AWS Advanced Networking Speciality（ANS）

・VPCエンドポイントでVPCからS3、DynamoDBなどに接続する際は、VPCとサービスが同じリージョンにある必要がある。
・CIDR block 10.0.0.0/24,reserved ID are:
　10.0.0.0: Network address
　10.0.0.1: Reserved by AWS for the VPC router
　10.0.0.2: Reserved by AWS for mapping to Amazon-provided DND
　10.0.0.3: Reserved by AWS for future use
　10.0.0.255 Network broadcast address
・VPC内のインスタンス同士は接続することができる（パブリックサブネットとプライベートサブネットでも）
**　ただし、両方のインスタンスのセキュリティグループが、相手からのICMPトラフィックを許可していること。**
・NATGatewayの通信はアウトバウンドのみを許可するが、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上のエンドポイントに対して接続を開始した場合に、その
　接続に対する応答は許可される。（例：LambdaがChatGPTに対して行った応答は受け取ることができる。）
・ENIが含む情報
1. 一つのプライマリIPv4アドレス（ENIを識別するためのネットワークアドレス）
2. 一つまたはそれ以上のセカンダリIPv4アドレス
3. 一つまたはそれ以上のIPv6アドレス
4. 一つまたはそれ以上のElastic IPアドレス（IPv4）
5. 一つのMACアドレス
6. セキュリティグループ
7. サブネットID

・DPDKはCPUの使用効率を向上させ、パケット処理の遅延を減らすためのソフトウェア開発キット
・Egress-Onlyインターネットゲートウェイは IPv6経由でインターネットからの送信を可能にする。
・プレフィックスは、CIDRのスラッシュ以降の部分（例：10.0.0.0/24の場合、24の部分をプレフィックスという）
・NAT Gatewayに割り当てられた単一のIPアドレスからの接続を許可するようにオンプレミスのファイアウォールを個性することで、運用上のオーバヘッドを回避することができる
　（VGWもしくはTransitGatewayを介してルーティングできる）
・CloudFormationとAWS Configを組み合わせることで、各AWSリージョンにConfig設定が可能
・バケットポリシーで暗号化されていない通信を拒否するにはaws:SecureTransport:falseでDenyルールを作成する。
・ローカル端末からパブリックサブネットのEC2にSSH接続するためには、インターネットゲートウェイの作成とルートテーブル設定が必要。
・別リージョンのセキュリティグループを参照することはできない
・Unicast IP：ひとつのサーバがひとつのIPアドレスを保有する
・Anycast IP：複数のサーバが同じIPアドレスを持つ
・サブネット作成時にデフォルトで作成されるメインルートテーブルは変更せずに置いておく。

（ルートテーブルを別途作成して、サブネットにアタッチする）
・インスタンスはサブネットを変更できない
・インスタンスのメタデータ取得（http://169.254.169.254/latest/meta-data/instance-id）ポート 80
・**Wireshark** は、ネットワーク上で送受信されるパケットの詳細をリアルタイムで視覚化し、保存し、分析できる。
（VPC フローログではできない、パケットレベルの情報を見れる。）

〇VPC フローログ

・モニタリングできるログの種類
  ・account-id
  ・protocol
  ・packets
  ・bytes
  ・start
  ・end
  ・action
  ・log-status
  ・srcaddr：ソース IP アドレス
  ・dstaddr：宛先 IP アドレス
  ・Srcport：ソースポート
  ・dstport：宛先ポート
  ・Action：リクエストがセキュリティグループ、ACL により成功もしくは失敗したか
・S3 もしくは CloudWatch Logs に送信できる（Athena もしくは CloudWatch Logs Insights で分析）

〇VPC Traffic Mirroring

・IDS-IPS システムを使いたかったり、レイテンシーを低下させたい場合に使う
・同じトラフィックを別のターゲットにミラーリングする。
（潜在的な脅威を検知できる）
・手順（VPC コンソール画面から設定できる）
  ・ミラーターゲットを作成
  ・トラフィックフィルターを定義
  ・ミラーセッションを作成
・ターゲット（UDP - 4789）
  ・ENI
  ・NLB
・フィルターパラメータ
  ・インバウンドとアウトバウンドで設定
  ・許可と拒否
  ・プロトコル：Layer4
  ・ソースポート範囲、宛先ポート範囲
  ・ソース CIDR 範囲、宛先 CIDR 範囲
・同じ VPC 内で設置できる
（異なる VPC の場合、VPC Peering もしくは Transit Gateway を使う）
・

○Transit Gateway

・リージョン間通信はTransit Gateway Peering（**リージョン間のルートはBGPではなく、Static**にする必要があり）

・中央のVPCに一括管理することで、VPCごとにVPCエンドポイントを作る必要がなくなる。

・有料

・推移的な接続も可能（他 VPCのVPCエンドポイントも使用可能）

・AWS Transit Gateway Network Manager：Transit Gatewayを利用する上で欲しい機能（ネットワークの全体把握、ネットワークトポロジー、TGWルートテーブルの疎通確認、モニタリングetc）を基本無料で提供してくれるTGWを補完するサービス

