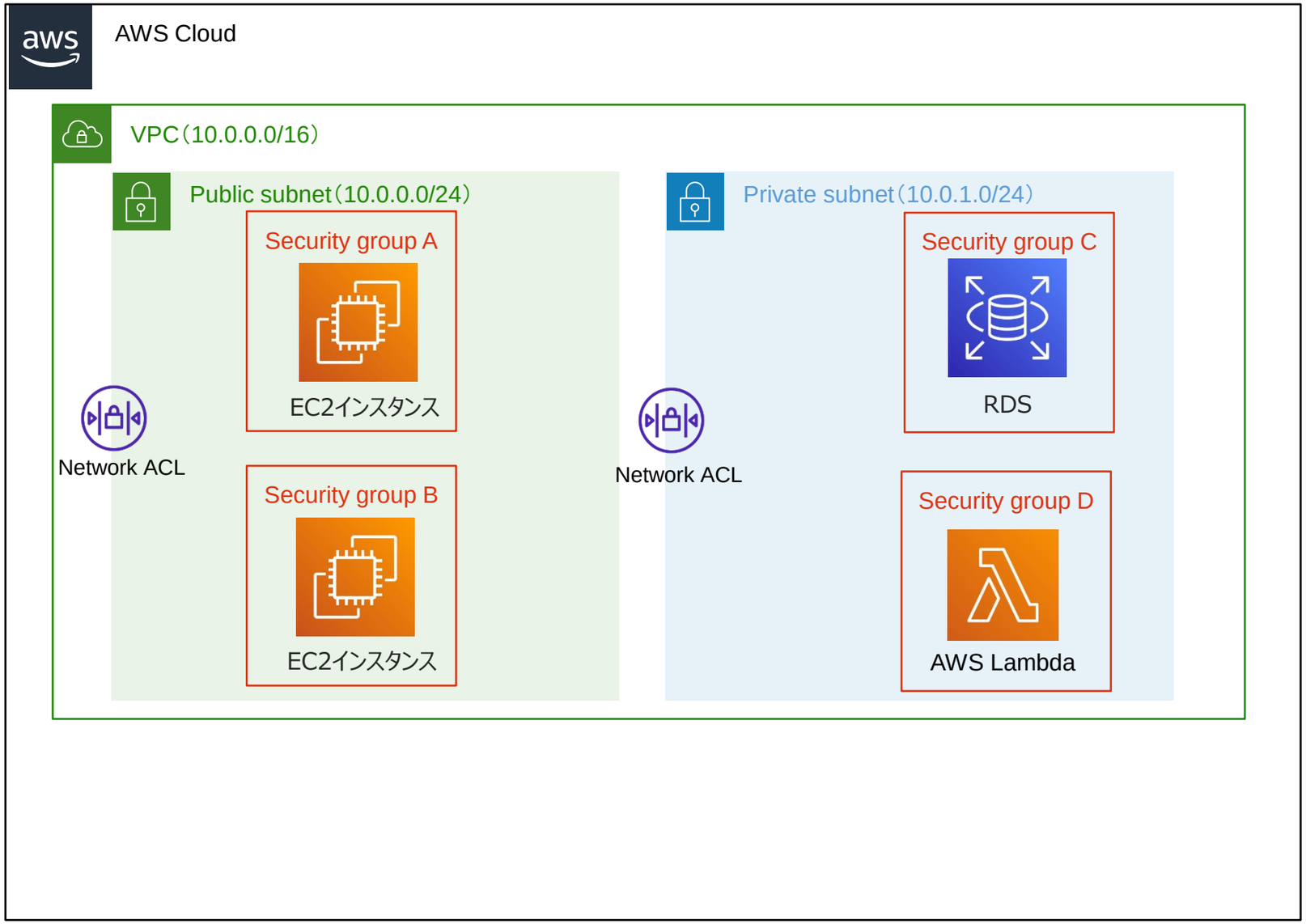
・AWSリソースに来る、自宅（1.2.3.4とする）からのSSHトラフィックを許可する
・AWSリソースに来る、HTTP、HTTPSのトラフィックを許可する
・ここに書かれていない全てのトラフィックを拒否する

セキュリティグループ

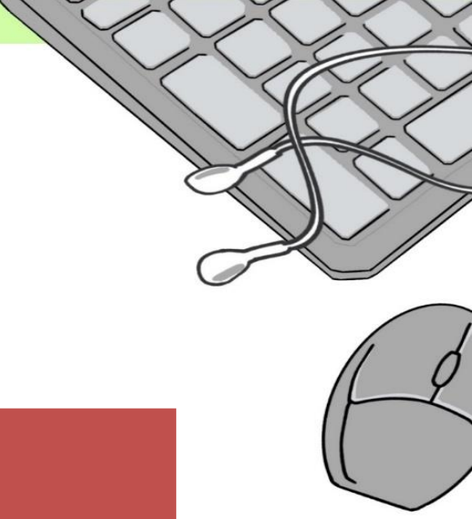
- アウトバウンドルールの設定一例

・AWSリソースから出る全てのトラフィックを許可する

VPCとサブネットの構成例



その他のVPCの構成要素



名称	機能
VPCピアリング	2つのVPCを接続することで、異なるVPC間での通信を可能にする機能
仮想プライベートゲートウェイ カスタマーゲートウェイ	インターネットからVPC内のリソースに安全にアクセスするための仮想的なVPN接続を提供する機能。カスタマーゲートウェイを使用することで、オンプレミスネットワークとのVPN接続も可能
トランジットゲートウェイ	複数のVPCを結びつけ、それら間でトラフィックを受け渡すためのサービス。複数のVPCの接続設定を一元化して管理可能

VPCの料金体系

- VPCやサブネットの作成は無料
- データ転送量に応じた従量課金
- NATゲートウェイの利用時間分の従量課金
- VPCエンドポイント（インターフェース型）の利用時間分の従量課金
- VPCピアリング、仮想ネットワークゲートウェイ等を使用したVPN接続、トランジットゲートウェイ使用時にも別途料金が発生
- その他「AWS　VPC　料金」で検索（実際のページを見ながら解説）