・**Transit Gatewayはリージョン単位**で動作する（オンプレミスとVPC間の通信を中央で制御する管理者のようなもの）

・ゲートウェイ向けにingressとEgressを分けてルーティングテーブルを作成することができる。

○AWS Resource Access Manager(RAM)

・Transit GatewayはAWS Resource Access Manager (RAM) を使って他のAWSアカウントと共有できる。

・Route53 ResolverルールはRAMで共有できる

○BGP

・iBGP：同じAS内のルート

・eBGP：別のAS間のルート

・ルーティング

  ・重み

  ・ASPATH：

  ・Local Preference：

  ・MED：

○Direct Connect

・ネットワーク要件

  ・Single-mode fiber

    ・1000BASE-LX(1310nm)：1G帯域

    ・10GBASE-LR(1310nm)：10G帯域

    ・1000GBASE-LR4：100G帯域

  ・802.1Q VALN

  ・Auto-negotiationは無効化する

・full-duplexとポート速度を手動で設定する
・Customer RouterはBGPとBGP MD5をサポート。
・（任意）BFDをサポート
・AWS IP Ranges(ip-ranges.json)

・Direct ConnectとVIF
・Public VIF
・グローバルにAWSにアクセスできる
・Public IPでアクセス
・
・Private VIF
・Gatewayエンドポイントへの推移的なアクセスはできない（Interfaceエンドポイントは可能）
・VPC内のDNSサーバーに直接アクセスはできない（Route53 Resolverを使う必要がある）
・

（DirectConnect<->S3,DynamoDB　:Public VIF、DirectConnect<->VPC：PrivateVIF、DIrectConnect<->DirectConnectGateway<->TransitGateway：TransitVIF）
・**DirectConnectでは、AutoNegotiationはオフにし、ポートスピードとFullDuplexを手動で設定**する必要があるl。
・DirectConnect DedicatedConnection：50このVIF作成可能、HostedConnection：1つのHostedConnectionにつきひとつのVIF
・1つのprivate VIFにひとつのDirectConnectGatewayが紐づく。そして、ひとつのDirectConnectGatewayには10このVPCが繋げれる
Dedicated Connectionの場合、50このVIFが作れるから、500このVPCとひとつのDirectConnectが繋げれる
・ひとつのDirectConnectGatewayを2つのDirectConnectRouterで共有することはできるが、DirectConnectRouterと繋がっているオンプレミス同士が
DirectConnectGatewayを介して推移的に通信することはできない
・AWS Direct Connect SiteLinkを使えば、DirectConnectGatewayを介して、2つのオンプレミス感をつなぐことができる
・JumboFrameのサポート（Propagatedルートでのみ利用可能、Staticはふか）
PublicVIF：利用不可、PrivateVIF：利用可能（MTU9001）、TransitVIF：利用可能（MTU1500or8500）
・Direct ConnectのDedicated ConnectionとHosted Connectionの違いは、AWSが専用線を所有するか、DirectConnectパートナーが所有するか。
・BFD(Bidirectional Forwarding Detection)：接続に問題が生じた場合に、1秒以内にフェールオーバーを実現する
（300ms間隔で障害を検知する。）
・10Gbps接続に対してはマルチモードファイバーインターフェースが必要で、1Gbps接続に対してはシングルモードファイバーインターフェースが必要
・802.1Q VLAN：VLANにタグをつけて、一意に識別するための技術
・**BFD（Bidirectional Forwarding Detection）は1秒以内にBGPの経路に問題がないかを検知することができる**
（Direct Connectのデフォルト検知では、最小で3秒の待機時間が発生する）

・DirectConnectの申請には、LOA-CFAを取得して行う
・Local Pref属性：BGPで特定の出口を優先してトラフィックを送るように制御する（デフォルト値は100）
・MEDは、同一の隣接ASからの複数のルートに対して相対的な優先度を提供する。
→低いMED値を持つルートは、同じ隣接ASからの他のルートよりも優先される。
・ASPATH（Autonomous System Path）；AS間の経路追跡するためのプロトコル
・DirectConnectのMACSec(Media Access Control Security)はイーサネットネットワークにおけるリンク層のセキュリティを提供するための規格。これを使用するには、MACSecをサポートするネットワークデバイスとDirectConnectが必要
・BGPコミュニティタグ：BGPルーティングポリシーの決定に役立つ情報を表すためにBGP更新メッセージに付加されるオプショナルな属性
・インバウンド（オンプレミス→AWS）
・7224:7100 - 優先設定：低
・7224:7200 - 優先設定：中
・7224:7300 - 優先設定：高
・7224:9100 - ローカルAWSリージョン
・7224:9200 -大陸内のすべてのAWSリージョン
・7224:9300 -Global(すべてのAWSリージョン)
・アウトバウンド（AWS→オンプレミス）
・7224:8100 - DirectConnectのプレゼンスポイントが関連づけられているAWSリージョンと同じリージョンから送信されるルート
・7224:8200 - DirectConnectのプレゼンスポイントが関連づけられている大陸と同じ大陸から送信されるルート

・経路選択の優先順位
Local Preference > AS-Path Prepend > Multi Exit Discriminator(MED)

・CGWは単一障害点なので、別のデータセンターからVPN接続を構築し、冗長化することで可用性を高める
・**一つの仮想プライベートゲートウェイ（VGW）に対して最大10個のカスタマーゲートウェイ（CGW)を関連づけることができる**
・DirectConnectGatewayで２つのVPCをVGWを通して繋げる。
２つのVPCをVPC AとVPC Bとすると、VPC AとVPC BはDirectConnectGatewayを介して推移的に接続することができないが、VPC AとVPC Bが異なる
DirectConnectGatewayにつながっている場合は、接続可能。
・S3用のインターフェースエンドポイント
オンプレミス > DirectConnect > VPCインターフェースエンドポイント > S3
（オンプレミスでS3インターフェースエンドポイントのDNS名を構成する必要がある。）
・BGPでアドバタイズできるルートは100未満
・7224:8100：AWS Direct Connect のプレゼンスポイントが関連づけられているAWSリージョンと同じリージョンから送信されるルート、
7224:8200：Direct Connectのプレゼンスポイントが関連づけられている大陸と同じ大陸から送信されるルート
・サイト間VPN接続またはDirect Connect接続から伝搬されたルートに他の既存の静的ルートと同じ宛先CIDRブロックがある場合、最長プレフィックスマッチは適用でき

ない。
・ルートの優先順位は、静的ルート＞DirectConnectルート > VPNルートになるので、同じ宛先でオンプレミスのルートを伝搬した場合、インターネットゲートウェイが優先される。
・AS Path Prepend：BGP（Border Gateway Protocol）のルーティングポリシーを制御するための一般的な戦略で、特定のネットワークパスを他のものよりも不適切または不望ましいと見なすために使用される。（特定のネットワークパスを他のパスよりも優先度が低いと見なしたい場合に、そのパスのAS PathにAS番号（通常は自分のAS番号）を追加（"prepend"）することで、AS Pathが人工的に長くなり、BGPが他のルートを選択するようになる。）
・Transit VIF（Virtual Interface）はDirectConnectの物理接続を複数の独立したネットワーク接続（複数のVPC接続）を可能にするための仮想インターフェース
・AWS Direct Connect Gatewayは異なるAWSリージョンに存在する仮想プライベートクラウドを1つのゲートウェイに接続できるようにする
→複数のリージョンに渡るVPCの接続、高速で安定した接続
・オンプレミス > DirectConnect > VPC > VPCエンドポイント > S3,DynamoDBの接続は不可
・TCP ポート 179は、BGP (Border Gateway Protocol) のために予約されている。
・Jumbo Frameをサポート（Private、Transit VIFだけ）
・Private：MTU 9001バイト
・Transit：MTU 8500バイト
・Direct Connect Gateway
・無料
・Direct Connect Gatewayに繋がったVPC間の推移的な通信はできない
同様にカスタマーセンター同士の通信も不可（SiteLinkでDirect Connect Gatewayを介してデータセンター間の通信ができるようになった）
→Transit VIFの場合、Transit Gateway PeeringでVPC間の推移的通信ができるようになる。
→カスタマーセンター同士を繋げたい場合は、Direct Connect Gatewayを2つ作って、Transit Gateway Peering経由で推移的な接続ができるようになる。
・同じAWSアカウント内で設定可能
・最大で100ルートの伝搬が可能（足りない場合は、CIDRをまとめる）
・2つのPrivate VIFを一つのDirect Connect Gatewayに接続することは可能（高可用性）
（一つのDirect Connect Gatewayには30このVIFを関連づけることが可能）
・Transit VIFを使ったDirect Connect Gatewayの場合
・1つのDirect Connect Gatewayには3つのTransit Gatewayが関連づけれる）
・1つのTransit Gatewayが伝搬できるルートは20（足りない場合は、CIDRを要約する）
・1つのDirect Connectで作成できるTransit VIFは1つ
・Direct Connect SiteLink
・Direct Connect Gatewayを介してデータセンター同士を接続できる
・PrivateとTransitVIFで有効にできる
・

〇Public ASN（公開ASN）とPrivate ASN（プライベートASN）

・Public ASN
・インターネット上で一意に公開されるASN
・セカンダリコネクションには、AS_Path Prependsで設定する
・Local Preferenceを使って、オンプレミスルーターがAWSに接続するときに正しいパスを選択するようにする。

・Private ASN
・プライベートなネットワーク内で使用されるASN
・より具体的なプレフィックスを使ってルートを決める。

〇VPNの接続パターン

1.クライアント > Client VPN > VPNエンドポイント > VPC
2.オンプレミス > Site-to-SiteVPN > VGW > VPC
3.オンプレミス > Site-to-SiteVPN > TGW > VPC
・複数のオンプレミスCGWとVGWを繋ぐことで、VGWを経由してオンプレミス同士の接続ができる。（CloudHub）
→CGWのASNを別々の値に設定する必要あり。
→ASNが同じCGWは通信されない。
→もしくはCGWの前にFirewallでフィルタリングする。
・Site-to-Site VPNではBFDをサポートしていないので、より早く障害を検知したい場合は、DPDを使う
・料金はVPNごとの時間料金＋データ転送料金
・1つのSite-to-Site VPN接続はVPCごとに作成される。

〇Site-to-Site VPN

・オンプレミスとVPCをインターネットを介さずに通信したいときに使うのが、VPN(Site to Site VPN)
・ターゲットゲートウェイ
・VGWもしくはTGW
・認証
・固定パブリックIPの場合は、事前共有キー（Pre-Shared Key）、固定パブリックIPでない場合は、プライベート証明書
・事前共有キー：対称暗号方式でセットアップが簡単。
・プライベート証明書：非対称暗号方式でセットアップが煩雑。ACMによる発行。
・MTU
・パケットサイズが1399以下となるように設定
（例：$ sudo ip link set dev eth0 mtu 1399）
・複数のVPN接続を束ね、Equal Cost Multi Path(ECMP)を利用し帯域を増す
（1つのIPsecトンネル当たり、最大1.25Gbps VPN接続を増やすことで、最大50Gbpsまでのバーストを検証済み）
・Acceleratedサイト間VPNオプション：AWS Global Acceleratorを使用してサイト間のVPN接続を高速化する。
→低遅延、高いスループット、冗長性と可用性

・CGW は AWS 側で作成する。EC2 のインスタンスで CGW 用のルーターを模擬できる。
・VPN はインターネットを介して通信が行われるため、パブリックアドレス IP が必要
・VPN は内部で冗長化されている。
・VPC 側のルートテーブルの編集から、VGW をターゲット、オンプレミスを宛先としたルートおよびルート伝搬（VGW を指定する）の設定を行う。
・オンプレミスのルートテーブルを、ターゲットを CGW、宛先を VPC として設定する。
・IPv4、IPv6 どちらもサポートしている。
・VPN の IPSec 通信には、**IP プロトコル番号 50 (ESP) および UDP ポート番号 500 (ISAKMP) の許可設定が必要**
（ESP：IPSec の暗号化セキュリティペイロード (ESP) 、ISAKMP：インターネットセキュリティアソシエーションキーマネージメントプロトコルでセキュリティ関連のパラメータをネゴシエートし、暗号鍵の交換を行うことで、安全な通信チャンネルを確立する。）
・IPSec デッドピア検出（DPD）：VPN 接続で他方のピア（エンドポイント）がまだ応答可能であるかどうかを検出するメカニズム。
・VPN 接続において、Active,Active の場合、片方のルートの優先度を上げることはできない。
・IPSec とは、データストリームの各 IP パケットを認証して暗号化すること
・クライアント VPN の使用には、クライアント側でクライアント VPN をインストールする必要がある。
・VPN 接続には Active/Passive 設定がある。（DirectConnect にはない。）
・Site-to-Site VPN 接続の高速化：Site-to-Site VPN 接続と Global Accelerator を組み合わせたアクセラレーションの有効化が可能。
（Transit Gateway にアタッチされた Site-to-Site VPN のみ設定可能）
・IPSec では、ESP（Encapsulating Security Payload）と呼ばれる IPSec プロトコルの一部としてプロトコル 50 が用いられる。
・VPN は内部で冗長化されており、パブリップ IP も 2 つある。
・VGW は VPC ごとに作成する必要あり。
・VGW は AES-256 と SHA-2 による暗号化をサポートしている
・BGP を使う場合、VGW の ASN は 64512～65534 の範囲（定義しなかったら、デフォルトの 64512 が割り当てられる）
・

○Transit VPC
・Hub：VPN ソフトウェアを備えた EC2 で通信を制御する（PaloAlto,Avitarix など）
・マルチ AZ による可用性を高める
・それぞれの SpokeVPC には、VGW を設置する
・中央の VPC がセキュリティチェックなどをしたいときに使える
・VPC 間での推移的通信はできない
・VPC とオンプレミスの CIDR が重複していても接続することができる
・オンプレミス →TransitVPC の IGW でインターネットへの通信ができる。
・TranisitGateway；TransitVPC と似ているが、TransitGateway の方が構成がシンプル
（オンプレミスと VPC での CIDR 重複はできない）
・

○AWS Client VPN

・Client端末からAWSにプライベートに接続したいときは、Client VPN Endpointを使う

クライアント端末にVPNソフトウェアのインストールが必要。ACMに認証キーをインポートする必要あり。

・ClientVPNEndpointはVPNTargetSubnetのENIと対応づける。（ひとつのCientVPNEndpointはひとつのサブネットにしか対応づけれない）

○CloudHubとは

・複数拠点間の通信をAWSで折り返す

（オンプレミスのCGWにはASNを設定する必要がある。）

（オンプレミスAのCGW > VPNルート > VGW > VPNルート > オンプレミスBのCGW）

○Route53

・DNSフォワーダー（DNS Forwarder）は、DNSクエリを解決できないローカルのDNSサーバーが、そのクエリを他のDNSサーバーに転送するための仕組み

・Route53はオンプレミスとAWS環境のルーティングを配分することができる

・Amazon Route53ResolverはAWSとオンプレミスでDNS解決をするためのサービス

AWS > オンプレミス：アウトバウンドエンドポイント、オンプレミス > AWS：インバウンドエンドポイント

・DNSルックアップはポート53でTCPとUDPを両方使用する。

（TCP：DNSゾーン転送や大きなクエリなど、大量のデータを送信するために使用される）

（UDP：一般的には、小さなクエリを送信するために使われる）

・Route53のヘルスチェッカーはVPC外にあるため、IPアドレスを使用してVPC内のエンドポイントの正常性をチェックするには、VPC内のインスタンスにパブリックIPアドレスを割り当てる必要がある。もしくはCloudWatchアラームを設定して、それをヘルスチェックすることにより、プライベートIPのインスタンスを間接的にヘルスチェックすることができる。

・オンプレミスからVPCのRoute53プライベートホストゾーンを参照するためには、EC2ベースのDNSフォワーダーが必要。

・ネストされたRoute53は、先に加重ルーティングを設定する

・DNS逆引きとは、IPアドレスからドメイン名を取得すること。

・VPCエンドポイントのDNS名前解決について

DNS名前解決を有効化していれば、サービス名で接続することができる(EC2.us-east-1.amazonaws.comなど)

有効化していない場合、VPCエンドポイントのDNSに対してリクエストを送る必要がある。

○DHCPオプションセットとRoute53 Resolverエンドポイントによる VPC⇔オンプレミスのDNSの違い

- **DHCP**
  - VPCのリソースがオンプレミスのDNSサーバに対してクエリを送信できる。（オンプレミスからVPCはできない）
  - インスタンスが使用するDNSサーバのIPアドレスを指定することができる。
  - カスタムドメイン名の設定：DHCPオプションセットは、VPC内のインスタンスに対してカスタムドメイン名を提供することも可能。
  - DHCPオプションセットはインスタンス内の名前解決。Route53はDNSに関連する幅広い機能を提供
  - 設定が簡単
- **Route53 Resolver**
  - オンプレミス⇔VPC双方向の通信が可能
  - 設定が複雑

○Route53のDNSSEC署名 (Domain Name System Security Extensions)

- DNS応答の改ざんを防止するためのセキュリティプロトコル
- 設定手順
  1.パブリックホストゾーンを作成（プライベートホストゾーンでは無効）
  2.そのホストゾーンでDNSSEC署名を有効にする
  3.DSレコードをドメイン登録業者に提供する
- ２種類の鍵が使用される
  - KSK(Key Signing Key)
  - ZSK(Zone Signing Key)

○CNAMEとエイリアスレコードの違い

- CNAME
  - ZoneApexでの登録は不可能（例：yurdomain.com）

- エイリアスレコード
  - ZoneApexでの登録が可能

CNAMEを用いた名前解決の応答例

www.example.com.	60	IN	CNAME	www-a.example.com.
www-a.example.com.	60	IN	CNAME	xxxx.cloudfront.net.
xxxx.cloudfront.net	60	IN	A	192.0.2.3

最終的に必要とするレコードデータ

エイリアスを用いた名前解決の応答例

www.example.com.	60	IN	A	192.0.2.3

○Route53へのネームサーバ移行

- Amazon Route53 Hosted Zoneを構成する
  - RFC1034、1035形式のゾーンファイルをインポートしてHosted Zoneを構成できる
- ネームサーバに関連するリソースレコードのTTLを短縮する

・ネームサーバの切り替えに要する時間を短縮できる（60秒～3600秒程度に短縮することが多い）
・DNSSECを無効にする
・親ゾーンと子ゾーンでDelegation（権限委譲）の設定を変更する
・旧ネームサーバの廃止

◯Amazon Route 53 Resolver

・VPCに標準で備わるDNSサーバ（フォワーダ- + フルサービスリゾルバー）
・VPC＋2のIPアドレスでアクセス可能
（例：VPCのCIDRが10.0.0.0/16の場合、10.0.0.2でアクセス）
・DirectConnectもしくはVPNを作成している場合、Route53 Resolverを使用することで、VPC > オンプレミス（アウトバウンドエンドポイント）、オンプレミス > VPC（インバウンドエンドポイント）の名前解決が可能になる。
・転送ルールタイプ
・転送：指定したドメイン名のDNSクエリをネットワークのネームサーバーに転送するルールタイプ
・システム：リゾルバーが転送ルールで定義されている動作を選択的に上書きするようにするルールタイプ
・再起的：ルールの存在しないドメイン名の再帰リゾルバーとして機能するルールタイプ
・VPC→オンプレミスの名前解決では、リゾルバーエンドポインに転送ルールを設定する。
（ドメイン：onprem.internal、タイプ：転送）
・VPC→オンプレミス→インターネットの名前解決では、李ゾルバーエンドポイントに転送ルールとシステムルールを設定する。
（ドメイン：example.com、タイプ：転送　　ドメイン：www.example.com、タイプ：システム）
・オンプレミス > Route53 Resolver インバウンド > Route53 プライベートホストゾーン > 各種AWSサービス
・Route53 Resolverルールは RAMでほかAWSアカウントと共有できる。
・Forwarding Ruleは System Ruleで上書きされる
・Resolverはリージョン単位
・Resolver Query Loggingでログが出力できる（CloudWatchLogs S3 Kinesis Data Stream）
・

〇Global Accelerator

・Global Acceleratorはリクエストを最も近い正常な利用可能なエンドポイントに再ルーティングする際に、自動フェイルオーバー機能を備えた**固定IP**を提供する。
・ALBとGlobal Acceleratorを使用することで、エンドポイントで常にクライアントIPアドレスの保持が可能になる。
・Global Acceleratorのカスタムルーティングアクセラレータ：一人以上のユーザーを特定のインスタンスに決定的にルーティングする。
→ユーザートラフィックがEC2インスタンスのどのセッションに送信されるかを制御するのに役立つ。

〇ELBの基本機能

・プライベートサブネットはALBを介してインターネットと通信できない。NATGatewayが必要

→もしくはNLBのエンドポイント経由でPrivateLinkを確立して通信する。

・Proxy ProtocolとX-Forwarded-For：バックエンドサーバーに接続する時に、プロトコルレベルでクライアントの接続情報を追加するためのメカニズム

Proxy Protocol：TCPレベル、X-Forwarded-For：HTTP,HTTPSレベル

・ELBのリスナー：ロードバランサーがListenするプロトコルとポート番号（1～65535) とロードバランサーからターゲットへの接続用のプロトコルとポート番号などを設

定

・ELBへは基本的にDNS名でアクセス（会社のドメインを指定する場合は、CNAMEもしくはエイリアスレコードで登録）

・ELBは負荷に応じて自動でスケールする。ALB/CLBはPre-Warmingの申請をサポートケースにて行う。これでスケールが間に合わなくなる心配がなくなる。（間に合わ

ない場合は、HTTP503を返す）

・CloudWatchによりELBはモニタリングできる。アクセスログも5分間隔で取得可能。S3バケットに保存される。

・**Gateway Load Balancer**

・GWLBエンドポイントのルートテーブルに、**GWLBのルートは記載する必要なし。**IGWだけのルートで良い。

（Destination:0,0,0,0　Target:igw-xxxxxx）

・GENEVE protocol on UDP port 6081

・IPv4のみサポート

・セキュリティグループの関連付けはできない。ターゲットのSGでGWLBからのIPアドレスを許可する。

・MTUは8500バイト

・Gateway Endpointを経由するインスタンスには、Route TableにTarget：Gateway Endpointを追加する

・Ingress Routingのルートテーブルにも、Target：Gateway Endpointを追加する。

・アルゴリズム：ALBはラウンドロビン、NLBはフローハッシュアルゴリズム、CLBはTCPがラウンドロビン

・**Application Load Balancer**

・IPアドレスをターゲットに設定することで、オンプレミスのサーバーにも接続できる

・ALBのアクセスログ機能は、クライアントIPアドレス、ターゲットIPアドレス、ターゲットポート、ユーザーエージェントを含むログが収集される。

・ALBはVPCPeering接続で、他のVPC内のEC2に接続できる

・地理的制限やIPによる制限ができない

→WAFをALBの前にデプロイして、WAF地理一致ステートメント、WAF IP setステートメントを使う

・SSL/TLS証明書を使うには、リージョンごと、FQDNごとにACMで証明書を発行して関連づける必要がある。
・ALBとCLBはセキュリティグループをアタッチできる。NLBはむり
・Connection Draining：バックエンドのEC2をELBから登録解除したり、ヘルスチェックが失敗したときに、新規リクエストの割り振りは中止して、処理中のリクエストは終わるまで一定期間まつ。
・スティッキーセッション：同じユーザーから来たリクエストを同じEC2に送る。（セッション情報はRDSなどに保存する）
・SSL/TLS Termination：ELBをSSL終端として使える。（EC2でSSL処理しなくて済む。）、SSLをバイパスしてバックエンドにTCPで送信。
・事前定義されたセキュリティポリシー：SSL/TLS利用時には、事前定義されたセキュリティポリシーを利用する。
・HTTPS/SSL利用時のTLSサーバ証明書：ACMを使用すれば証明書のリクエスト、管理、更新、プロビジョニングが容易に実行可能
・ALBはカスタムセキュリティポリシーをサポートしない
・ALBはオンプレミスとAWS間で負荷分散を行うことができる
・ALBにはホスト条件とパス条件がある。
（それぞれの用途は、ホスト条件はexample1.com,example2.comの両方を同じALBにポイントして、それぞれ異なるターゲットグループにルーティングするという場合。パス条件はexample.com/productsとexample.com/blogsのリクエストをそれぞれ異なるターゲットにルーティングするが考えられる。）

・**Network Load Balancer**
・トラフィックの送信元IPはNLBではなく、実際のクライアントIPもしくはポートを設定する
・NLBのセキュリティグループ
→NLBからのトラフィックは、直接インスタンスに送信されるので、NLB自体にセキュリティグループを設定できない。
→代わりに、NLBのターゲットであるEC2に対してセキュリティグループを設定する。
・NLBはTCPリスナーを使用して、トラフィックを復号化せずに通過させることができる
・NLBはターゲットグループごと55,000接続/分をサポートする。
・NLBとVPCエンドポイントはIPアドレスが重複していても機能する。
・固定IPアドレス（自動割り当てされたIPアドレス、またはNLB作成時に指定した自分が持っているElastic IPのいずれか）
・NLBはターゲットグループごとに最大5,5000接続/分をサポート
・NLB用のPrivateLinkエンドポイントを作成することで、ほかAWSアカウントのIPアドレスを用いてEC2にプライベートな接続をすることができる
・Network Load Balancerには最低8つのIPアドレスが必要

〇Proxy-protocolとX-Fowarded-Forの違い
・Proxy ProtocolはTCP通信で使用できる（NLBでは、使わなくてもクライアントIPが保持される）
・X-Forwarded-ForはHTTP通信のヘッダーに付与する形でIPアドレスを保持する

・VPC Peering、VPC Endpoints、VPC PrivateLink

・VPC Interface Endpointsは他AWSサービスのAPIを呼び出すために、セキュリティグループで**TCPポート443(HTTPS)のアウトバウンド**（ソースはVPCのCIDRアドレス）を許可する必要がある。

○AWS WAF

・AWS WAFのレートベースのルール機能：5分以内に大量のHTTPリクエストを行う送信元IPアドレスを検出し、問題のある送信もとIPからのリクエストを自動的にブロックする

・AWS WAFでは、geo matchステートメントにより特定の国をブロックすることができる。

特定のIPを許可するには、WAF IP setステートメントを作成する。

○AWS GuardDuty

・最小限の操作でDNSリクエストとVPCフローログを検査することで、トラフィックパターンを分析できる。そのほかにも、S3ログ、CloudTrail管理イベントログ、DNSログ、EBSボリュームデータ、Kubernetes Audit Logs、RDSログインアクティビティのAWS CloudTrailデータイベントなどのデータソースを分析して処理するモニタリングサービス。

○CloudFront

・CloudFrontはエッジロケーションにルーティングされ、キャッシュされたコンテンツがあればそれを提供することで高速化できる。

・CloudFrontによるCustom Headerの制限
→CloudFront側でリクエストに対してCustom Headerを指定する設定を行う
→ALBでCustomHeaderがない場合、通信をDenyする設定を行う。

・CloudFrontはSSL/TLS終端を行うことができる。

・CloudFrontの署名付きURLと署名付きCookieのユースケース
署名付きURL：アプリケーションのインストールのダウンロードなど、個々のファイルへのアクセスを制限する場合（有効期限の日時などの追加情報が含まれている）
署名付き Cookie： 現在の URL を変更したくない場合、または複数の制限されたファイル (メンバー内のすべてのファイルなど) へのアクセスを提供したい場合に、コンテンツにアクセスできるユーザーを制御できる。

・CloudFront署名付きURLの定型ポリシー（有効期限の日時を指定できる）
カスタムポリシー（有効期限だけでなく、**特定のIPアドレスからのアクセスを許可したり、アクセスする日時を指定**できる。）

・HTTP5002(Bad Gateway)：オリジンサーバとの通信に問題がある。

HTTP500(INTERNAL ERROR)：サーバ内部で問題が発生したこと。
・AWS Lambda@Edgeは、CloudFrontのエッジロケーションでで AWS Lambdaを実行するためのサービス
（S3バケットやHTTPサーバなどに転送する前に、リクエストのヘッダーや認証トークンを検査することができる。）

〇拡張ネットワーキング
・**SR-IOV(Single Root IO Virtualization)**は、物理的なI/Oリソース（ネットワークインターフェースなど）を複数の仮想I/Oリソースに分割するための標準化された仮想化方法
・AWSの**拡張ネットワーキング**は、Amazon EC2インスタンスでSR-IOVを使用するための機能。
（拡張ネットワーキングは、ネットワークを集中的に使用するワークロード（例えば、高パフォーマンスコンピューティング（HPC）、データ分析、ネットワーク強化のアプ
リケーションなど）に特に適している。）
・EFAはENAの拡張版
→Linuxのみ動作する。Windowsはただの ENAとして動作する
→拡張ネットワーキングと併用はできない

〇拡張ネットワーキングとクラスタープレイスメントグループの違い
・拡張ネットワーキング：EC2インスタンス間で高パフォーマンス、低遅延のネットワークを提供(マシンラーニングなどに使う)SR-IOVという技術を使用して、高PPSを実現する。
・クラスタープレイスメントグループ：単一のAZに配置されたEC2のローカルネットワークでの低遅延通信を可能にする。常にEC2間の通信が必要な場合に有効

◯VPCエンドポイント
・ゲートウェイVPCエンドポイント
・VPCエンドポイント用のサブネットのルートテーブルをに追加する必要あり。
・VPCのDNS解決を有効にする必要あり

・インターフェースVPCエンドポイント
・ENIに対してアクセスするため、アクセス制御はセキュリティグループで行う。
（対象のEC2からHTTPS(443)を受け取れるセキュリティグループとするのが基本）
・有料
・DNS 解決と DNS ホスト名を有効にする
・VPCエンドポイント用のルートテーブルは設定必要なし
・Route53 Resolverに自動で登録された、各サービスのDNS名で名前解決ができるようになる。

・

〇各AWSリソースのMTU

1. **EC2**: EC2 インスタンスは標準的に1500バイトのMTUをサポートする。
   また、一部のインスタンスタイプでは、ネットワークパフォーマンス強化のためにジャンボフレーム（MTU 9001バイト）をサポートする。
2. **EBS**: EC2インスタンスとEBSボリューム間の通信は、EC2インスタンスがサポートするMTUサイズに依存する。
3. **VPC内**: MTU 9001バイト（ジャンボフレーム）をサポートする
4. **AWS Direct Connect**: MTU 1500バイトまたは9001バイト（ジャンボフレームをサポート）
5. **RDS**: RDSは、ネットワークパフォーマンスに影響を与えるMTUサイズの設定は提供していない。
6. **VPCピアリング**
   ・intraリージョン：9001バイト
   ・interリージョン：1500バイト
7. **インターネットゲートウェイ：**MTU 1500バイト
8. **VPN：**MTU 1500バイト
9. **VPCエンドポイント：**MTU 8500バイト

〇Elastic Kubernetes Service(EKS)

・Container：アプリケーションのコード、ランタイム、システムツール、システムライブラリなど、アプリケーションが正常に動作するために必要なすべてを含む。
・Pod：Kubernetesにおけるデプロイ可能な最小の単位。1つのPodは、1つ以上のコンテナ（通常は1つ）とストレージリソース、独自のネットワークIP、コンテナが実行するためのオプションをまとめたもの。
・Node：Podを実行するために必要なサービスを含む。
・PodはIPアドレス単位。Nodeはインスタンス単位。
・ELBとEKS
  ・NLBの場合：
    ・externalTrafficPolicy
      ・Cluster：ノードにpodがない場合、他のノードにルーティングされる。クライアントIPは保持されない
      ・local：ノードにpodがない場合、ドロップされる。クライアントIPが保持される
  ・ALBの場合：
    ・X-Forwarded-Forでクライアント IPを保持できる

〇VPC PeeringとPrivate Linkの違い

・**VPCピアリング**
・双方向の通信が可能にするもの
・**VPC Peeringは最大125でCIDRオーバーラッピングができない。**

・**PrivateLink(VPCインターフェースエンドポイント)**
・あるVPCから別のVPCへ通信を行うための一方向の通信を行う。
（VPC内の1つのアプリケーションに接続したいときに使う。）
・CIDRが重複していても通信可能
・VPC Interface Endpointを使用するためには、DNS名前解決が必要
・**PrivateLinkは制限なしでCIDRオーバーラッピングもできる**
・**1000のVPCを接続できる**
・**Network Load Balancer(Service VPC)とENI(Customer VPC)もしくはGWLBが必要**

〇コマンド
$ip addr：IPアドレスが確認できる。
$show ip route：ネットワークデバイスのルートテーブルに関する情報が得られるBGPが学習したルートを確認できる。
$show ip bgp：BGPのルーティングテーブルに関する情報が得られる。（ASパス、宛先ネットワーク、経路属性など）
$configure；ネットワーキング機器やネットワーク管理ソフトウェアで使用される一般的なコマンド
$curl -s https://example.com/api/data
https://example.com/api/dataにGETリクエストを送信し、レスポンスの内容を取得する。
$ echo $(hostname -f)：カレントホスト名を取得（-fはフルホスト名を表示するためのコマンド）
example.com
$dig：ドメイン名の解決やDNSレコードの取得、DNSサーバの設定確認など
$nslookup：指定したドメイン名やホスト名に関連する情報を取得するためのコマンド。
$ip link show eth0：MTUをチェックできる
$sudo ip link set dev eth0 mtu 9001：MTU値をLinuxにセットできる
$tracepath amazon.com：クライアントデバイスとターゲットエンドポイント間のMTUをチェックできる

〇証明書
・EC2には自己証明書とCAによる証明書の選択肢がある。
・自己証明書は内部通信で使われる。（OpenSSL）

〇intraリージョンとinterリージョンの違い

・intra リージョン：同じ AWS リージョン内での通信
・inter リージョン：異なる AWS リージョン間の通信

◯AWS BlackBelt

・Gateway Load Balancer
・AWS Direct Connect
・AWS Transit Gateway
・Elastic Load Balancing (ELB)
・AWS re:Invent 2022 - Improve performance and availability with AWS Global Accelerator (NET301)
・AWS Site-to-Site VPN
・Amazon Route 53 導入編【AWS Black Belt】

◯BlackBelt 資料

・Amazon Route 53 Hosted zone 編
・Amazon Route 53 Resolver
・AWS Site-to-Site VPN
・Gateway Load Balancer
・オンプレミスと **AWS** 間の冗長化接続
・Elastic Load Balancing (ELB)

〇AWS ハンズオン (https://aws.amazon.com/jp/events/aws-event-resource/hands-on/)

・Network 編#1 AWS 上にセキュアなプライベートネットワーク空間を作成する
・Network 編#2 Amazon VPC 間および Amazon VPC とオンプレミスのプライベートネットワーク接続
・Network 編#3 クライアント VPN をつかって、リモート接続環境を構築しよう
・Amazon CloudFront および AWS WAF を用いて エッジサービスの活用方法を学ぼう

〇AWS Training and Certification ()

Exam Prep: AWS Certified Advanced Networking - Specialty (ANS-C01)